

MANUAL DE REFERENCIA

**Mc
Graw
Hill**

SEXTA EDICIÓN

Richard Petersen

LINUX

Manual de referencia

Acerca del autor

Richard Petersen, MLIS, imparte cursos de Unix y C/C++ en la Universidad de California en Berkeley. Es el autor de *Linux: Manual de referencia* (las seis ediciones), *Red Hat Enterprise and Fedora Linux: The Complete Reference*, *Red Hat Linux*, *Linux Programming*, *Red Hat Linux Administrator's Reference*, *Linux Programmer's Reference*, *Introductory C with C++*, *Introductory Command Line Unix for Users* y muchos libros más. Es colaborador de linux.sys-con.com (*Linux World Magazine*) con artículos sobre IPv6, el sistema operativo Fedora, Yum, depósitos de Fedora, Global File System (GFS), administración de dispositivos udev y Hardware Abstraction Layer (HAL).

Acerca del revisor técnico

Dean Henrichsmeyer ha servido como revisor técnico de la edición anterior de *Linux: Manual de referencia* y de varias ediciones de otra obra, *Red Hat Linux: The Complete Reference*. Tiene licenciatura en informática y ha trabajado con Linux por más de una década. Actualmente es director de sitio en SourceForge, Inc., el grupo de medios responsable para sitios Web como SourceForge.net, Linux.com, Slashdot.org, freshmeat.net y ThinkGeek.com.

LINUX

Manual de referencia

Sexta edición

Richard Petersen

Traducción

Jorge Arturo Pineda Sánchez
Traductor profesional



MÉXICO • BOGOTÁ • BUENOS AIRES • CARACAS • GUATEMALA • LISBOA • MADRID
NUEVA YORK • SAN JUAN • SANTIAGO • AUCKLAND • LONDRES • MILÁN • MONTREAL
NUEVA DELHI • SAN FRANCISCO • SINGAPUR • ST. LOUIS • SIDNEY • TORONTO

Director editorial: Fernando Castellanos Rodríguez
Editor de desarrollo: Miguel Ángel Luna Ponce
Supervisora de producción: Jacqueline Brieño Álvarez
Tipografía y formación: Ma. Eugenia Carrillo M.

LINUX Manual de referencia
Sexta edición

Prohibida la reproducción total o parcial de esta obra,
por cualquier medio, sin la autorización escrita del editor.



Educación

DERECHOS RESERVADOS © 2009, respecto a la sexta edición en español por
McGRAW-HILL INTERAMERICANA EDITORES, S.A. DE C.V.

A Subsidiary of The McGraw-Hill Companies, Inc.

Corporativo Punta Santa Fe
Prolongación Paseo de la Reforma 1015 Torre A
Piso 17, Colonia Desarrollo Santa Fe,
Delegación Álvaro Obregón
C.P. 01376, México, D.F.

Miembro de la Cámara Nacional de la Industria Editorial Mexicana, Reg. Núm. 736

ISBN10: 970-10-6758-4

ISBN13: 978-970-10-6758-1

Translated from the 6th English edition of

Linux: The complete reference

By: Richard Petersen

ISBN: 978-0-07-149247-8

6789012345

0876543219

Impreso en México

Printed in Mexico

**Para mis sobrinas,
Aleina y Larisa**

Contenido

Parte I Introducción

1	Introducción a Linux.....	3
2	Primeros pasos.....	17

Parte II La shell y la estructura de archivos de Linux

3	La shell.....	35
4	Secuencias de comandos y programación de la shell.....	65
5	Configuración de la shell.....	89
6	Archivos, directorios y archiveros de Linux.....	115

Parte III Escritorio

7	X Windows System, Xorg y administradores de despliegue	145
8	GNOME.....	169
9	KDE	197

Parte IV Software de Linux

10	Administración de software	219
11	Aplicaciones de oficina y bases de datos	237
12	Herramientas gráficas y multimedia	255
13	Clientes de correo y noticias	265
14	Clientes Web, FTP y Java	281
15	Herramientas de red	301

Parte V Seguridad

16	Cifrado, verificaciones de integridad y firmas.....	313
17	Linux con seguridad mejorada	327
18	IPsec y redes privadas virtuales	349
19	Secure Shell y Kerberos	359
20	Firewalls	373

Parte VI Internet y servicios de red

21	Administración de servicios	401
22	Servidor FTP	423
23	Servidores Web	443
24	Servidores proxy	467
25	Servidores de correo	477
26	Servidores de impresión, noticias, búsqueda y bases de datos	503

Parte VII Administración del sistema

27	Administración básica del sistema	523
28	Administración de usuarios	551
29	Sistemas de archivos.....	583
30	RAID y LVM.....	615
31	Dispositivos y módulos	639
32	Administración del kernel.....	671
33	Administración de copias de seguridad	693

Parte VIII Servicios de administración de red

34	Administración de redes TCP/IP.....	707
35	Configuración automática de red con IPv6, DHCPv6 y DHCP	745
36	NFS y NIS.....	761
37	Sistemas de archivos de red distribuidos.....	777
	Dónde obtener distribuciones de Linux.....	785
	Índice.....	787

Agradecimientos

Quisiera agradecer a todos aquellos que en McGraw-Hill hicieron de este libro una realidad, sobre todo a Jane Brownlow, editora, por su continuo estímulo y análisis, además de la administración de este proyecto tan complejo; a Dean Henrichsmeyer, revisor técnico, cuyo análisis y sugerencias resultaron muy profundas y útiles; a Jennifer Housh, coordinadora de compras, que proporcionó recursos necesarios y consejos útiles; a Sally Engelfried, corrector de estilo, por su excelente trabajo en edición además de sus interesantes comentarios; a la administradora de proyecto, Sam RC, quien, junto con la directora editorial, Patty Mon, incorporaron gran cantidad de características encontradas en este libro, además de coordinar la intrincada tarea de generar la versión final. Gracias también a Scott Rogers, que inició el proyecto.

Agradecimientos especiales a Linus Torvalds, el creador de Linux, y a quienes siguen desarrollando Linux como un sistema operativo abierto, profesional y efectivo accesible para todos. También doy las gracias a la comunidad académica cuya dedicación especial ha desarrollado Unix como un sistema operativo flexible y versátil. También quisiera agradecer a profesores y estudiantes de la Universidad de California en Berkeley, por la experiencia y el apoyo para desarrollar nuevas y diferentes formas de entender las tecnologías del sistema operativo.

También quisiera agradecer a mis padres, George y Cecelia, y a mis hermanos, George, Robert y Mark, por su apoyo y ánimo para este proyecto tan difícil. También Valerie y Marylou y mis sobrinos y nietos, Aleina, Larisa, Justin, Christopher y Dylan, por su apoyo y recordatorios de límites de tiempo.

Introducción

El sistema operativo Linux se ha vuelto uno de los más importantes en uso hoy en día, porque trae a la PC todo el poder y la flexibilidad de las estaciones de trabajo Unix, además de un conjunto completo de aplicaciones de Internet y una interfaz de escritorio totalmente funcional. Este libro está diseñado no sólo para servir como referencia completa en Linux, sino también para proporcionar explicaciones detalladas y claras sobre las características de Linux. No se supone que debe tener conocimientos previos de Unix; Linux es un sistema operativo que cualquiera puede utilizar.

Con el gran número de distribuciones de Linux disponibles, es fácil perder de vista el hecho de que casi todas sus operaciones son las mismas. Todos utilizan el mismo escritorio, shell, sistemas de archivos, servidores, soporte de administración y configuraciones de red. Muchas distribuciones proporcionan sus propias herramientas GUI, pero éstas son sólo portales de los mismos comandos básicos de Linux. Este libro es independiente de distribuciones, porque proporciona una explicación detallada y concisa de tareas comunes de todos los sistemas Linux. Al menos el 95% del sistema Linux involucra operaciones que son las mismas para todas las distribuciones. Se utiliza este libro sin importar qué distribución de Linux particular esté utilizando.

Las distribuciones de Linux incluyen características que se han vuelto un estándar, como los escritorios; la compatibilidad de Unix; los servidores de red; y varias aplicaciones de software como aplicaciones de oficina, multimedia e Internet. GNOME y K Desktop Environment (KDE) se han vuelto los escritorios estándar de la interfaz gráfica de usuario (GUI, Graphical User Interfaces) para Linux, conocidos por su poder, flexibilidad y uso sencillo. Ambos se han vuelto componentes integrales de Linux, con aplicaciones y herramientas para cada tipo de tarea y operación.

Linux también es un sistema operativo Unix totalmente funcional. Tiene todas las características estándar de un sistema Unix poderoso, incluido un conjunto completo de shell de Unix como BASH, TCSH y Z. Quienes están familiarizados con la interfaz Unix utilizan cualquiera de estas shells, con los mismos comandos, filtros y características de configuración de Unix.

En Linux, opera un amplio conjunto de aplicaciones. En los depósitos de distribución se lanzan continuamente diversas aplicaciones de escritorio. El software de GNU General Public License (GPL) proporciona aplicaciones en un nivel profesional como herramientas de desarrollo de programación, editores y procesadores de palabra, además de varias aplicaciones especializadas como las gráficas o de sonido.

Cómo utilizar este libro

Este libro identifica siete temas principales de Linux: entornos de shell, escritorios, aplicaciones, seguridad, servidores, administración de sistema y administración de red. En realidad son varios

libros en uno (un libro de escritorio, uno de usuario de shell, uno de seguridad, uno de servidor y uno de administración); la manera en que desee usarlo dependerá de cómo quiera utilizar su sistema Linux. Casi todas las operaciones de Linux se llevan a cabo al utilizar la interfaz GNOME o KDE. Se puede concentrar en los capítulos de GNOME y KDE y sus herramientas y aplicaciones correspondientes en los diferentes capítulos del libro. Por otra parte, si quiere explorar a fondo los aspectos de Unix en Linux, revise los capítulos de shell y las correspondientes aplicaciones de shell en otros capítulos. Si sólo quiere utilizar Linux para sus aplicaciones y clientes de Internet, entonces concéntrese en la sección de aplicaciones. Si quiere utilizar Linux como un sistema de varios usuarios que da servicio a muchos usuarios o se integra en una red local, se utiliza la información detallada de administración de sistema, archivos y red que se proporciona en los capítulos. Ninguna de estas tareas es exclusiva. Si está trabajando en un entorno de negocios, tal vez quiera utilizar los tres aspectos. Los usuarios se concentran más en el escritorio y las aplicaciones, mientras que tal vez los administradores utilicen más las características de seguridad y red.

Temas por partes

En la primera parte de este libro se proporciona una revisión general y se cubren algunos de los temas de inicio que serán útiles para los usuarios. Se proporciona una introducción a las listas de recursos, sitios de software, sitios de documentación, grupos de noticias y sitios de desarrollo y noticias de Linux. Las distribuciones se cubren brevemente. En el siguiente capítulo se cubren los temas de inicio, como instalación general, lo básico de GNOME y KDE, además de acceso a Windows.

En la parte II de este libro se trata con los entornos de shell de Linux, que cubren las shell BASH y TCSH, secuencias de comandos Shell, configuración de shell y el sistema de archivos de Linux. Todos estos capítulos operan a partir de una interfaz de línea de comandos, que le permite administrar los archivos y las shells y acceder a ellos directamente.

En la parte III de este libro se cubren los escritorios y sus herramientas de soporte GUI, como X Window System y administradores de despliegue. Aquí se le presentarán los escritorios KDE y GNOME. Se describen con detalle diferentes características como applets, el Panel y herramientas de configuración.

En la parte IV del libro se analizan a fondo muchas aplicaciones de oficina, multimedia e Internet que se utilizan en su sistema Linux, que incluyen conjuntos de oficina como OpenOffice.org y KOffice. También se analizan los diferentes sistemas de administración de bases de datos disponibles, junto con las ubicaciones de sitio Web donde se descargan. Linux instala automáticamente aplicaciones de correo, noticias, FTP y explorador Web, además de servidores FTP y Web. KDE y GNOME vienen con un conjunto completo de correo, noticias, clientes FTP y exploradores Web.

En la parte V se muestra cómo implementar precauciones de seguridad al utilizar cifrado, autenticación y firewalls. La cobertura de GNU Privacy Guard (GPG) le muestra cómo implementar cifrado con base en claves públicas y privadas. Con Luks (Linux Unified Key Setup) se cifran con facilidad sistemas de archivos. SE Linux proporciona control refinado y completo de todas sus redes y los recursos del sistema. Las herramientas de IPsec le permiten utilizar el protocolo IPSEC para cifrar y autenticar transmisiones de red. Los temas de seguridad de red cubren firewalls y cifrado al utilizar Netfilter (IPtables) para proteger su sistema, Secure Shell (SSH) para proporcionar transacciones remotas seguras y Kerberos para proporcionar autenticación segura.

En la parte VI se analizan servidores de Internet que se ejecutan en Linux, incluidos FTP, Web y servidores de correo. El capítulo de servidor Web Apache cubre directivas de configuración estándar como las de indización automática, además de las nuevas directivas de hosts virtuales. También se cubren los servidores Web Sdnmail, Postfix, IMAP y POP, además de que se examinan el servidor de noticias INN, los servidores de impresión CUPS, el servidor de base de datos MySQL y el servidor proxy Squid.

En la parte VII se analizan temas de administración de sistema, incluida la administración de usuario, software, sistemas de archivos, sistema, dispositivo y kernel. Hay descripciones detalladas de archivos de configuración utilizados en tareas de administración y sobre la manera de crear entradas en éstos. En primer lugar, se cubren las tareas de administración de sistema básicas, como seleccionar niveles de ejecución, monitorear su sistema y programar apagados. Después, se analizan los aspectos de configuración y control de usuario y grupos. Se cubren los diferentes métodos de virtualización, como completa (KVM) y paravirtualización (Xen). También se cubren las diferentes tareas de sistemas de archivos, como montar sistemas de archivos, administrar sistemas de archivos con HAL y udev, y configurar dispositivos RAID y volúmenes LVM. Los dispositivos se detectan de manera automática con udev y la Capa de Abstracción de Hardware (HAL, Hardware Abstraction Layer).

En la parte VIII se cubren temas de administración de red, como configurar interfaces de red y direcciones IP. También se aprende cómo implementar su propio servidor de protocolo de configuración dinámica de host (DHCP) IPv4 para asignar direcciones IP de host de forma dinámica y cómo operan el direccionamiento automático y la reenumeración de IPv6. Se presentan las interfaces y los servicios de sistemas de archivos de red (NFS) como GFS versión 2, NFS para Unix y redes NIS.

PARTE

Introducción

CAPÍTULO 1

Introducción a Linux

CAPÍTULO 2

Primeros pasos

Introducción a Linux

Linux es un sistema operativo rápido y estable de fuente abierta para computadoras personales (PC) y estaciones de trabajo; ofrece servicios de Internet a nivel profesional, herramientas de desarrollo extensas, interfaces gráficas de usuario (GUIs) completamente funcionales y gran cantidad de aplicaciones que van desde suites para oficina, hasta aplicaciones multimedia. Linux fue desarrollado a principios de la década de 1990 por Linus Torvalds, junto con programadores de todo el mundo. Como sistema operativo, Linux realiza muchas funciones de Unix, Macintosh, Windows y Windows NT. Sin embargo, se distingue por su poder y flexibilidad, además de su disponibilidad gratuita. La mayor parte de los sistemas operativos de PC, como Windows, empezaron su desarrollo en los confines de PCs pequeñas y restringidas, que sólo recientemente se han vuelto máquinas más versátiles. Tales sistemas operativos se actualizan constantemente para mantenerse al día con las siempre cambiantes capacidades del hardware de PC. Linux, por otra parte, fue desarrollado en un contexto diferente. Linux es una versión para PC del sistema operativo Unix utilizado por décadas en mainframes y minicomputadoras; es el sistema elegido para servidores de red y estaciones de trabajo. Linux lleva la velocidad, eficiencia, escalabilidad y flexibilidad de Unix a su PC, aprovechando todas las capacidades que pueden proporcionar las PC.

Técnicamente, Linux consta del programa del sistema operativo, conocido como *kernel*, la parte desarrollada originalmente por Linus Torvalds. Pero siempre ha sido distribuido con gran número de aplicaciones de software, que van desde servidores de red y programas seguridad, hasta aplicaciones de oficina y herramientas de desarrollo. Linux ha evolucionado como parte del movimiento del software de fuente abierta, para el que programadores independientes unieron fuerzas, a fin de proporcionar software gratuito de gran calidad para cualquier usuario. Linux se ha vuelto la plataforma principal para software de fuente abierta, en gran medida creado por el proyecto GNU de Free Software Foundation. Muchas de estas aplicaciones se incluyen en las distribuciones estándar de Linux. Actualmente, hay miles de aplicaciones de fuente abierta disponibles para Linux en sitios como sourceforge.net de SourceForge, Inc., kde-apps.org de K Desktop Environment (KDE) y gnomefiles.org de GNU Network Object Model Enviromnments (GNOME). Casi todas también incorporadas en almacenes de distribución, haciendo uso de paquetes que siguen las normas de distribución.

Junto con las opciones del sistema operativo Linux, se incluyen poderosas características de red, entre las que hay soporte para Internet, intranets e interconexiones de Windows. Como norma, los distribuidores de Linux incluyen servidores de Internet rápidos, eficientes y estables, como los

servidores Web, de protocolo de transferencia de Archivos (FTP, File Transfer Protocol) y DNS, junto con servidores proxy, de noticias y correo electrónico. En otras palabras, Linux tiene todo lo necesario para configurar, dar soporte y mantenimiento a una red completamente funcional.

Con GNOME y KDE, Linux también ofrece GUIs con el mismo nivel de flexibilidad y poder. A diferencia de Windows y Mac, Linux permite elegir la interfaz deseada y luego ajustarla a su medida agregando paneles, applets, escritorios virtuales y menús, todo con capacidades completas para arrastrar y colocar, así como herramientas útiles para Internet.

Linux hace todo esto al precio justo. Linux es gratis, incluidos servidores de red y escritorios GUI. A diferencia del sistema operativo oficial de Unix, Linux se distribuye de manera gratuita mediante una licencia pública general GNU, como especifica la Free Software Foundation, haciéndolo disponible para cualquiera interesado en utilizarlo. GNU (el acrónimo representa “GNUs Not Unix”, GNU no es Unix) es un proyecto iniciado y administrado por la Free Software Foundation, para proporcionar software gratuito a usuarios, programadores y desarrolladores. Linux está registrado en derechos de autor, no es de dominio público. Sin embargo, una licencia pública GNU tiene casi el mismo efecto que si el software fuera de dominio público. La GPL (GNU Public Licence, licencia pública de GNU) está diseñada para asegurar que Linux se mantenga gratuito y, al mismo tiempo, estandarizado. Linux es técnicamente un sistema operativo de kernel (las operaciones centrales) y sólo existe un kernel oficial de Linux. Ocasionalmente, la gente tiene la impresión errónea de que Linux por alguna razón es un sistema operativo menos profesional porque es gratuito. Linux es, en realidad, una versión de Unix para PC, estaciones de trabajo y servidores. Muchos lo consideran mucho más estable y poderoso que Windows. Estas características han hecho que Linux sea el sistema operativo elegido para servidores de red.

Para apreciar Linux en su totalidad, necesita entender el contexto especial en que se desarrolló el sistema operativo Unix. Éste, a diferencia de gran parte de sistemas operativos, se desarrolló en un ambiente académico e investigación. Unix es el sistema usado con más frecuencia en universidades, laboratorios de investigación, centros de datos y empresas. Su desarrollo fue paralelo al de las computadoras y la revolución de las comunicaciones en décadas pasadas. Los profesionales en computación a menudo desarrollaron nuevas tecnologías computacionales en Unix. IBM, Sun y Hewlett-Packard venden y mantienen sus propias versiones de Unix. Las demandas únicas para programas de investigación a menudo requieren que Unix se ajuste a la medida de sus necesidades. Esta flexibilidad inherente en el diseño de Unix no disminuye su calidad. En realidad, esta flexibilidad confirma la fortaleza de Unix, permitiendo se adapte a casi cualquier entorno. Este es el contexto en que se desarrolló Linux. Por eso es, en este sentido, otra versión de Unix (una versión para PC). El hecho de que Linux se haya desarrollado por profesionales de la computación trabajando en un ambiente similar al de la investigación, refleja la manera en que solían desarrollarse las versiones de Unix. Linux da licencias públicas y gratuitas (reflejando el origen que Unix tiene en instituciones académicas, con su sentido de servicio y soporte público). Linux es un sistema operativo de primera clase, accesible para cualquier persona y completamente gratuito.

Distribuciones de Linux

A pesar de que sólo hay una versión estándar de Linux, existen varias distribuciones. Diferentes compañías y grupos han empaquetado Linux, así como software de Linux en diferentes formas. Cada compañía o grupo lanza el paquete de Linux, generalmente en CD-ROM. Los futuros lanzamientos pueden incluir versiones actualizadas de programas o software nuevo. Algunas distribuciones más populares son Red Hat, Ubuntu, Mepis, SUSE, Fedora y Debian. El kernel de Linux se distribuye centralmente a través de **kernel.org**. Todas las distribuciones usan el mismo kernel, aunque puede estar configurado de diferente forma.

Linux ha producido gran variedad de distribuciones. Muchas enfocadas en proporcionar una solución amplia que brinde soporte para todas y cada una de las tareas. Esto incluye distribuciones como SUSE, Red Hat y Ubuntu. Algunas son variaciones de otras distribuciones, como Centos, basada en Red Hat Enterprise Linux, y Ubuntu, derivada de Debian Linux. Otras se han desarrollado para tareas más especializadas o soporte de ciertas características. Distribuciones como Debian proporcionan desarrollos de vanguardia. Algunas distribuciones ofrecen versiones más comerciales, generalmente empaquetadas con aplicaciones como bases de datos o servidores seguros. Algunas compañías, Red Hat y Novell entre ellas, otorgan una distribución comercial correspondiente a otra gratuita con soporte técnico. La distribución gratuita se usa para desarrollar nuevas características, como Fedora Project para Red Hat. Otras distribuciones, Knoppix y Ubuntu por mencionar unas, se especializan en Live-CD, el sistema operativo completo de Linux en un solo CD.

En la actualidad, **distrowatch.com** presenta una lista de numerosas distribuciones de Linux. Revise el sitio para conocer detalles acerca de distribuciones actuales. En la tabla 1-1 se muestra una lista de sitios Web con varias distribuciones populares de Linux. Los sitios FTP de estas usan el prefijo **ftp** en vez de **www**, como **ftp.redhat.com**. También se muestra en la lista de la tabla 1-1 el sitio del kernel de Linux, donde se proveen los lanzamientos más recientes del kernel oficial de Linux. Estos sitios corresponden a portales FTP donde puede descargar actualizaciones y versiones nuevas.

NOTA Las distribuciones utilizarán sus propios programas de instalación y actualización. Revise la documentación del distribuidor para conocer más detalles.

URL	Descripción del sitio
redhat.com	Red Hat Linux
fedoraproject.org	Fedora Linux
centos.org	Centos Linux
opensuse.com	openSUSE Linux
debian.org	Debian Linux
ubuntu.com	Ubuntu Linux
mepis.org	Mepis Linux
gentoo.org	Gentoo Linux
turbolinux.com	Turbo Linux
knoppix.org	Knoppix Linux
linuxiso.com	Imágenes CD-ROM ISO para distribuciones Linux
distrowatch.com	Información detallada acerca de las distribuciones Linux
kernel.org	Kernel de Linux

TABLA 1-1 Distribución de Linux y sitios Kernel

Sistemas operativos y Linux

Un *sistema operativo* es un programa para administrar hardware y software de computadora para el usuario. Los sistemas operativos originalmente fueron diseñados para realizar tareas de hardware repetitivas, centradas en administración de archivos, ejecución de programas y recepción de comandos del usuario. La interacción con un sistema operativo se da a través de una *interfaz de usuario*, permitiendo al sistema operativo recibir e interpretar instrucciones enviadas por el usuario. Sólo se necesita enviar una instrucción al sistema operativo para realizar una tarea, como leer un archivo o imprimir un documento. La interfaz de usuario de un sistema operativo puede ser tan simple que permita el ingreso de comandos en una línea o tan compleja que facilite la selección de menús e iconos en un escritorio.

Un sistema operativo también administra aplicaciones de software. Para realizar diferentes tareas, como editar documentos o realizar cálculos, necesita especificar las aplicaciones de software. Un *editor* es un ejemplo de una aplicación de software para editar un documento, hacer cambios y agregar texto nuevo. El editor por sí solo es un programa que consta de instrucciones que la computadora debe ejecutar. Para que se use el programa, primero debe cargarse en la memoria de la computadora y después se ejecutarán las instrucciones. El sistema operativo controla carga y ejecución de todos los programas, incluida cualquier aplicación de software. Cuando quiera usar un editor, sólo dé la instrucción al sistema operativo para cargar la aplicación del editor y ejecutarla.

La administración de archivos y programas, además de interacción con el usuario son características tradicionales, comunes en todos los sistemas operativos. Linux, como todas las versiones de Unix, agrega dos o más características. Linux es un sistema de multiusuario y multitareas. Al tratarse de un sistema *multitareas*, puede pedir al sistema realice varias tareas simultáneamente. Mientras efectúa una, puede trabajar en otra. Por ejemplo, editar un archivo mientras se imprime otro. No tiene que esperar al término de la impresión para editar otro archivo. Asimismo, al ser sistema *multiusuarios*, varios usuarios pueden iniciar sesión en el sistema al mismo tiempo, cada uno interactuando con el sistema a través de su propia terminal.

Al tratarse de una versión de Unix, Linux comparte la misma flexibilidad del sistema; flexibilidad surgida desde los orígenes de la investigación en Unix. Desarrollado por Ken Thompson en AT&T Bell Laboratories, a finales de la década de 1960 y principios de 1970, el sistema Unix incorporó numerosos desarrollos nuevos en el diseño de sistemas operativos. Originalmente, Unix se planteó como sistema operativo para investigadores. Una de las principales metas fue crear un sistema para apoyar las cambiantes demandas de los investigadores. Para lograrlo, Thompson debió diseñar un sistema capaz de manejar muchos tipos de tareas diferentes. La flexibilidad se volvió más importante que la eficiencia del hardware. Como Unix, Linux tiene la ventaja de afrontar diversas tareas con que pueda lidiar cualquier usuario. No se encuentra confinado a interacciones limitadas ni rígidas del sistema operativo. En cambio, está pensado para crear un conjunto de herramientas muy efectivas disponibles para el usuario. Esta filosofía orientada al usuario significa que se puede configurar y programar el sistema para satisfacer sus necesidades específicas. Con Linux, el sistema operativo se convierte en un entorno operativo.

La historia de Unix y Linux

Como una versión de Unix, Linux naturalmente tiene su origen en Unix. La historia comienza a finales de la década de 1960, cuando se dio un esfuerzo concertado para desarrollar nuevas técnicas en sistemas operativos. En 1968, un consorcio de investigadores pertenecientes a General Electric, AT&T Bell Laboratories y Massachusetts Institute of Technology, llevaron a cabo un proyecto de investigación especial de sistemas operativos denominado MULTICS (Multiplexed Information and

Computing Service, servicio de información y computación multiplexada). MULTICS incorporó nuevos conceptos en multitareas, administración de archivos e interacción con el usuario.

Unix

En 1969, Ken Thompson, Dennis Ritchie y los investigadores de AT&T Bell Laboratories desarrollaron el sistema operativo Unix, incorporando muchas características del proyecto de investigación MULTICS. Ellos hicieron el sistema a la medida de las necesidades de entornos de investigación, diseñándolo para ejecutarse en minicomputadoras. Desde el comienzo, Unix fue un sistema operativo multiusuario y multitareas, accesible y eficiente.

El sistema Unix se volvió popular en los laboratorios Bell a medida que más y más investigadores usaban el sistema. En 1973, Dennis Ritchie colaboró con Ken Thompson para reescribir el código del programa para el sistema Unix en el lenguaje de programación C. Unix gradualmente dejó de ser un diseño creado por una persona y pasó a ser un producto de software estándar, distribuido por muchos comercializadores, como Novell e IBM. Inicialmente, Unix fue tratado como producto de investigación. La primera versión de Unix se distribuyó sin costo alguno en los departamentos de informática de varias universidades destacadas. Durante la década de 1970, los laboratorios Bell comenzaron a publicar versiones oficiales de Unix y a dar licencias de los sistemas a diferentes usuarios. Uno de esos usuarios fue el departamento de informática de la universidad de California en Berkeley. Berkeley agregó muchas características nuevas al sistema, que más adelante se convirtieron en el estándar. En 1975, Berkeley lanzó su propia versión de Unix, conocida por su brazo de distribución, Berkeley Software Distribution (BSD). Esta versión BSD de Unix se convirtió en el competidor principal de la versión de AT&T Bell Labs. AT&T desarrolló distintas versiones de investigación de Unix y en 1983 lanzó la primera versión comercial, denominada System 3. Esta fue seguida por System V, que se convirtió en un producto de software con soporte comercial.

Al mismo tiempo, la versión BSD de Unix se desarrolló a través de varios lanzamientos. A finales de la década de 1970, BSD Unix se convirtió en la base de un proyecto de investigación de la Advanced Research Projects Agency (DARPA), del Departamento de Defensa de Estados Unidos. Como resultado, en 1983, Berkeley emitió una versión poderosa denominada BSD, versión 4.2. Este lanzamiento incluía una administración de archivos sofisticada, además de características de interconexión basadas en protocolos de red de Internet (los mismos que ahora se usan para Internet). BSD versión 4.2 se distribuyó ampliamente y adoptó por muchos vendedores, como Sun Microsystems.

A mediados de la década de 1980, surgieron dos estándares en competencia, uno basado en la versión Unix de AT&T y otro en la versión BSD. Unix System Laboratories de AT&T desarrolló System V versión 4. Varias otras compañías, como IBM y Hewlett-Packard, establecieron la Open Software Foundation (OSF), a fin de crear sus propias versiones estándar de Unix. Después existieron dos versiones estándar comerciales de Unix (la versión OSF y System V versión 4).

Linux

Diseñado originalmente de manera específica para PC basadas en Intel, Linux inició en la universidad de Helsinki, como proyecto personal de un estudiante de informática llamado Linus Torvalds. En ese momento, los estudiantes usaban un programa denominado *Minix*, presentando diferentes características de Unix. Minix fue creado por el profesor Andrew Tanenbaum y se distribuyó ampliamente a través de Internet a estudiantes de todo el mundo. La intención de Linus fue crear una versión eficaz para PC de Unix para los usuarios de Minix. Fue denominada Linux y, en 1991, Linus lanzó la versión 0.11. Linux se distribuyó ampliamente a través de Internet, en los

años siguientes, otros programadores lo refinaron y agregaron cosas, incorporando casi todas las aplicaciones y características ahora presentes en los sistemas estándar Unix. Todos los administradores de ventanas importantes han migrado a Linux. Éste tiene todas las herramientas de interconexión: soporte a FTP, exploradores Web, así como todo el rango de servicios de red: correo electrónico, servicio de nombres de dominio, además de configuración de host dinámico, junto con servidores FTP, Web y de impresión. También tiene un conjunto completo de utilidades para desarrollo de programas: compiladores y depuradores C++, por mencionar algunos. Dadas todas las características, el sistema operativo Linux se mantiene pequeño, estable y rápido. En su formato más simple, Linux puede ejecutarse de manera eficiente con sólo 2MB de memoria.

A pesar de que Linux se desarrolló en un entorno de Internet libre y abierto, se apega a estándares oficiales de Unix. Debido a la proliferación de versiones de Unix en décadas anteriores, el Institute of Electrical and Electronics Engineers (IEEE), desarrolló un estándar de Unix independiente para el American National Standard Institute (ANSI). A este nuevo Unix cumpliendo con el estándar ANSI se le denomina Portable Operating System Interface for Computer Enviroments (POSIX, interfaz transportable de sistema operativo para entornos computacionales). El estándar define cómo debe operar un sistema parecido a Unix, especificando tales detalles como llamadas e interfaces del sistema. POSIX define un estándar universal al que deben apegarse todas las versiones de Unix. Las más populares son compatibles ahora con POSIX. Linux fue desarrollado desde el principio acorde con el estándar POSIX. Linux también se adhiere a la jerarquía de archivos de sistema estándar (FHS), especificando la ubicación de archivos y directorios en la estructura de archivos de Linux. Consulte pathname.com/fhs para conocer más detalles.

El desarrollo de Linux es ahora supervisado por The Linux Foundation (linux-foundation.org), una fusión de The Free Standards Group y Open Source Development Labs (OSDL). Este es el grupo con que Linus Torvalds trabajó para desarrollar las nuevas versiones de Linux. Los kernels de Linux más recientes se publican en kernel.org.

Revisión de Linux

Cómo Unix, generalmente Linux puede dividirse en tres componentes principales: kernel, entorno y estructura de archivos. El *kernel* es el programa central para ejecución de programas y administración de dispositivos de hardware, como discos e impresoras. El *entorno* proporciona una interfaz para el usuario. Éste recibe comandos del usuario y los envía al kernel para su ejecución. La *estructura de archivos*, organiza la manera en que se almacenan los archivos en un dispositivo de almacenamiento, como un disco. Los archivos se organizan en directorios. Cada directorio puede contener cualquier cantidad de subdirectorios, cada uno de éstos almacenando archivos. Juntos, kernel, entorno y estructura de archivos, forman la estructura del sistema operativo básico. Con estos tres, puede ejecutar programas, administrar archivos e interactuar con el sistema.

Un entorno proporciona la interfaz entre kernel y usuario. Puede describirse como un intérprete. Esta interfaz interpreta los comandos ingresados por el usuario, enviándolos al kernel para ejecución. Linux proporciona varios tipos de entornos: escritorios, administradores de ventanas y shells de líneas de comandos. Cada usuario de un sistema Linux tiene su propia interfaz. Dependiendo de sus necesidades especiales, los usuarios pueden hacer sus entornos a la medida, ya sean shell, administradores de ventanas o escritorios. En este sentido, para el usuario, el sistema operativo funciona más como entorno operativo, que puede ser controlado según el gusto del usuario.

En Linux, los archivos se organizan en directorios, de manera muy similar a Windows. El sistema de archivos completo de Linux es un gran conjunto de directorios interconectados, cada uno con archivos. Algunos directorios son reservados para uso estándar del sistema. Puede crear directorios propios para sus archivos, además de moverlos fácilmente de un directorio a otro.

Incluso puede mover directorios enteros, además de compartir directorios y archivos con otros usuarios de su sistema. Con Linux, también puede configurar permisos en directorios y archivos, permitiendo a otros acceder a ellos o restringir el acceso para que sólo usted pueda acceder a ellos. Los directorios de cada usuario están, en realidad, conectados a los directorios de otros usuarios. Los directorios se organizan en una estructura de árbol jerárquico, empezando por un directorio raíz inicial. Todos los demás directorios derivan al final de cuentas del primer directorio raíz.

Con KDE y GNOME, Linux ahora tiene una GUI completamente integrada. Puede realizar todas sus operaciones en Linux desde cualquiera de estas interfaces. KDE y GNOME son escritorios totalmente operacionales apoyando operaciones de arrastre y colocación, permitiéndole arrastrar iconos al escritorio y configurar sus propios menús en un panel Aplicaciones. Ambos dependen de un sistema X Windows System, esto significa que mientras ambos estén instalados en su sistema, las aplicaciones de una pueden ejecutarse en el otro escritorio. Los sitios de GNOME y KDE son muy útiles para documentación, noticias y software que puede descargar para esos escritorios. Ambos escritorios pueden ejecutar cualquier programa de X Windows System, asimismo cualquier programa basado en cursores como Emacs y Vi, diseñados para trabajar en un entorno shell. Al mismo tiempo, se ha escrito gran cantidad de aplicaciones sólo para esos escritorios, incluidas en sus distribuciones. KDE y GNOME tienen conjuntos completos de herramientas de Internet, junto con editores de imágenes, multimedia y aplicaciones del sistema. Revise los sitios Web en gnome.org y kde.org para conocer los últimos desarrollos. A medida que se publican nuevas versiones, incluyen nuevo software.

Software de fuente abierta

Linux fue desarrollado como un esfuerzo conjunto de fuente abierta a través de Internet, así que ninguna compañía o institución controla Linux. El software desarrollado por Linux refleja su trayectoria. El desarrollo suele presentarse cuando los usuarios de Linux deciden trabajar juntos en un proyecto. El software se publica en un sitio de Internet y cualquier usuario de Linux puede acceder a este sitio y descargarlo. El desarrollo del software para Linux siempre ha operado en un entorno de Internet y tiene un alcance global, con programadores de todo el mundo. Lo único que necesita para iniciar un proyecto de software basado en Linux es un sitio Web.

Casi todo el software de Linux se ha desarrollado como software de fuente abierta. Esto significa que el código fuente de una aplicación se distribuye de manera libre junto con la aplicación. En Internet, los programadores pueden hacer sus propias contribuciones al desarrollo de un paquete de software, modificando y corrigiendo el código fuente. Además, Linux es un sistema operativo de fuente abierta. Su código fuente también se incluye en todas sus distribuciones y está disponible de manera gratuita en Internet. Muchos esfuerzos importantes de desarrollo de software son también proyectos de fuente abierta, como los escritorios KDE y GNOME, junto con la mayoría de aplicaciones. El paquete del explorador Web Netscape Communicator también se ha vuelto de fuente abierta, con su código fuente disponible de manera gratuita. La suite de oficina OpenOffice, a la que Sun da soporte técnico, es un proyecto de fuente abierta basado en el paquete de oficina StarOffice (StarOffice es, en esencia, la versión comercial de OpenOffice de Sun). Muchas aplicaciones de fuente abierta en ejecución para Linux han ubicado sus sitios Web en SourceForge (sourceforge.net), un sitio host diseñado específicamente para apoyar proyectos de fuente abierta. Encontrará más información acerca del movimiento de fuente abierta en opensource.org.

El software de fuente abierta está protegido por licencias públicas. Esto evita que empresas comerciales tomen control del software de fuente abierta, agregando unas cuantas modificaciones propias, registrar en derechos de autor esos cambios y vender el software como su producto. La licencia pública más popular es GNU GPL, proporcionada por Free Software Foundation. Es la licencia bajo la que se distribuye Linux. GNU GPL retiene los derechos de autor y otorga licencias

gratuitas con los requisitos de que el software y cualquier modificación hecha, siempre estén disponibles de manera gratuita. También se han creado licencias públicas para dar soporte a demandas de diferentes tipos de proyectos de fuente abierta. La licencia pública general menor, GNU (LGPL), permite que aplicaciones comerciales usen bibliotecas de software con licencia de GNU. La licencia pública qt (QPL) permite a los desarrolladores de fuente abierta usar bibliotecas esenciales Qt para el escritorio KDE. Encontrará una lista completa en opensource.org.

Linux se encuentra registrado en derechos de autor bajo la licencia pública GNU proporcionada por Free Software Foundation, a menudo conocida como software GNU (consulte gnu.org). El software GNU se distribuye de manera gratuita, con la condición de que se distribuya a otros de esta manera. Se ha probado que el software GNU es confiable y eficaz. Muchas utilerías populares de Linux, como compiladores C, shells y editores, son aplicaciones de software GNU. Instalados con la distribución Linux se encuentran los compiladores GNU C++ y Lisp, los editores Vi y Emacs, las shells BASH y TCSH, además de formadores de documentos TeX y Ghostscript. También existen muchos proyectos de software de fuente abierta con licencia bajo GNU GPL.

De acuerdo con los términos de GNU GPL, el autor original mantiene derechos de autor, aunque cualquiera puede modificar el software y redistribuirlo, siempre y cuando el código fuente esté incluido, se haga público y sea gratuito. Tampoco existen restricciones para vender el software o darlo gratis. Un distribuidor puede cobrar por el software, mientras otro puede no hacerlo. Las principales compañías de software también proporcionan versiones de Linux de sus aplicaciones más populares. Oracle proporciona una versión para Linux de su base de datos Oracle. (A la fecha, no parece haber planes para aplicaciones de Microsoft.)

Software de Linux

Todo el software para Linux se encuentra disponible en depósitos en línea. Es posible descargar las aplicaciones para escritorios, servidores de Internet, suites de oficina y paquetes de programación, entre otras. Los paquetes de software pueden distribuirse a través de depósitos en línea.

Administrador y actualizador de su software de escritorio manejan automáticamente descargas y actualizaciones.

Además, puede descargar software de terceros en forma de archivos comprimidos o paquetes de software como RPM y DEB. Los paquetes RPM se guardan en archivos usando Red Hat Package Manager, usado en varias distribuciones. Los archivos comprimidos tienen extensiones como **.tar.gz** o **tar.Z**, mientras los paquetes RPM tienen una extensión **.rpm**, DEB utiliza extensiones **.deb**.

Cualquier paquete RPM descargado directamente desde cualquier sitio, puede instalarse de manera sencilla con un clic, usando un administrador de software de distribución en un escritorio. También puede descargar la versión fuente y compilarla directamente en su sistema. Esto se ha vuelto un proceso simple, casi tan sencillo como instalar versiones compiladas RPM.

Las distribuciones Linux también tienen gran número de sitios espejo para descargar paquetes de software de versiones actuales. Si tiene problemas para conectarse a un sitio principal FTP, trate con uno de los espejos.

Depósitos de software

En el caso de muchas distribuciones, puede actualizar al software más reciente de depósitos en línea, empleando un actualizador de software. Las distribuciones de Linux ofrecen una selección muy completa de software, que va desde aplicaciones de oficina o multimedia hasta servidores de Internet y servicios administrativos. Muchas aplicaciones populares no están incluidas, aunque tal vez se proporcionen en sitios de software asociados. Durante la instalación, su instalador de software está configurado para acceder al depósito de distribución.

Debido a restricciones de licencia, el soporte multimedia para formatos populares como MP3, DVD y DivX no se incluye en las distribuciones. Sin embargo, es posible que un sitio de distribución asociado facilite soporte y desde allí pueda descargar software para MP3, DVD y DivX.

Por ejemplo, es posible usar un complemento gstreamer MP3 de licencia gratuita desde **fluendo.com**. Muchas distribuciones no soportan controladores gráficos oficiales de Nvidia o ATI, pero puede encontrarlos en sitios de distribución asociados. Las distribuciones de Linux incluyen controladores genéricos X.org de Nvidia y ATI, que permitirán funcionar a sus tarjetas gráficas.

Depósitos de software para Linux de terceros

A pesar de que casi todas las aplicaciones deben incluirse en los depósitos de distribución de software, puede descargar e instalar software desde depósitos de terceros. Siempre revise primero si el software de su interés se encuentra en depósitos de distribución. Si no está disponible, entonces descárguelo de otro depósito de terceros.

Varios depósitos de terceros facilitan ubicar y buscar aplicaciones además de información sobre éstas. En particular **sourceforge.net**, **rpmfind.net**, **gnomefiles.org** y **kde-apps.org**. En las siguientes tablas se presenta una lista de diferentes sitios de software de Linux. Algunos depósitos de terceros y archivos para software de Linux se encuentran en la lista de la tabla 1-2, así como varios sitios especializados, de software comercial y juegos. Cuando descargue paquetes de software, siempre revise si existen versiones en paquetes para su distribución en particular.

Software de oficina y bases de datos para Linux

Muchas bases de datos y paquetes de oficina profesionales están disponibles para Linux. Entre éstos se incluyen las bases de datos de Oracle e IBM, además de las suites OpenOffice y KOffice. En la tabla 1-3 se muestra una lista de sitios para suites de oficina y bases de datos. Muchos paquetes de oficina, además de MySQL y PostgreSQL, se incluyen en los depósitos de distribución y pueden ser parte de su disco de instalación. Muchos sitios proporcionan versiones “confeccionadas” de su software para Linux y otros son totalmente gratuitos. Puede descargarlos directamente de los sitios e instalar el software en su sistema Linux.

URL	Descripción del sitio
sourceforge.net	Lista sitios de desarrollo de software de fuentes abierta, para aplicaciones de Linux y depósitos de software
jpackage.org	Depósitos para aplicaciones y herramientas de Java
gnomefiles.org	Aplicaciones GNOME
kde-apps.org	Depósitos de software KDE
freshmeat.net	Nuevo software Linux
rpmfind.net	Depósitos de paquetes RPM
gnu.org	Archivo GNU
happypenguin.org	Linux Game Tome
linuxgames.com	Juegos para Linux
fluendo.com	Codecs con licencia para multimedia Gstreamer (GNOME) y complementos (MP3, MPEG2, etc.)

TABLA 1-2 Archivos, depósitos y vínculos de software de terceros para Linux

URL	Software de base de datos
Database Software	
oracle.com	Oracle
sybase.com	Sybase
software.ibm.com/data/db2/linux	IBM DB2
mysql.com	MySQL
ispras.ru/~kml/gss	GNU SQL
postgresql.org	PostgreSQL
Software de oficina	
openoffice.org	OpenOffice
koffice.kde.org	KOffice
sun.com/software/star/staroffice	StarOffice
gnomefiles.org	Aplicaciones de oficina y productividad GNOME

TABLA 1-3 Software de bases de datos y de oficina

Servidores de Internet

Una de las características más importantes de Linux, al igual que todos los sistemas Unix, es su conjunto de clientes y servidores para Internet. Internet fue diseñado y desarrollado en sistemas Unix, al igual que clientes y servidores de Internet, como FTP y Web, implementados primero en versiones BSD de Unix. DARPANET, el precursor de Internet, se configuró para vincular sistemas Unix en diferentes universidades de Estados Unidos. Linux tiene un conjunto completo de clientes y servidores de Internet, incluidos correo electrónico, noticias, FTP y Web, además de clientes y servidores proxy. En la tabla 1-4 hay una lista de sitios para software de servidores de red y seguridad disponibles para Linux. Todos ellos se encuentran incluidos en la mayoría de los

URL	Descripción del software
apache.org	Servidor Web Apache
vsftpd.beasts.org	Un servidor FTP muy seguro
proftpd.org	Servidor FTP ProFTPd
isc.org	Consorcio de Software de Internet: BIND, INN y DHCPD
sendmail.org	Servidor de correo electrónico Sendmail
postfix.org	Servidor de correo electrónico Postfix
squid-cache.org	Servidor Squid proxy
samba.org	Servidor Samba SMB (red Windows)
netfilter.org	Firewall de tablas de IP
web.mit.edu/kerberos/www	Protocolo de autenticación de red Kerberos
openssh.com	Open Secure Shell (versión gratuita de SSH)

TABLA 1-4 Software de servidores y seguridad de redes

URL	Descripción del sitio
gnu.org	Compiladores y herramientas de Linux (gcc)
java.sun.com	Sitio Web de Sun Java
perl.com	Sitio Web de Perl y software Perl para Linux
developer.gnome.org	Sitio Web para desarrolladores GNOME
developer.kde.org	Librería para desarrolladores KDE

TABLA 1-5 Sitios de programación Linux

depósitos de distribución y pueden ser parte de su disco de instalación; sin embargo, puede obtener noticias y documentación directamente desde el sitio Web del servidor.

Recursos para desarrollo

Linux siempre ha ofrecido gran soporte para lenguajes y herramientas de programación. Todas las distribuciones incluyen el compilador GNU C y C++ (gcc) con herramientas de soporte como make. Las distribuciones de Linux frecuentemente incluyen soporte íntegro para desarrollo en los escritorios KDE y GNOME, permitiéndole crear sus propias aplicaciones GNOME y KDE. También puede descargar la versión para Linux de Java Software Development Kit, orientado a la creación de programas en Java. Una versión de Perl para Linux también se incluye con casi todas las distribuciones. Puede descargar versiones actuales desde sus sitios Web. En la tabla 1-5 se muestran diferentes sitios de interés para programación en Linux.

Fuentes de información en línea relacionadas con Linux

Existen extensos recursos en línea sobre casi cualquier tema de Linux. En las tablas de este capítulo se muestran sitios donde puede obtener software, desplegar documentación y leer artículos de los desarrollos más recientes. Muchos sitios Web relacionados con Linux proporcionan noticias, artículos e información acerca de Linux. Varios, como linuxjournal.com, se basan en revistas populares de Linux. Algunos se especializan en áreas particulares, como linuxgames.com, para dar a conocer los juegos más recientes migrados a Linux. Actualmente, muchos sitios Web de Linux facilitan el acceso a noticias, información y artículos sobre los desarrollos de Linux, además de documentación, vínculos para software y otros recursos. Estos se muestran en la lista de la tabla 1-6.

Documentación de Linux

La documentación de Linux también se ha desarrollado a través de Internet. Mucha de la documentación disponible para Linux puede descargarse de sitios FTP de Internet. Un proyecto especial denominado Linux Documentation Project (LDP), encabezado por Matt Welsh, ha implementado un conjunto completo de manuales para Linux. La documentación está disponible en la página de inicio del sitio LDP, tldp.org. Los documentos de Linux proporcionados por LDP se encuentran en la lista de la tabla 1-7, junto con sus sitios de Internet. La documentación de Linux para el software instalado estará disponible en el directorio `/usr/share/doc`.

Un vasto número de espejos se mantienen para LDP. Puede ir a cualquier vínculo de estos sitios desde diversas fuentes, como la página de inicio del sitio LDP, tldp.org y linuxjournal.org. La documentación incluye guía de usuario, introducción y guías administrativas.

URL	Descripción del sitio
tldp.org	The Linux Documentation Project
lwn.net	Linux Weekly News
linux.com	Linux.com
linuxtoday.com	Linux Today
linuxplanet.com	LinuxPlanet
linuxfocus.org	Linux Focus
linuxjournal.com	Linux Journal
linuxgazette.com	Linux Gazette
linux.org	Linux Online
slashdot.org	Foro sobre Linux
opensource.org	Información sobre fuente abierta

TABLA 1-6 Información de Linux y sitios de noticias

Esta documentación está disponible en formatos de texto, PostScript o páginas Web. También puede encontrar explicaciones más breves en lo que se conoce como documentos HOW-TO.

Los sitios Web de la distribución correspondiente contienen grandes cantidades de documentación y software. El sitio **gnome.org** aloja información para el escritorio GNOME, mientras **kde.org** documenta el escritorio KDE. Las tablas de este capítulo muestran una lista de sitios disponibles. Encontrará otros a través de páginas de recursos almacenando vínculos a sitios Web (por ejemplo, el sitio Web de Linux en World Wide Web en **tldp.org/links.html**).

Sitios	Sitios Web
tldp.org	Sitio Web LDP en español
Guías	Formato del documento
<i>Linux Installation and Getting Started Guide</i>	DVI, PostScript, LaTeX, PDF y HTML
<i>Guía del Usuario de Linux</i>	DVI, PostScript, HTML, LaTeX y PDF
<i>Guía para administradores de sistemas GNU/Linux</i>	PostScript, PDF, LaTeX y HTML
<i>Guía de administración de redes</i>	DVI, PostScript, PDF y HTML
<i>Guía Linux de programación</i>	DVI, PostScript, PDF, LaTeX y HTML
<i>Guía del núcleo</i>	HTML, LaTeX, DVI y PostScript
<i>Linux Kernel Hacker's Guide</i>	DVI, PostScript y HTML
<i>Linux HOW-TOs</i>	HTML, PostScript, SGML y DVI
<i>Linux FAQs</i>	HTML, PostScript y DVI
<i>Linux Man Pages</i>	Página de Man

TABLA 1-7 Proyecto de documentación de Linux

Además de los sitios Web, también hay disponibles grupos de noticias Usenet relacionados con Linux. Mediante su conexión a Internet, puede acceder a grupos de noticias de Linux para leer comentarios de otros usuarios y publicar mensajes propios. Existen varios grupos de noticias de Linux; todos ellos comienzan con **comp.os.linux**. Un grupo de noticias de particular interés para principiantes es **comp.os.linux.help**, donde puede publicar preguntas. En la tabla 1-8 aparece una lista de grupos de noticias Usenet de Linux que puede revisar, sobre todo para publicar preguntas.

Grupos de noticias	Descripción
comp.os.linux.announce	Anuncio de desarrollos de Linux
comp.os.linux.development.apps	Para programadores que desarrollan aplicaciones Linux
comp.os.linux.development.system	Para programadores que trabajan en el sistema operativo Linux
comp.os.linux.hardware	Para especificaciones de hardware relacionadas con Linux
comp.os.linux.admin	Preguntas de administración de sistema
comp.os.linux.misc	Preguntas y problemas especiales
comp.os.linux.setup	Problemas de instalación
comp.os.linux.answers	Respuestas a problemas de comando
comp.os.linux.help	Preguntas y respuestas para problemas en particular
comp.os.linux.networking	Preguntas y problemas de red Linux
linux.dev.group	Numerosos grupos de noticias sobre desarrollo que inician con linux.dev , como linux.dev.admin y linux.dev.doc

TABLA 1-8 Grupos de noticias Usenet de Linux

Primeros pasos

El uso de Linux se ha vuelto un proceso intuitivo, con una interfaz de fácil uso, incluidos inicios de sesión gráficos e interfaces gráficas de usuario (GUI, Graphical User Interfaces) como GNOME y KDE. Incluso la interfaz de línea de comando estándar de Linux se ha vuelto más amigable para el usuario, con comandos permitiendo edición, listas de historial y herramientas basadas en cursor. Las herramientas de instalación de las distribuciones también utilizan GUIs simples. Instalar se ha vuelto un procedimiento muy sencillo, que sólo toma unos minutos. El uso de depósitos en línea por parte de muchas distribuciones permite instalaciones iniciales que luego pueden mejorarse con software adicional seleccionado.

Para comenzar a usar Linux, necesita saber cómo acceder a su sistema Linux y, una vez dentro de él, cómo ejecutar comandos y aplicaciones. El acceso tiene soporte mediante el inicio de sesión gráfico predeterminado o un inicio de sesión por línea de comandos. En el caso de un inicio de sesión gráfico, aparecerá una ventana sencilla con menús para seleccionar las opciones de inicio de sesión y cuadros de texto para insertar el nombre de usuario y contraseña. Una vez que se tiene acceso al sistema, puede interactuarse con éste usando la interfaz de línea de comandos o GUI. Con las interfaces GUI como GNOME y KDE, es posible utilizar ventanas, menús e iconos para interactuar con su sistema.

Linux es conocido porque proporciona un acceso sencillo a extensa documentación de ayuda. Es fácil obtener información rápida acerca de cualquier comando y utilidad de Linux, mientras esté en sesión del sistema. Tiene la opción de acceder al manual en línea describiendo cada comando u obtener ayuda para explicaciones más detalladas acerca de diferentes características de Linux. Un conjunto completo de manuales proporcionados por Linux Documentation Project (LPD) están disponibles en su sistema para explorarlos o imprimirlos. Ambos escritorios, GNOME y KDE, ofrecen sistemas de ayuda para un acceso sencillo al escritorio, sistema y archivos de ayuda de la aplicación.

Problemas de instalación

Cada distribución tiene su propia herramienta de instalación gráfica con que instalar Linux de manera muy sencilla. A menudo, para instalar sólo necesita hacer clic en varios botones. Sin embargo, CD y DVD de instalación sólo proporcionan un subconjunto básico de lo realmente disponible, pues la cantidad de software ha crecido tanto que casi todas las distribuciones proporcionan depósitos en línea para descargarlos. Ahora sólo se necesita establecer una configuración inicial para instalar, misma que después puede expandirse utilizando tales depósitos en línea. Muchas distribuciones también permiten crear sus propios discos de instalación,

personalizando la colección de software que quiere en su CD o DVD de instalación. Entre otras consideraciones relacionadas con la instalación se incluyen las siguientes:

- Muchas distribuciones proporcionan LiveCD para instalaciones mínimas. Esto ayuda a evitar la descarga de un CD o DVD demasiado grande. Luego puede instalar aquellos paquetes de su interés desde depósitos en línea.
- El uso de depósitos en línea significa que casi todo el software instalado necesita descargarse o actualizarse desde depósitos tras la instalación. El software de CD y DVD de instalación pierde actualidad con rapidez.
- Algunas distribuciones proporcionan versiones actualizadas de una versión, incluyendo software actualizado desde el lanzamiento original. Estos, a menudo se proporcionan por proyectos de distribución separados. Revise los sitios de distribución para saber si hay disponibilidad.
- Casi todo su hardware se detecta automáticamente, incluidos tarjeta gráfica y monitor.
- La mayor parte de distribuciones utilizan Parted para configurar sus particiones. Parted es una herramienta de administración de particiones muy fácil de usar.
- La instalación puede realizarse desde varias fuentes, al utilizar métodos de red como NFS, protocolo de transferencia de archivos (FTP, File Transfer Protocol) y protocolo de transferencia de hipertexto (HTTP, HyperText Transfer Protocol).
- Se da soporte a instalaciones de arranque dual con los administradores de arranque GRUB o Linux Loader (LILO). Los administradores de arranque de Linux pueden configurarse de manera sencilla para iniciar con Windows, Mac u otras instalaciones de Linux presentes en el mismo sistema.
- Las distribuciones se diferencian entre lanzamientos de 32 ó 64 bits. Casi todas las CPU en los equipos nuevos soportan 64 bits, mientras sistemas antiguos o menos potentes tal vez no los soporten.
- En general, la configuración de red es automática, mediante el protocolo de configuración de host dinámico (DHCP, Dynamic Host Configuration Protocol) o IPv6 para conectarse a un enrutador de red.
- Durante la instalación, puede optar por la personalización de sus particiones, permitiéndole configurar un sistema de archivos RAID y LVM, si lo desea.
- Si está utilizando un sistema de archivos LVM o RAID, asegúrese de tener una partición de arranque perteneciente a un tipo de sistema de archivos estándar de Linux.
- Casi todas las distribuciones realizan un procedimiento tras la instalación enfocada en tareas de configuración básicas, como establecer la fecha y hora, configurar su firewall y crear una cuenta de usuario (una cuenta [administrativa] de arranque se configura durante la instalación).

La mayor parte de las distribuciones proporcionan medios para acceder a su sistema Linux en modo de rescate. En caso de que su sistema deje de funcionar, puede acceder a sus archivos usando el disco de instalación para iniciar Linux con una interfaz de línea de comando y acceder a su sistema de archivos instalado. Esto permite corregir su problema al editar o reemplazar archivos de configuración (`/etc/X11/xorg.conf` es útil para problemas con X Windows System).

Si tiene problemas con el cargador de arranque GRUB, puede reinstalarlo con el comando **grub-install**. Esto llega a pasar si instala después Windows en su sistema. Windows sobrescribirá el administrador de arranque. Utilice **grub-install** con el nombre del dispositivo del disco duro para reinstalar el administrador de arranque de Linux. Asegúrese de poner una entrada para su sistema Windows. Tenga en cuenta que algunas distribuciones usan cargadores de arranque alternos como LILO.

Accediendo a su sistema Linux

Para acceder a su sistema Linux y utilizarlo, necesita seguir con cuidado los procesos necesarios de arranque y apagado. No sólo debe apagar su equipo. Sin embargo, Linux implementa una opción para recuperar su sistema automáticamente en caso de que su computadora se quede sin energía eléctrica y apague repentinamente.

Si ha instalado el cargador de arranque GRUB, cuando enciende o restaura su computadora, el cargador de arranque primero decide qué sistema operativo cargar y ejecutar. GRUB desplegará un menú de sistemas operativos, entre los que podrá escoger uno.

Si en vez de eso espera un momento o presiona **ENTER**, el cargador de arranque cargará el sistema operativo predeterminado. En caso de encontrar en la lista un sistema Windows, puede elegir que se inicie éste.

Considere que su sistema operativo Linux actúa en dos niveles diferentes y uno se ejecuta encima del otro. El primer nivel es cuando inicia su sistema Linux y donde el sistema se carga y ejecuta. Tiene control de su equipo y todos sus periféricos. Sin embargo, todavía no puede interactuar con el sistema. Tras iniciar Linux, despliega una pantalla de inicio de sesión, donde espera a que un usuario inicie sesión en el sistema y comience a utilizarlo. No puede acceder a Linux mientras no inicie sesión primero.

Considere que iniciar sesión y usar Linux es el siguiente nivel. Ahora puede utilizar comandos para instruir a Linux que realice tareas. Puede emplear utilidades y programas como editores y compiladores, incluso juegos. Sin embargo, dependiendo de la elección tomada durante la instalación, puede interactuar con su sistema empleando la interfaz de línea de comandos simple, o directamente el escritorio. Existen tanto indicadores de inicio de sesión en línea de comandos como ventanas de inicio de sesión gráfica. Casi todas las distribuciones utilizarán una interfaz gráfica como opción predeterminada, presentándole una ventana gráfica de inicio de sesión en la que puede ingresar nombre de usuario y contraseña. Si decide no recurrir a la interfaz gráfica, se le presentará un indicador de línea de comando simple para insertar su nombre de usuario.

Los administradores de pantalla: GDM y KDM

Con el inicio de sesión gráfico, su GUI se activa de inmediato, desplegando una ventana de inicio de sesión con cuadros para el nombre de usuario y contraseña. Al insertar su nombre de usuario y contraseña, para oprimir enter, iniciará su GUI predeterminada.

En casi todas las distribuciones, los inicios de sesión gráficos son administrados por GNOME Display Manager (GDM) o KDE Display Manager (KDM). GDM y KDM administran la interfaz de inicio de sesión además de la autenticación del nombre de usuario y contraseña del mismo; luego el inicio del escritorio seleccionado. Si ocurren problemas por uso de la GUI, puede forzar la salida de ésta con el comando de teclado **CTRL-ALT-RETROCESO**, que lo devolverá a la ventana de inicio de sesión (o la línea de comandos, si inició su GUI desde ahí). También, desde el administrador de pantalla, puede cambiar a la interfaz de línea de comandos con las teclas **CTRL-ALT-F1** y regresar a la GUI con las teclas **CTRL-ALT-F7**.

NOTA DEL TRADUCTOR Los nombres de los comandos en español pueden variar, dependiendo de la versión usada.

Cuando cierra su sesión desde el escritorio, regresará a la ventana de inicio de sesión del administrador de pantalla. Desde el menú Opciones, puede seleccionar escritorio o administrador de ventanas que quiere iniciar. Aquí selecciona entre KDE para iniciar K Desktop, por ejemplo, en vez de GNOME. El menú Idioma presenta una lista con variedad de idiomas diferentes soportados por Linux. Elija uno para cambiar el idioma de la interfaz.

Para apagar su sistema Linux, haga clic en el botón Apagar. Para reiniciar, seleccione la opción Reiniciar del menú Opciones. Como alternativa, puede apagar o reiniciar desde su escritorio. Desde el menú Sistema, seleccione la entrada Apagar. GNOME desplegará una pantalla de diálogo con los botones Suspende, Apagar y Reiniciar. Apagar es la opción predeterminada y ocurrirá automáticamente tras algunos segundos.

Si selecciona Reiniciar se apagará y reiniciará su sistema. KDE preguntará si quiere terminar una sesión, apagar o cerrar sesión. (También tiene la opción de abrir una ventana de terminal e ingresar el comando **shutdown**, **halt** o **reboot**, como se describe después; **halt** cerrará la sesión y apagará su sistema.)

Cambio de usuarios

Una vez haya iniciado sesión en su escritorio, puede cambiar a un usuario diferente sin salir de su sesión y terminar su sesión de usuario actual. En GNOME use la herramienta Selector de usuarios, applet de GNOME en el panel. Para KDE use la entrada Cambiar usuario en el menú Principal.

Selector de usuarios: GNOME

En GNOME, el selector se mostrará en el panel con el nombre del usuario que inició sesión. Si hace clic en el nombre, se desplegará una lista de todos los usuarios. Las casillas de verificación a un lado de los nombres mostrarán qué usuarios están conectados y en ejecución. Para cambiar a un usuario, seleccione uno desde este menú. Si todavía no ha iniciado sesión, aparecerá el administrador de inicio de sesión (GDM) y podrá ingresar una contraseña de usuario. Si el usuario ha iniciado sesión, aparecerá la ventana Inicio de sesión de la pantalla de bloqueo (puede deshabilitar la pantalla de bloqueo). Sólo escriba la contraseña de usuario. La sesión original del usuario continuará con la misma ventana abierta y las aplicaciones en ejecución, tal y como cuando cambió de usuario. Puede cambiar de manera sencilla entre usuarios que iniciaron sesión, reteniendo la sesión con todos los usuarios como la dejaron la última vez que ingresaron. Cuando cambia de un usuario a otro, los programas de ese usuario continuarán ejecutándose en el fondo.

Al hacer clic con el botón derecho en el selector, se desplegará una lista con elementos de administración de usuario, como configuración de la pantalla de inicio de sesión, administración de usuarios o cambio de la contraseña de usuario e información personal. El elemento Preferencias permite configurar cómo se despliega el Selector de usuarios en su panel. En vez del nombre de usuario, puede usar el término Usuarios o un icono de usuario. También puede elegir si quiere utilizar el bloqueo de pantalla cuando cambia de usuario. Al deshabilitar la opción bloqueo de pantalla podrá cambiar discretamente entre usuarios que iniciaron sesión.

Cambiar usuario: KDE

En KDE, la entrada Cambiar usuario en el Menú principal desplegará una lista de usuarios entre los que puede cambiar. También es posible elegir un inicio de sesión diferente, escondiendo su sesión actual. En efecto, esto permite iniciar de nuevo su escritorio como el mismo usuario. También puede bloquear su sesión actual antes de iniciar una nueva. Puede hacer referencia a nuevas sesiones a partir de la tecla F7, la primera sesión. Use CTRL-ALT-F7 para acceder a la primera sesión y CTRL-ALT-F8 para la segunda sesión.

Acceso a Linux desde la interfaz de línea de comandos

En el caso de la interfaz de línea de comandos, al principio se le presenta un indicador de comandos de inicio de sesión. El sistema se está ejecutando ahora y esperando que un usuario inicie sesión y la use. Puede insertar su nombre de usuario y contraseña para usar el sistema. El indicador de comandos de inicio de sesión es precedido por el nombre de host que dio a su sistema. En este

ejemplo, el nombre de host es **tortuga**. Cuando termine de utilizar Linux, primero deberá cerrar su sesión. Después Linux despliega exactamente el mismo indicador de comandos de inicio de sesión, esperando que otro usuario vuelva a iniciar sesión. Esto es equivalente a la ventana de inicio de sesión proporcionada por GDM. Luego podrá iniciar sesión en otra cuenta.

```
Linux release
Kernel 2.6 on an i686
```

```
tortuga login:
```

Inicio y cierre de sesión con la línea de comandos

Una vez haya iniciado sesión en una cuenta, puede ingresar y ejecutar comandos. Para iniciar sesión en la cuenta Linux se dan dos pasos: insertar el nombre de usuario y después ingresar la contraseña. Escriba el nombre de usuario de su cuenta. Si se equivoca, puede borrar caracteres con la tecla **RETROCESO**. En el siguiente ejemplo, el usuario escribe el nombre de usuario **richlp** y luego se le pide inserte su contraseña:

```
Linux release
Kernel 2.6 on an i686

tortuga login: oscludo
Password:
```

Cuando escribe su contraseña, no aparece en la pantalla. Esto es para evitar que sea vista por otras personas. Si inserta el nombre de usuario o contraseña equivocados, el sistema responderá con un mensaje de error “Login incorrect” y pedirá de nuevo el nombre de usuario, comenzando el proceso de inicio de sesión otra vez. Tras ello puede volver a insertar su nombre de usuario y contraseña.

Una vez introduzca su nombre de usuario y contraseña correctamente, habrá iniciado sesión en el sistema. Se despliega un indicador para su línea de comandos, esperando que escriba un comando. Observe que el indicador de línea de comando es un signo de pesos (\$), no de número (#). El signo \$ indica comandos para usuarios regulares, mientras # es sólo para usuarios raíz. En esta versión de Linux, su indicador de comandos es precedido por el nombre de host y directorio en que está. Ambas se encuentran unidas por un conjunto de corchetes.

```
[tortuga /home/oscludo]$
```

Para terminar su sesión, escriba el comando **logout** o **exit**. Esto lo devuelve a la petición de inicio de sesión y Linux espera otro usuario para iniciar sesión:

```
[tortuga /home/oscludo]$ logout
```

Apagado de Linux desde la línea de comando

Si quiere apagar su equipo, primero debe salir de Linux. En caso de no apagar el sistema, Linux requerirá que se realice una revisión exhaustiva del sistema cuando inicie nuevamente. El sistema se apaga de dos formas. Primero inicie sesión con una cuenta y después inserte el comando **halt**. Este comando terminará su sesión y apagará el sistema.

```
$ halt
```

Como opción, puede utilizar el comando **shutdown** con la opción **-h**. O, con la opción **-r**, el sistema se apaga y después reinicia. En el siguiente ejemplo, el sistema se apaga luego de cinco minutos. Para apagar el sistema inmediatamente, puede utilizar **+0** o la palabra **now**.

```
# shutdown -h now
```

SUGERENCIA *Para apagar se requiere una serie de acciones importantes: desmontar los sistemas de archivo y apagar cualquier servidor. Nunca debe apagar simplemente su equipo, aunque se puede recuperar normalmente.*

También puede forzar su sistema para reiniciar en el indicador de comandos de inicio de sesión al mantener oprimida las teclas CTRL y ALT y después SUPR (CTRL-ALT-SUPR). Su sistema hará el procedimiento de apagado estándar y después reiniciará su equipo.

Los escritorios GNOME y KDE

Es posible instalar dos escritorios GUI alternativos en la mayor parte de sistemas Linux: GNOME y KDE. Cada uno con estilo y apariencia propios. GNOME usa el tema Clearlooks para su interfaz con el fondo de pantalla e icono de menú de la distribución como opción predeterminada.

Es importante tener en cuenta que, pese a la semejanza de las interfaces GNOME y KDE, en realidad son dos interfaces de escritorio totalmente diferentes con herramientas separadas para seleccionar preferencias. El menú Preferencias en GNOME y KDE despliegan selecciones muy diferentes de las herramientas de configuración del escritorio.

A pesar de que GNOME y KDE son escritorios totalmente integrados, en realidad interactúan con el sistema mediante un administrador de ventanas —Metacity en el caso de GNOME y el administrador de ventanas KDE para KDE. Puede utilizar un administrador de ventanas diferente, compatible con GNOME o KDE, si así lo desea, o utilizar sólo un administrador de ventanas en lugar de KDE o GNOME. Encontrará información detallada acerca de diferentes administradores de ventanas disponibles para Linux en el sitio Web de X11 en xwinman.org.

KDE

K Desktop Environment (KDE) despliega un panel en la parte inferior de la pantalla, muy similar al que se despliega en la parte superior del escritorio GNOME. El administrador de archivos tiene un aspecto ligeramente distinto, pero opera casi de la misma forma que el de GNOME. Hay una entrada Centro de control en el Menú principal, para abrir el centro de control KDE, desde donde puede configurar cada aspecto de KDE, como temas, paneles, periféricos (impresoras y teclados), incluso la capacidad del administrador de archivos KDE para explorar la Web.

NOTA *En ambos casos, GNOME y KDE, el administrador de archivos es sensible a Internet. Puede utilizarlo para acceder directorios FTP remotos y desplegar o descargar sus archivos, aunque en KDE el administrador de archivos es también un explorador Web completamente funcional.*

XFce4

XFce4 es un escritorio ligero diseñado para ejecutarse de manera rápida, sin la excesiva capacidad de trabajo visto en escritorios llenos de características como KDE y GNOME. Incluye administrador de archivos y panel propios, pero el énfasis está en la modularidad y simplicidad. El escritorio consta de una colección de módulos, incluidos administrador de archivos xffm, panel xfce4-panel y administrador de ventanas xfwm4. Para mantener el enfoque en la simplicidad, su escala pequeña lo hace apropiado para equipos portátiles o sistemas dedicados que no requieren la complejidad encontrada en otros escritorios.

GNOME

El escritorio GNOME muestra tres menús: Aplicaciones, Lugares y Sistema. El menú Lugares accede a ubicaciones utilizadas comúnmente como el directorio home, la carpeta Escritorio para

cualquier archivo en su escritorio y la ventana Equipo, que accede a dispositivos, sistemas de archivos compartidos y todos los directorios en su sistema local. El menú Sistema incluye los menús Preferencias y Administración. El menú Preferencias se usa para configurar sus opciones GNOME, como el tema que quiere usar o el comportamiento de su ratón.

SUGERENCIA Si su escritorio soporta una configuración `xdg-users-dirs`, entonces su directorio home ya tendrá directorios predeterminados, creados para archivos utilizados comúnmente. Estos incluyen, Descargas, Imágenes, Documentos y Videos.

Para mover una ventana, haga clic y arrastre la barra de título. Cada ventana da soporte a los botones Maximizar, Minimizar y Cerrar. Al hacer doble clic en la barra de título se maximizará la ventana. Cada ventana tendrá un botón correspondiente en el panel inferior. Puede usar este botón para minimizar o restaurar la ventana. El escritorio apoya la capacidad para arrastrar y soltar. Puede mover carpetas, iconos y aplicaciones al escritorio u otras ventanas abiertas, del administrador de archivos a otras carpetas. Mover es la operación de arrastre predeterminada (también puede oprimir la tecla MAYÚS mientras arrastra). Para copiar archivos, oprima la tecla CTRL, después haga clic y arrastre antes de soltar el botón del ratón. Para crear un vínculo, mantenga oprimidas las teclas CTRL y mayús mientras arrastra el icono a la ubicación donde quiere el vínculo, como el escritorio.

GNOME ofrece varias herramientas para configurar su escritorio. Éstas se encuentran en una lista del menú Sistema | Preferencias. Las herramientas de preferencias de configuración están organizadas en varios submenús: Personal, Visualización y comportamiento, Internet y red, Hardware y Sistema. Las herramientas que pertenezcan a una categoría se presentan directamente en la lista. Varias herramientas se analizan en este y otros capítulos del libro. El botón Ayuda en cada ventana de preferencia desplegará una descripción detallada y ejemplos. Algunas de las herramientas más importantes se analizan aquí.

La configuración Combinaciones de teclas (Personal | Combinaciones de teclas), permite asignar teclas a ciertas tareas (por ejemplo, asignar las teclas multimedia de un teclado a tareas multimedia: reproducción o poner en pausa). Con la configuración Gestión de archivos (Personal | Gestión de archivos) determina la forma en que archivos y directorios se despliegan, junto con información agregada para mostrar capturas de icono y vistas de listas. La configuración Ventanas (Visualización y comportamiento | Ventanas) es donde habilita características como rotación de ventanas, teclas de desplazamiento y selección de ventanas con el ratón.

Las preferencias de Ratón y Teclado son las herramientas principales para configurar su ratón y teclado (Hardware | Ratón). En las preferencias de Ratón se elige una imagen de ratón, configuración de movimiento y la mano con que lo manejará. La ventana de preferencias Teclado muestra varios paneles para seleccionar el modelo de su teclado (disposición), configurar teclas (opciones de disposición) y retraso de teclas (Teclado), incluso forzar pausas como una medida de precaución de salud.

Applets de GNOME y KDE

Las applets de GNOME son programas pequeños operando desde su panel. Es muy fácil agregar applets. Haga clic con el botón derecho en el panel y seleccione la entrada Agregar. Así se presenta una lista de applets disponibles. Algunos applets útiles son búsqueda en diccionario; informe meteorológico; monitor del sistema, mostrando el uso de la CPU; monitor de frecuencia de la CPU para procesadores inactivos; y Buscar, que rastrea archivos en su sistema, además de los botones Bloquear, Apagar y Salir. Algunos de éstos, incluidos Buscar, Bloquear y Salir, ya se encuentran en el menú Lugares. Puede arrastrarlos directamente del menú al panel para agregar el applet.

Después del explorador Web y los iconos de correo electrónico tiene, de izquierda a derecha: Buscar archivos, búsqueda de diccionario, tomador de notas Tomboy, Monitor de red, Monitor de escalada de CPU, Monitor del sistema, Informe meteorológico, Ojos que siguen su ratón, Selector de usuarios, además de los botones Cerrar sesión, Apagar y Bloquear pantalla.

En KDE, haga clic con el botón derecho en el panel y seleccione Agregar applet al panel. Desde la ventana de applets de KDE, puede seleccionar applets similares, como Monitor de sistema y Mezclador de audio.

Inicio de una GUI desde la línea de comandos

Una vez haya iniciado sesión en el sistema desde la línea de comandos, todavía tiene opción para iniciar una GUI de X Windows System, como GNOME o KDE. En Linux, el comando **startx** inicia un escritorio. El comando **startx** inicia el escritorio GNOME como opción predeterminada. Una vez cierre el escritorio, regresará a la interfaz de línea de comandos, todavía con su sesión.

```
$ startx
```

Operaciones de escritorio

Tal vez quiera aprovechar alguna de las diversas operaciones de escritorio cuando lo configura por vez primera. Entre éstas se incluye seleccionar temas, configurar tamaños de fuente más grandes para monitores de alta resolución, grabar CD/DVD, buscar archivos en su escritorio, usar medios extraíbles como memorias USB y acceder a host remotos.

Temas de escritorio

En GNOME, utilice la herramienta Preferencias de temas para seleccionar o personalizar un tema. Los temas controlan la apariencia de su escritorio. Cuando abre la herramienta Tema, se muestra una lista de los ya instalados. Al principio, se selecciona el tema GNOME. Puede recorrer la lista para seleccionar uno diferente, si así lo desea. Si ha descargado temas adicionales desde sitios como **art.gnome.org**, puede hacer clic en el botón Instalar, para localizarlos e instalarlos. Una vez instalados, los temas adicionales también se desplegarán en la lista de la herramienta Preferencias de temas. Si descargó o instaló un tema o conjunto de iconos del depósito Fedora, se se presentará de inmediato.

El poder real de los temas está en su capacidad para que los usuarios personalicen cualquier tema. Los temas se organizan en tres componentes: controles, bordes de ventana e iconos. Controles cubre la apariencia de ventanas y cuadros de diálogo, además de botones y barras de desplazamiento. Bordes de ventana comprende el despliegue de barras de título, bordes y botones de ventana. Iconos involucra todos los iconos utilizados en el escritorio, ya sea el administrador de archivos, escritorio o panel. Puede mezclar y hacer coincidir componentes desde cualquier tema instalado para crear su propio tema. Incluso puede descargar e instalar componentes separados como conjuntos de iconos, que después puede utilizar en un tema personalizado.

Al hacer clic en el botón Personalizar, se abrirá a la ventana Detalles de tema, con paneles de los diferentes componentes del tema. Los que se encuentren en uso para el tema actual ya estarán seleccionados. En los paneles de control, bordes de ventana e icono, verá una lista de diferentes temas instalados. Un panel adicional Colores permite configurar fondo y color del texto de las ventanas, cuadros de inserción y elementos seleccionados. Después puede mezclar y hacer coincidir diferentes componentes como iconos, estilos de ventana y controles, creando su propio tema personalizado. Tras seleccionar un componente, su escritorio cambia automáticamente, mostrando el aspecto que tendrá.

Una vez creado un tema personalizado, una entrada Tema personalizado aparecerá en la lista de temas. Para guardar un tema personalizado, haga clic en el botón Guardar tema. Esto abrirá un

cuadro de diálogo donde puede insertar el nombre del tema y cualquier nota, además de especificar si también quiere mantener el fondo del tema. Entonces el tema guardado aparece en la lista de temas.

En KDE, abra el Gestor de temas, en el Centro de control de KDE, bajo Aspecto y temas. Seleccione el tema que quiera desde el panel Tema. Se desplegará el tema seleccionado en el panel derecho. Varios botones en la sección Personalizar permiten crear un tema personalizado, seleccionar fondos, iconos, colores, estilos, fuentes e incluso protector de pantalla. Para descargar nuevos temas, haga clic en el vínculo Obtener nuevos temas, en la esquina superior derecha. Esto abre la página Web KDE-look para temas KDE. Deberá descargar temas, extraerlos y luego hacer clic en el botón Instalar tema, tras localizar y seleccionar el archivo **.kth** del tema descargado. Este método sólo funciona para temas en el formato del Administrador de temas, **kth**. Los temas que no cuenten con este formato deberán instalarse de forma manual.

Los temas e iconos de GNOME instalados directamente por el usuario se colocan en los directorios **.themes** e **.icons**, en el directorio home del usuario. Si quiere que estos temas estén disponibles para todos los usuarios, puede moverlos de los directorios **.themes** e **.icons** a los directorios **/usr/share/icons** y **/usr/share/themes**. Asegúrese de iniciar sesión como usuario root. Después necesita cambiar la propiedad de los temas e iconos movidos al usuario root:

```
chown -R root:root /usr/share/themes/nuevotema
```

Los temas de KDE se colocan en el directorio **.kde/share/apps/kthememanager**.

Fuentes

Ahora casi todas las distribuciones utilizan el método fontconfig para administrar las fuentes (**fontconfig.org**). Puede cambiar de manera sencilla los tamaños de fuente, agregar nuevas y configurar sus características, como antialias. GNOME y KDE facilitan herramientas para seleccionar, cambiar tamaño y agregar fuentes.

Cambio del tamaño de las fuentes del escritorio

A medida que los monitores grandes con altas resoluciones son cada vez más comunes, una característica que los usuarios encuentran útil es la capacidad para incrementar el tamaño de las fuentes del escritorio. En un monitor de pantalla panorámica grande, las resoluciones menores a las nativas tienden a escalar mal. Un monitor siempre luce mejor en resolución nativa. Sin embargo, con una resolución nativa alta como 1900 x 1200, el tamaño del texto se vuelve tan pequeño resulta difícil leerlo. Puede solucionar este problema incrementando el tamaño de la fuente. Use las herramientas de fuente en su escritorio para cambiar dichos tamaños (Sistema | Preferencias | Visualización y comportamiento | Tipografías en GNOME; para KDE, seleccione la entrada Tipos de letra, en el Centro de control Aspecto y temas).

Adición de fuentes

Para agregar una nueva fuente (GNOME y KDE), sólo inserte el URL **fonts:/** en una ventana de administrador de archivos (Abrir ubicación, en el menú Archivo de GNOME). Esto abre la ventana de fuentes. Arrastre y coloque su archivo de fuente en ésta. Cuando reinicie, su fuente estará disponible para usarse en su escritorio. KDE tendrá las carpetas Personal y Sistema para fuentes, mostrando iconos para cada una. En el caso de fuentes de usuario, abra la ventana Fuentes personales. Las fuentes en un archivo Zip, deben abrirse primero con el administrador de archivos y después podrán arrastrarse del administrador al visor de fuentes. Para quitar una fuente, haga clic con el botón derecho en el visor de fuente y seleccione Mover a la papelera o Eliminar.

Las fuentes de usuario se instalarán en el directorio **.fonts** de un usuario. Para que las fuentes estén disponibles para todos los usuarios, deben instalarse en el directorio **/usr/share/fonts**,

haciéndolas fuentes del sistema. En KDE, se hace esto al abrir la carpeta Sistema, en vez de la carpeta Personal, cuando inicia el visor de fuentes. Puede hacer esto desde cualquier inicio de sesión de usuario. Después arrastre cualquier paquete de fuentes a esta ventana **fonts/System**.

En GNOME, necesita iniciar sesión como usuario raíz y copiar manualmente las fuentes al directorio **/usr/share/fonts**. Si su sistema tiene instalado GNOME y KDE, puede instalar fuentes del sistema al usar KDE (administrador de archivos Konqueror), y también estarán disponibles para GNOME.

Para proporcionar acceso rápido a las fuentes del sistema, debe crear archivos de caché de información de las fuentes para el directorio **/usr/share/fonts**. Para ello, ejecute el comando **fc-cache** como usuario root.

Configuración de fuentes

En GNOME, para afinar el despliegue de sus fuentes, puede usar la herramienta de representación. Abra la herramienta Preferencias de fuente (Sistema | Visualización y comportamiento | Apariencia | Tipografías). En la sección Renderizado de fuentes, hay características básicas de representación de fuentes como Monocromo, Mejor contraste, Mejores formas y Suavizado de subpíxel. Elija la que funcione mejor. En el caso de LCDs, elija Suavizado de subpíxel. Para una configuración detallada, haga clic en el botón Detalles. Aquí puede configurar las características suavizado, contorno (antialias) y orden del subpíxel. Esta última opción dependerá del hardware. En KDE, en el Centro de control, seleccione la entrada Tipos de letras, en Aspecto y temas. Ponga una marca en la casilla de verificación Usar antialias para fuentes y después haga clic en el botón Configurar, para abrir una ventana que le permitirá seleccionar opciones de antialias y subpíxel.

En GNOME, al hacer clic en una entrada de fuente en la herramienta Preferencias de tipografías se abrirá un cuadro de diálogo Escoja una tipografía, mostrando una lista de fuentes disponibles. En KDE, al hacer clic en cualquiera de los botones Seleccionar, en el panel Tipo de letras del Centro de control, también se abrirá una ventana presentando las fuentes disponibles. También puede generar una lista con el comando **fc-list**. La lista se mostrará desordenada, así que debe canalizarla primero al comando **sort**. Puede utilizar **fc-list** con cualquier nombre de fuente o patrón de nombre para buscar fuentes, con opciones para buscar por idioma, familia o estilo. Consulte la documentación de **/etc/share/fontconfig** para conocer más detalles.

```
fc-list | sort
```

SUGERENCIA Las fuentes Web comunes de Microsoft están disponibles gratuitamente en **fontconfig.org**. Estas fuentes se encuentran archivadas en el formato cab de Microsoft. Deberá descargar e instalar la herramienta **cabextract** (disponible en muchas colecciones y depósitos de distribución de software) para extraer las fuentes. Una vez extraídas, puede copiarlas a una carpeta en el directorio **/usr/share/fonts** para estar disponibles para todos los usuarios. Si tiene acceso a un sistema Windows, también puede copiar las fuentes directamente desde el directorio de fuentes de Windows a su directorio **/usr/share/fonts**.

Configuración de su información personal

En GNOME, el cuadro de diálogo de preferencias Acerca de mí permite configurar información personal para ser usada en sus aplicaciones de escritorio, además de cambiar su contraseña. Al hacer clic en el icono Imagen, en la esquina superior izquierda, se abrirá una ventana del explorador donde puede seleccionar la imagen que se usará. El directorio Faces se selecciona como opción predeterminada, con imágenes que puede utilizar. La imagen seleccionada se despliega a la derecha de la ventana del explorador. Para ver una fotografía personal, puede usar la carpeta Imágenes, desde su directorio home. Debe colocar una imagen o fotografía allí, después podrá seleccionarla

como su imagen personal. La imagen se utilizará en la pantalla de Inicio de sesión, cuando se presenta la entrada de su usuario. Si desea cambiar su contraseña, puede hacer clic en el botón Cambiar contraseña, en la esquina superior derecha.

Existen tres paneles: Contacto, Dirección y Datos personales. En el panel Contacto, puede ingresar direcciones de correo electrónico (casa y trabajo), teléfono y direcciones de mensajes instantáneos. En el panel Dirección puede introducir direcciones del trabajo y casa, mientras en el panel Datos personales puede hacer una lista de sus direcciones Web e información del trabajo.

En KDE, puede seleccionar el panel Contraseña, en la entrada Seguridad, en el Centro de control de KDE. Aquí puede seleccionar una imagen para su cuenta. La información de contacto es controlada por otras aplicaciones, como Kontact para el correo electrónico y la información del usuario.

Sesiones

Puede configurar su escritorio para restaurar ventanas y aplicaciones abiertas previamente, así como especificar programas para el arranque. Cuando sale de su sesión, tal vez quiera que ventanas abiertas y aplicaciones en ejecución se reactiven automáticamente cuando vuelva a iniciar sesión. En efecto, está guardando su sesión actual, además de hacer que se restauren los parámetros cuando vuelva a iniciar sesión. Por ejemplo, si está trabajando en una hoja de cálculo, puede guardar su trabajo pero no cerrar el archivo. Después salga de su sesión. Cuando vuelva a iniciar sesión, su hoja de cálculo seguirá abierta automáticamente donde la dejó.

En el caso de GNOME, guardar sesiones no está activado como opción predeterminada. Utilice las preferencias del cuadro de diálogo Sesiones, del panel Opciones de sesión (Sistema | Preferencias | Personal | Sesiones) para guardar sesiones. Puede guardar la sesión actual manualmente u optar por que todas sus sesiones se guarden automáticamente cuando salga de una, restaurándolas siempre regrese a sesión.

En KDE, puede configurar su administrador de sesión al seleccionar Gestor de sesiones desde la entrada Componentes de KDE, en el Centro de control. Como opción predeterminada, la sesión previa se restaura al iniciar sesión. También puede determinar el comportamiento predeterminado de apagado.

Uso de dispositivos y medios extraíbles

Los escritorios de Linux soportan ahora dispositivos y medios extraíbles, como cámaras digitales, PDA, lectores de tarjetas, incluso impresoras USB. Estos dispositivos se manejan automáticamente con una interfaz del dispositivo apropiado configurado al vuelo cuando es necesaria. Estos dispositivos se identifican y, cuando es apropiado, sus iconos se presentan en la ventana del administrador de archivo. Por ejemplo, al conectar un dispositivo USB a su sistema, será detectado y desplegado como dispositivo de almacenamiento con su propio sistema de archivos.

SUGERENCIA Al insertar un CD o DVD en blanco, se abrirá ventana con el nombre CD/DVD Creator. Grabar datos en un DVD o CD es sólo cuestión de arrastrar archivos a esa ventana y hacer clic en el botón Escribir en el disco.

Instalación de soporte multimedia: MP3, DVD y DivX

Debido a licencias y otras restricciones, muchos distribuidores de Linux no incluyen soporte multimedia para MP3, DVD o DivX en sus versiones gratuitas. Debe comprar versiones comerciales, que incluyen las licencias apropiadas para su soporte. Como opción, puede obtener este soporte desde operaciones independientes como las de fluendo.com. El soporte DivX puede obtenerse desde labs.divx.com/DivXLinuxCodec. Revise las páginas de información multimedia en el sitio Web de su distribución para documentarse más al respecto.

Interfaz de línea de comandos

Cuando utiliza la interfaz de línea de comandos, sólo se ofrece un indicador de comandos donde puede escribir su comando. Aún con una GUI, algunas veces necesita ejecutar comandos en una línea de comandos. La ventana Terminal ya no está disponible en el menú del escritorio GNOME. Ahora debe acceder a ésta desde el menú Aplicaciones | Herramientas del sistema o mediante la combinación de teclas ALT-F2, donde escribiría el comando `gnome-terminal`. Si utiliza frecuentemente la ventana Terminal, tal vez prefiera arrastrar la entrada del menú al escritorio para crear un icono de escritorio para la ventana Terminal. Sólo haga clic para abrir.

Los comandos de Linux usan ampliamente opciones y argumentos. Coloque sus argumentos y opciones con cuidado en el orden correcto en la línea de comandos. El formato para un comando de Linux es el nombre del comando, seguido por opciones y luego argumentos, como se muestra aquí:

```
$ nombre-comando opciones argumentos
```

Una *opción* es un código de letra precedido por uno o dos guiones, modificando el tipo de acción que toma el comando. Las opciones y argumentos pueden o no ser prescindibles, dependiendo del comando. Por ejemplo, el comando `ls` puede tomar una opción, `-s`. Este despliega una lista de archivos en su directorio y la opción `-s` agrega el tamaño de cada archivo en bloques. El comando y su opción se insertan en la línea de comandos de la siguiente forma:

```
$ ls -s
```

Un *argumento* son los datos que el comando necesita para ejecutar la tarea. En muchos casos, el argumento es un nombre de archivo. Un argumento se inserta como palabra en la línea de comandos tras cualquier opción. Por ejemplo, para desplegar el contenidos de un archivo, puede usar el comando `more` con el nombre de archivo como argumento. El comando `less` o `more` usado con el nombre de archivo `misdatos` se debe insertar en la línea de comandos de la siguiente forma:

```
$ less misdatos
```

En realidad, la línea de comandos es un búfer que permite la edición. Antes de oprimir enter, puede editar el texto de los comandos. Las capacidades de edición brindan una forma de corregir errores cometidos cuando escribe un comando y sus opciones. Las teclas retroceso y `supr` permiten eliminar el carácter recién escrito. Con esta capacidad de eliminación de caracteres, puede aplicar RETROCESO a toda una línea, si así lo desea, eliminando lo que insertó. CTRL-U elimina toda la línea y permite volver a iniciar desde el principio en el indicador de comandos.

SUGERENCIA Puede usar la tecla flecha hacia arriba para desplegar otra vez el último comando ejecutado. Después puede volver a ejecutar ese comando, o editarlo y ejecutarlo ya modificado. Esto es útil cuando debe repetir ciertas operaciones una y otra vez, como editar el mismo archivo. También es útil cuando ya ha ejecutado un comando que ingresó incorrectamente.

Recursos de ayuda

Una gran cantidad de documentación de soporte se encuentra instalada en su sistema y también puede acceder desde fuentes en línea. En la tabla 2-1 se muestra una lista de herramientas y recursos disponibles de Ayuda en la mayoría de sistemas Linux. Los escritorios GNOME y KDE brindan sistemas de Ayuda sustentados en una interfaz similar a un explorador para desplegar archivos de ayuda. Después puede seleccionar desde las respectivas guías de usuario de escritorio, incluido el manual de KDE, las páginas Linux Man y de información GNU. El explorador de ayuda

Recurso	Descripción
Centro de ayuda de KDE	Herramienta de ayuda de KDE, GUI para documentación del escritorio y aplicaciones KDE, páginas MAN además de documentos de información
Navegador de ayuda de GNOME	Herramienta de ayuda de GNOME, GUI para acceder a documentación del escritorio y aplicaciones GNOME, páginas MAN y documentos de información
<code>/usr/share/doc</code>	Ubicación de la documentación de aplicación
<code>man comando</code>	Páginas Man de Linux, información detallada de comandos Linux, incluidas sintaxis y opciones
<code>info aplicación</code>	Páginas de información de GNU, documentación de aplicaciones GNU

TABLA 2-1 Recursos de información

de GNOME también accede a documentos para aplicaciones GNOME como la herramienta de archivo File Roller y el cliente de correo electrónico Evolution. El explorador de ayuda de GNOME y el centro de ayuda de KDE también incorpora capacidades de exploración, incluidos marcadores y listas de historial para documentos ya vistos.

Ayuda sensible al contexto

GNOME y KDE, junto con las aplicaciones, proporcionan ayuda sensible al contexto. Cada aplicación de KDE y GNOME ofrece manuales detallados desplegados mediante sus respectivos navegadores de ayuda. Además, las herramientas administrativas ofrecen explicaciones detalladas para cada tarea.

Documentación de aplicaciones

En su sistema, el directorio `/usr/share/doc` contiene documentación de archivos instalados para cada aplicación. En cada directorio, suele encontrar documentos HOW-TO, README e INSTALL para cada aplicación.

Las páginas Man

También puede acceder a páginas Man, manuales de comandos de Linux disponibles para la interfaz de línea de comandos, al utilizar el comando `man`. Ingrese `man` con el comando sobre el que quiere información. En el siguiente ejemplo se pide información del comando `ls`:

```
$ man ls
```

Al oprimir la tecla BARRA ESPACIADORA se avanza a la siguiente página. Pulsando la tecla `b` se regresa a la página anterior. Cuando termine, oprima la tecla `q` para salir de la utilidad Man y regresar a la línea de comandos. Puede activar una búsqueda presionando ya sea el signo de diagonal (`/`) o interrogación (`?`). El signo `/` busca adelante; el signo `?` busca atrás. Cuando oprime el signo `/`, se abre una línea en la parte inferior de su pantalla, y puede ingresar una palabra para la búsqueda. Oprima `ENTER` para activar la búsqueda. Puede repetir la misma búsqueda al oprimir la tecla `n`. No necesita ingresar el patrón de nuevo.

SUGERENCIA También puede usar el sistema de ayuda de GNOME o KDE para desplegar páginas MAN y de información.

Las páginas de información

También existe documentación en línea para aplicaciones de GNU, como el compilador GNU C y C++ (gcc) y el editor Emacs, como páginas de *información* disponibles desde los centros de ayuda de GNOME y KDE. También puede acceder a esta documentación introduciendo el comando **info**. Esto iniciará una pantalla especial mostrando una lista de diferentes aplicaciones de GNU. La interfaz info tiene su propio conjunto de comandos. Aprenderá más acerca de las páginas de información ingresando **info info**. Al escribir **m** se abre una línea en la parte inferior de la pantalla, donde puede escribir las primeras letras de la aplicación. Al oprimir ENTER se abre el archivo de información de esa aplicación.

Depósitos de software

En casi todas las distribuciones de Linux, el software ha crecido tanto y experimenta actualizaciones tan frecuentes que ya no tiene sentido utilizar discos como principal medio de distribución. En cambio, la distribución se efectúa mediante depósitos de software en línea. Este depósito contiene una amplia colección de software adecuado para la distribución.

Todo este enfoque anuncia un cambio en la manera de pensar, que va de considerar casi todo el software de Linux como algo incluido en unos cuantos discos, a ver el disco como un punto de partida desde el que puede expandir cuanto quiera su software instalado desde depósitos en línea. La mayor parte del software está localizado en depósitos conectados a Internet. Ahora puede pensar en ese software como una extensión instalable, de manera sencilla, en su colección actual. Confiar en discos para su software se ha vuelto obsoleto, en cierto sentido.

Acceso y aplicaciones de Windows

En muchos casos, necesitan hacerse algunos ajustes para los sistemas Windows. Casi todos los sistemas Linux son parte de redes que también ejecutan sistemas Windows. Al emplear servidores Samba de Linux, sus sistemas Linux y Windows pueden compartir directorios e impresoras. Además, es posible que también necesite ejecutar aplicaciones de Windows directamente en su sistema Linux. A pesar de que existe una cantidad enorme de software de Linux disponible, en algunos casos necesitará o preferirá ejecutar una aplicación de Windows. La capa de compatibilidad Wine permite hacer eso para muchas aplicaciones de Windows (pero no todas).

Configuración del acceso a redes Windows: Samba

La mayor parte de redes locales y caseras incluyen sistemas que trabajan con Microsoft Windows y otros con Linux. Tal vez necesite permitir que un equipo con Windows acceda un sistema Linux o viceversa. Debido a su gran presencia en el mercado, Windows tiende a beneficiarse del soporte de controladores y aplicaciones que no se encuentran en Linux. A pesar de que existen aplicaciones equivalentes en Linux, muchas de las cuales son igual de buenas o mejores, algunas aplicaciones se ejecutan mejor en Windows, por la única razón de que el proveedor sólo desarrolla controladores para Windows.

Una solución es usar las capacidades superiores del servidor y almacenamiento de Linux para administrar y almacenar datos, mientras emplea sistemas Windows con sus controladores y aplicaciones únicas para ejecutar aplicaciones. Por ejemplo, puede instrumentar un sistema Linux para almacenar imágenes y videos, mientras utiliza Windows para mostrarlos. Las imágenes y videos pueden transmitirse a través de su enrutador al sistema que quiere los ejecute. En realidad, muchos sistemas DVR comerciales utilizan una versión de Linux para administrar grabación y almacenamiento de videos. Otro uso es permitir que los sistemas Windows manejen dispositivos como impresoras, que pueden conectarse a un sistema Linux o a la inversa.

Para permitir que Windows acceda a Linux y Linux a un sistema Windows, se usa el servidor Samba. Samba tiene dos métodos de autenticación, compartidos y usuarios, aunque el método para compartir se ha descontinuado. La autenticación de usuario requiere que existan cuentas correspondientes en sistemas Windows y Linux. Es necesario configurar un usuario de Samba con una contraseña de Samba. El usuario de Samba debe tener el mismo nombre que una cuenta establecida. El usuario de Windows y Samba pueden tener el mismo nombre, aunque un usuario de Windows puede correlacionarse con uno de Samba. Es posible abrir un recurso compartido a usuarios específicos y funcionar como extensión del espacio de almacenamiento del usuario. En las distribuciones más recientes, la información de usuario y contraseña de Samba puede mantenerse en archivos de base de datos de Samba tdb (trivial data base, base de datos trivial) de los archivos de la base de datos de Samba, para editarse y agregarse mediante el comando **pdbedit**.

Para configurar la posibilidad de compartición de archivos simples en un sistema Linux, primero necesita configurar su servidor Samba. Esto se hace editando directamente el archivo **/etc/samba/samba.conf**. Si editó el archivo **/etc/samba/samba.conf**, necesita especificar primero el nombre de su red Windows. Samba proporciona una herramienta de configuración denominada SWAT, para usarse con cualquier explorador para configurar su servidor Samba, agregar usuarios y configurar archivos compartidos. Algunas distribuciones, como Ubuntu, configuran Samba automáticamente. KDE también proporciona una configuración de Samba.

Una vez configurado, GNOME y KDE permiten explorar y acceder a archivos compartidos de Samba desde su escritorio, permitiéndole también acceder a directorios e impresoras compartidos de Windows en otros sistemas. En GNOME haga clic en Red y después en el icono Red de Windows, desde la ventana Mi PC. Verá un icono para su red de Windows. En GNOME o KDE puede insertar **smb:** URL en la ventana del administrador de archivos para acceder a sus redes de Windows.

Cuando un usuario de Windows quiere acceder a los archivos compartidos en el sistema Linux, abre Mis sitios de red (Red en Vista) y después selecciona Agregar un sitio red, para añadir un recurso de red compartido, o Ver grupos de trabajo para ver las computadoras en su red de Windows. Al seleccionar el servidor Samba de Linux se desplegarán sus recursos compartidos de Samba. Para acceder a los recursos compartidos, se requerirá que el usuario ingrese nombre de usuario y contraseña de Samba. Tiene la opción de hacer que se recuerde el nombre de usuario y contraseña para un acceso automático.

NOTA La herramienta *Fuse-smb* le permite explorar toda su red Windows de una sola vez.

Ejecución de software de Windows en Linux: Wine

Wine es una capa de compatibilidad de Windows que permitirá ejecutar muchas aplicaciones de Windows de manera nativa en Linux. Aunque puede ejecutar el sistema operativo de Windows dentro de ésta, el sistema operativo de Windows real no es necesario. Las aplicaciones de Windows se ejecutarán como si fueran aplicaciones de Linux, capaces de acceder al sistema de archivos completo de Linux y usar dispositivos conectados a Linux. Es muy probable que no se ejecuten las aplicaciones en total dependencia de un controlador, como juegos con gran cantidad de gráficos. Asimismo, es probable que otros, como los lectores de noticias, que no recaen en ningún controlador especializado, se ejecuten muy bien. En el caso de algunas aplicaciones, tal vez también necesite copiar varios DLL específicos de Windows, desde un sistema Windows en ejecución a su directorio **system32** o **system** de Wine Windows.

Para instalar Wine en su sistema, busque **wine** en los depósitos de distribuciones. En el caso de algunas distribuciones, quizás deba descargar wine directamente desde **winehq.org**. Se proporcionan binarios para varias distribuciones.

SUGERENCIA Para jugar juegos de Windows en Linux, puede intentar usar *cedega*. Estos son controladores comerciales baratos que se configuran para dar soporte a muchos juegos populares, **cedega.com**, que permite una aceleración completa de gráficos.

Una vez instalado, un menú Wine aparecerá en el menú Aplicaciones. El menú Wine almacena entradas para la configuración de Wine, el software para desinstalar Wine y el archivo del explorador Wine, además de un editor de registro *regedit*, bloc de notas y una herramienta de ayuda de Wine.

Para configurar Wine, un usuario inicia la herramienta Configuration de wine. Esto abre una ventana con paneles para Aplicaciones, Bibliotecas (selección DLL), Audio (controladores de sonido), Controladores, Integración de escritorio y Gráficos. En el panel Aplicaciones puede seleccionar para qué versión de Windows está diseñada la aplicación. En el panel Controladores se mostrará una lista de particiones detectadas, además de sus controladores emulados de Windows, no una partición de tamaño fijo. Su sistema de archivos actual de Linux se mostrará en la lista como Zi.

Una vez configurado, Wine configura un directorio **.wine** en el directorio home del usuario (el directorio está escondido, así que habilite Mostrar los archivos ocultos en el menú Ver, del explorador de archivos para desplegarlo). Dentro de ese directorio encontrará el directorio **drive-c**, que funciona como disco C:, para almacenamiento de archivos del sistema y archivos de programa de su sistema Windows, dentro de subdirectorios **Windows** y **Program File**. Los directorios **System** y **System32** se localizan en el directorio **Windows**. Aquí es donde coloca cualquier archivo DLL necesario. El directorio Program Files almacenará sus programas de Windows instalados, como se instalarían en un directorio **Archivos de programa** de Windows.

Para instalar una aplicación de Windows con Wine, puede usar la herramienta de configuración de Wine o abrir una ventana Terminal y ejecutar el comando **wine**, con la aplicación de Windows como argumento. En el siguiente ejemplo se instala un programa *newsbin* popular:

```
$ wine newsbin.exe
```

Para instalar con la herramienta Configuración de Windows, seleccione el panel Aplicaciones y después haga clic en Agregar.

Algunas aplicaciones, como *newsbin*, también requerirán que use archivos DLL desde un sistema operativo Windows en ejecución. Los archivos DLL normalmente se copian en el directorio **.wine/drive_c/Windows/system32**.

Los iconos del software instalado de Windows se mostrarán en su escritorio. Sólo haga doble clic en un icono para iniciar la aplicación. Se ejecutará normalmente dentro de una ventana de Linux, como cualquier aplicación de Linux.

Instalar las fuentes de Windows en Wine sólo es cuestión de copiar fuentes desde un directorio de fuentes de Windows a su directorio de Wine **.wine/drive_c/Windows/fonts**. Puede copiar cualquier archivo **.ttf** de Windows a este directorio para instalar una fuente. Asimismo es posible usar las fuentes de Microsoft disponibles en Web en **fontconfig.org** (esto requiere **cabextract** para extraerlos).

Wine usará un estilo de ventana simple para características como botones y la barra de título. Si quiere utilizar el estilo XP, descargue e instale el tema Royal desde Microsoft. Sin embargo, tenga en cuenta que el soporte a este tema consume muchos recursos y es probable que haga más lento su sistema.

SUGERENCIA Como opción, puede usar la capa de compatibilidad de Windows llamada *CrossoverOffice*. Es un producto comercial probado, para ejecutar aplicaciones como Microsoft Office. Revise la página **codeweavers.com** para adquirir más información. *CrossoverOffice* se basa en Wine, que *CodeWeavers* soporta directamente.

La shell y estructura de archivos Linux

CAPÍTULO 3

La shell

CAPÍTULO 4

Secuencias de comandos y
programación de la shell

CAPÍTULO 5

Configuración de la shell

CAPÍTULO 6

Documentos, directorios
y archivos de Linux

3

CAPÍTULO

La shell

La *shell* es un intérprete de comandos cuya interfaz se orienta al trabajo en línea, interactiva y no interactiva, entre usuario y sistema operativo. Usted inserta comandos en la línea de comandos; las *shell* los interpreta y envía instrucciones al sistema operativo (se puede acceder a la interfaz de línea de comandos GNOME y KDE a través de las ventanas Terminal, del menú Aplicaciones/ Accesorios). También puede colocar comandos en un archivo de secuencia de comandos para ejecutarse consecutivamente, de forma similar a un programa. Esta capacidad de interpretación de la *shell* incorpora muchas características sofisticadas. Por ejemplo, la *shell* tiene un conjunto de archivos de coincidencia de caracteres que generan nombres de archivo. La *shell* puede redirigir la entrada y salida, además de ejecutar operaciones en segundo plano, evitándole realizar otras tareas.

Varios tipos diferentes de *shells* se han desarrollado para Linux: Bourne Again Shell (BASH), Korn, TCSH y Z. TCSH es una versión mejorada de *shell* C, usada en varios sistemas Unix, especialmente las versiones BSD. Sólo necesita un tipo de *shell* para su trabajo. Linux incluye casi todas las *shells*; sin embargo, se instala y usa la *shell* BASH, como opción predeterminada. Si utiliza la *shell* de línea de comandos, entrará a BASH, a menos que especifique otra. En este capítulo se analiza principalmente la *shell* BASH, que comparte muchas características con otras *shell*. Un breve recuento de las *shell* C, TCSH y Z sigue al final del capítulo, donde se indican las diferencias.

Conocerá más acerca de las *shell* en sus sitios Web respectivos, como figura en la lista de la tabla 3-1. Además, está disponible un manual en línea detallado para cada *shell* instalada. Introduzca el comando **man** y la palabra clave de la *shell* para acceder a ésta, **bash** para BASH, **ksh** para Korn, **zsh** para Z y **tsch** para TSCH. En el caso de la *shell* C, puede usar **csch**, vinculada con **tsch**. Por ejemplo, el comando **man bash** accederá al manual en línea de la *shell* BASH.

NOTA Conocerá más acerca de la *shell* BASH en gnu.org/software/bash. Un manual en línea detallado está disponible en su sistema Linux, al usar el comando **man** con la palabra clave **bash**.

La línea de comandos

La interfaz de línea de comandos de Linux consta de una sola línea en que se ingresan comandos con cualquiera de sus opciones y argumentos. Desde GNOME o KDE, accederá a la interfaz de línea de comandos abriendo la ventana Terminal. En caso de que quiera iniciar Linux con la interfaz de línea de comandos, se presentará la correspondiente a BASH cuando inicie sesión.

URL	Shell
gnu.org/software/bash	El sitio Web de BASH con el manual en línea, FAQ y lanzamientos recientes
gnu.org/software/bash/manual/bash.html	Manual en línea de BASH
zsh.org	Sitio Web de la shell Z con referencias a documentos FAQ y descargas recientes
tcsch.org	Sitio Web de TCSH con soporte detallado que incluye manuales, sugerencias, FAQ y lanzamientos recientes
kornshell.com	Sitio Web de Korn, con manuales, FAQ y referencias

TABLA 3-1 Sitios Web de las shells de Linux

Como opción predeterminada, la shell BASH tiene un indicador de comandos de signo pesos (\$), pero Linux tiene varios otros tipos de shells, cada uno con su propio indicador de comandos (% para la shell C, por ejemplo). El usuario root tendrá un indicador de comandos diferente, el signo # (almohadilla). Un *indicador de comandos* de shell, como el mostrado aquí, marca el comienzo de la línea de comandos:

```
$
```

Puede insertar un comando junto con opciones y argumentos en el indicador. Por ejemplo, con la opción **-l**, el comando **ls** desplegará una línea de información acerca de cada archivo, mostrando una lista de datos, como tamaño, hora y fecha en que fue modificado. Un guión antes de la opción **-l** es necesario. Linux emplea para distinguir una opción de un argumento.

```
$ ls -l
```

Si quiere que la información se despliegue sólo para un archivo en particular, puede agregar el nombre del archivo como argumento, seguido de la opción **-l**:

```
$ ls -l misdatos
-rw-r--r-- 1 chris weather 207 Feb 20 11:55 misdatos
```

SUGERENCIA Algunos comandos pueden ser complejos y tomar tiempo para ejecutarse. Cuando ejecuta por error el comando equivocado, puede interrumpir y detener dichos comandos con la tecla de interrupción (CTRL-C).

Puede ingresar un comando en varias líneas al escribir una diagonal invertida justo antes oprimir ENTER. La diagonal invertida actúa como “escape” al comando ENTER (es decir, omite el final del comando), prosiguiendo la misma orden para la siguiente línea. En el siguiente ejemplo, el comando **cp** se ingresa en tres líneas:

```
$ cp -l \
misdatos \
/home/george/miproyecto/nuevosdatos
```

También puede insertar varios comandos en la misma línea al separarlos con punto y coma (;). En realidad el punto y coma actúa como una operación de ejecución. Los comandos se ejecutarán en la secuencia que se hayan insertado. El siguiente comando ejecuta **ls** seguido por **date**.

```
$ ls ; date
```

También puede ejecutar condicionalmente varios comandos en la misma línea con el operador **&&** (consulte el Capítulo 4). Un comando sólo se ejecuta si el anterior es correcto. Esta característica es útil cuando procesa varias secuencias de comandos dependientes en la misma línea. En el siguiente ejemplo, el comando **ls** sólo se ejecuta si **date** es correcto.

```
$ date && ls
```

SUGERENCIA Los comandos también pueden ejecutarse como argumentos en una línea de comandos, al usar sus resultados para otros comandos. Para ejecutar un comando en una línea de comandos, debe encerrarlo entre comillas inversas; consulte “Valores de comandos de Linux”, en el capítulo 4.

Edición en la línea de comandos

La shell BASH, que es la predeterminada, tiene capacidades en edición de línea de comandos especiales, que resultarán útiles mientras aprende Linux (vea la tabla 3-2). Puede modificar de manera sencilla los comandos que haya insertado antes de ejecutarlos, moviéndose por cualquier lugar de la línea de comandos, insertando o eliminando caracteres. Esto es útil, sobre todo en el caso de comandos complejos. Puede usar CTRL-F o FLECHA A LA DERECHA para moverse hacia delante un carácter o CTRL-B o FLECHA A LA IZQUIERDA para moverse hacia atrás un carácter. CTRL-D o SUPR elimina el carácter donde se encuentra el cursor y CTRL-H o RETROCESO elimina el carácter anterior al cursor. Para agregar texto, utilice las teclas de las flechas para mover el cursor a donde quiera insertar texto y luego escriba los nuevos caracteres. Incluso puede cortar palabras con las teclas CTRL-W o ALT-D y después utilizar CTRL-Y para pegarlas en otra posición, moviendo así las palabras. Como regla general, la versión CTRL del comando opera en los caracteres, y la versión ALT funciona con palabras, como CTRL-T para trasponer caracteres y ALT-T para trasponer palabras. En cualquier momento, puede oprimir ENTER para ejecutar el comando. Las asociaciones actuales de teclas y sus tareas, junto con la configuración global, se especifican en el archivo **/etc/inputrc**.

SUGERENCIA Las capacidades de edición de la línea de comandos BASH, se derivan de Readline, que soporta numerosas operaciones. Incluso puede establecer una operación de edición para una tecla. Readline usa el archivo **/etc/inputrc** para configurar la unión teclas a dichas operaciones. Su archivo de configuración **shell /etc/profile**, lee automáticamente este archivo cuando inicia sesión (consulte el Capítulo 5). Los usuarios pueden personalizar sus comandos de edición al crear un archivo **.inputrc** en su directorio home (éste es un archivo de “punto”). Tal vez sea mejor copiar (respaldar) primero el archivo **/etc/inputrc**, al igual que **.inputrc**, y después editarlo. **/etc/profile** revisará primero un archivo **.inputrc** local antes de acceder al archivo **/etc/inputrc**. Conocerá más acerca de Redline en el manual de referencia de la shell BASH en gnu.org/manual/bash.

Comandos de desplazamiento	Operación
CTRL-F, FLECHA A LA DERECHA	Se mueve un carácter adelante.
CTRL-B, FLECHA A LA IZQUIERDA	Se mueve un carácter atrás.
CTRL-A O INICIO	Se mueve al principio de la línea.
CTRL-E O FIN	Se mueve al final de la línea.
ALT-F	Se mueve una palabra hacia adelante.
ALT-B	Se mueve una palabra hacia atrás.
CTRL-L	Limpia la pantalla y coloca la línea en la parte superior.
Comandos de edición	Operación
CTRL-D O SUPR	Elimina el carácter en que está el cursor.
CTRL-H O RETROCESO	Elimina el carácter anterior al cursor.
CTRL-K	Corta lo que sobra de la línea desde la posición del cursor (hacia la derecha).
CTRL-U	Corta desde la posición del cursor al principio de la línea (hacia la izquierda).
CTRL-W	Corta la palabra previa.
CTRL-C	Corta toda la línea.
ALT-D	Corta lo que sobra de una palabra.
ALT-SUPR	Corta desde el cursor al principio de la palabra.
CTRL-Y	Pega el texto cortado antes.
ALT-Y	Pega desde el conjunto de texto cortado antes.
CTRL-Y	Pega el texto cortado antes.
CTRL-V	Inserta el texto entre comillas, usado para insertar teclas de control o meta (ALT) como texto, por ejemplo CTRL-B para retroceso o CTRL-T para tablas.
ALT-T	Transpone la palabras actual y previa.
ALT-L	Pone en minúscula la palabra actual.
ALT-U	Pone en mayúscula la palabra actual.
ALT-C	Pone en mayúsculas la primera letra de la palabra actual.
CTRL-SHIFT-__	Deshace los cambios previos.

TABLA 3-2 Operaciones de edición de la línea de comandos

Ayuda para completar comandos y nombres de archivo

La línea de comandos BASH tiene una característica integrada, completa la línea de comandos y nombres de archivo. Éstos se concluyen automáticamente al usar la tecla TAB. Si inserta un patrón inconcluso, como un argumento de comando o nombre de archivo, puede presionar la tecla TAB

para activar la característica que remata el comando o nombre de archivo. Los directorios tendrán un / adjunto al nombre. Si más de un comando o archivo tienen el mismo prefijo, la shell sólo hace beep, en espera de que presione nuevamente la tecla TAB. Después despliega una lista de posibles comandos y espera que agregue caracteres suficientes para seleccionar un comando o nombre específico de archivo. En situaciones donde sabe que existen varias posibilidades, puede oprimir la tecla ESC en vez de dos TAB. En el siguiente ejemplo, el usuario envía el comando **cat** con un nombre de archivo incompleto. Cuando el usuario oprime la tecla TAB, el sistema busca una coincidencia y, al encontrarla, concluye el nombre de archivo. El usuario puede oprimir ENTER para ejecutar el comando.

```
$ cat pre tab
$ cat prefacio
```

La acción automática para completar también funciona con nombres de variables, usuarios y hosts. En este caso, el texto parcial debe antecederse con un carácter especial indicando el tipo de nombre. Las variables inician con un signo \$, así que cualquier texto cuyo inicio es con el signo \$ se trata como variable que habrá de completarse. Las variables se seleccionan a partir de variables predefinidas, como las de la shell del sistema (consulte el capítulo 4). Los nombres de usuario inician con una tilde (~). Los nombres de host inician con el signo @, con posibles nombres tomados del archivo **/etc/hosts**. A continuación se muestra una lista de opciones automáticas para completar:

- Los nombres de archivo inician con cualquier texto o /.
- El texto para variables de la shell comienza con el signo \$.
- El texto de nombre de usuario comienza con el signo ~.
- El texto de nombre host comienza con el signo @.
- Los comandos, alias y texto en archivos comienzan con texto normal.

Por ejemplo, para completar la variable HOME dando sólo \$HOM, ingrese simplemente un carácter TAB.

```
$ echo $HOM <tab>
$ echo $HOME
```

Si inserta sólo una H, entonces puede ingresar dos tabuladores para ver todas las variables posibles que empiezan con H. La línea de comandos se desplegará de nuevo, permitiéndole completar el nombre.

```
$ echo $H <tab> <tab>
$HISTCMD $HISTFILE $HOME $HOSTTYPE HISTFILE $HISTSIZE $HISTNAME
$ echo $H
```

También puede seleccionar específicamente qué tipo de texto se completará, usando comandos de tecla correspondientes. En este caso, no importa con que tipo de signo comience el nombre. Por ejemplo, ALT-~ tratará el texto actual como nombre de usuario. ALT-@ lo tratará como nombre de host y ALT-\$ como variable. ALT-! lo tratará como comando. Para desplegar una lista de opciones para completar, use las teclas CTRL-X con la tecla apropiada, como en CTRL-X-\$, para presentar una lista de opciones para completar variables. Vea una lista completa en la tabla 3-3.

Comandos (CTRL-R para mostrar una lista de opciones para completar)	Descripción
TAB	Completa automáticamente
TAB TAB O ESC	Muestra una lista de opciones posibles para completar
ALT-/ , CTRL-R-/	Completa el nombre de archivo, texto normal para completar automáticamente
ALT-\$, CTRL-R-\$	Completa variables shell, \$ para completar automáticamente
ALT-~ , CTRL-R-~	Completa el nombre de usuario, ~ para completar automáticamente
ALT-@ , CTRL-R-@	Completa el nombre de host, @ para completar automáticamente
ALT-! , CTRL-R-!	Completa el nombre de comando, texto normal para completar automáticamente

TABLA 3-3 Comandos para completar texto en la línea de comandos

Historial

BASH mantiene una lista de comandos ingresados previamente, denominada *lista de historial*. Puede desplegar cada comando, uno tras otro, en su línea de comandos oprimiendo la tecla FLECHA HACIA ARRIBA. La tecla FLECHA HACIA ABAJO lo desplaza hacia abajo en la lista. Puede modificar y ejecutar cualquiera de estos comandos previos al desplegarlos en su línea de comandos.

***SUGERENCIA** La capacidad para volver a desplegar un comando es útil cuando ya ha ejecutado uno que insertó incorrectamente. En este caso, se le presenta un mensaje de error y una nueva línea de comandos vacía. Al oprimir la tecla flecha hacia arriba de nuevo, desplegará su comando previo y podrá corregirlo, para volver a ejecutarlo después. De esta forma, no necesita escribir todo el comando de nuevo.*

Eventos de historial

En BASH, la *utilería de historial* mantiene un registro de comandos ejecutados más recientes. Los comandos se numeran comenzando desde el 1 y existe un número límite de comandos recordados (la opción predeterminada es 500). La *utilería de historial* es un tipo de memoria a corto plazo, que da seguimiento a los comandos más recientes ejecutados. Para ver el conjunto de comandos más recientes, escriba **history** en la línea de comandos y oprima ENTER. Se despliega una lista de comandos recientes, precedidos por un número.

```
$ history
1 cp misdatos hoy
2 vi misdatos
3 mv misdatos informes
4 cd informes
5 ls
```

A cada uno de estos comandos se le conoce de manera técnica como eventos. Un *evento* describe acciones tomadas (un comando ejecutado). Los eventos se enumeran de acuerdo con su secuencia de ejecución. El evento más reciente tiene el número mayor. Cada uno de estos eventos puede identificarse por su número o el carácter inicial del comando.

La utilidad `history` permite hacer referencia a un evento anterior, colocándolo en su línea de comandos y permitiéndole ejecutarlo. La manera más sencilla para hacer esto es mediante las teclas `FLECHA HACIA ARRIBA` y `FLECHA HACIA ABAJO`, para presentar eventos del historial en su línea de comandos, de uno en uno. No necesita desplegar primero la lista con `history`. Oprimiendo la tecla `FLECHA HACIA ARRIBA` una vez, coloca el último comando del historial. Al oprimir la tecla `FLECHA HACIA ABAJO`, coloca el siguiente evento en la línea de comandos.

Puede usar ciertos controles y meta teclas (combinaciones de teclado) para realizar otras operaciones, como buscar en la lista del historial. Una meta tecla es `ALT` (o `ESC`, en teclados que carecen de ella). Aquí se usará La tecla `ALT`. `ALT-<` lo llevará al principio de la lista del historial; `ALT->` hará una búsqueda. `CTRL-S` y `CTRL-R` realizarán búsquedas incrementales, desplegando comandos coincidentes mientras escribe una búsqueda. En la tabla 3-4 se muestra una lista de diferentes comandos para referir la lista del historial.

SUGERENCIA Si más de un evento del historial coincide con lo ingresado, escuchará un `beep` y después podrá ingresar más caracteres, ayudando que se identifique únicamente el evento.

Comandos del historial	Descripción
CTRL-N O FLECHA HACIA ABAJO	Se mueve hacia abajo, al siguiente evento en la lista del historial.
CTRL-P O FLECHA HACIA ARRIBA	Se mueve hacia arriba, al evento anterior en la lista del historial.
ALT-<	Se dirige al principio de la lista del historial de eventos.
ALT->	Se dirige al final de la lista del historial de eventos.
ALT-N	Siguiente búsqueda, siguiente elemento coincidente.
ALT-P	Búsqueda anterior, elemento coincidente previo.
CTRL-S	Búsqueda hacia adelante del historial, búsqueda incremental hacia adelante.
CTRL-R	Búsqueda en reversa del historial, búsqueda incremental en reversa.
fc referencia-evento	Edita un evento con el editor estándar y luego lo ejecuta Opciones -l Muestra una lista del historial de eventos recientes; igual al comando <code>history</code> -e editor referencia-evento; invoca un editor específico para editar un evento específico
Referencias a eventos de historial	
!núm de evento	Hace referencia a un evento por su número.
!!	Hace referencia al comando previo.
!caracteres	Hace referencia a un evento comenzando con un carácter específico.
!?patrón?	Hace referencia a un evento conteniendo un patrón específico.
!-núm de evento	Refiere un evento con desplazado a partir del primero.
!núm-núm	Refiere a un rango de eventos.

TABLA 3-4 Comandos y referencias a eventos de historial

También puede referir eventos del historial y ejecutarlos mediante el comando `!` de historial. El signo `!` es seguido por una referencia que identifica el comando. La referencia puede ser el número de evento o conjunto inicial de caracteres en el evento. En el siguiente ejemplo, se hace referencia al tercer comando en la lista del historial, primero por número y después por caracteres iniciales.

```
$ !3
mv misdatos informes
$ !mv mi
mv misdatos informes
```

También puede hacer referencia a un evento usando un desplazamiento desde el final de la lista. Un número negativo desplazará desde el final de la lista a ese evento, por lo que hará referencia a éste. En el siguiente ejemplo, se hace referencia al cuarto comando, `cd misdatos`, usando un desplazamiento negativo y después se ejecuta. Recuerde que el desplazamiento se efectúa desde el final de la lista (en este caso, el evento 5) hacia arriba, al principio de la lista, el evento 1. Un desplazamiento de 4, empezando desde el evento 5, lo coloca en el evento 2.

```
$ !-4
vi misdatos
```

Para hacer referencia al último evento, use un `!` posterior, como en `!!`. En el siguiente ejemplo, el comando `!!` ejecuta el último comando aplicado por el usuario (en este caso, `ls`):

```
$ !!
ls
misdatos hoy informes
```

Edición del historial de eventos

También puede editar cualquier evento en la lista del historial antes de ejecutarlo. En la shell BASH, esto es posible de dos formas. Puede usar la capacidad del editor de línea de comandos para referir cualquier evento en la lista del historial y editarlo. También puede usar una opción de comando `fc` del historial para dirigirse a un evento y editarlo con el editor Vi completo. Cada método requiere dos capacidades de edición diferentes. La primera opción se limita los comandos en el editor de línea de comandos, que sólo edita una línea con un subconjunto de comandos Emacs. Sin embargo, al mismo tiempo, le permite referir eventos de manera sencilla en la lista del historial. El segundo método invoca el editor estándar Vi con todas sus características, pero sólo para un evento específico del historial.

Con el editor de línea de comandos, no sólo puede editar el comando actual, también puede dirigirse a un evento anterior en la lista del historial para editarlo y ejecutarlo. Después, el comando `CTRL-P` lo llevará al evento anterior en la lista. El comando `CTRL-N` lo conduce hacia abajo de la lista. El comando `ALT-<` lo dirige a la parte superior de la lista y `ALT->` a la parte inferior. Incluso puede usar un patrón para buscar un evento dado. Una diagonal, seguida de un patrón, busca hacia atrás en la lista y un signo de interrogación, seguido de un patrón, busca hacia adelante. El comando `n` repite la búsqueda.

Una vez haya localizado el evento que desea editar, puede usar los comandos de edición de Emacs, en la línea de comandos, para editar la línea. `CTRL-D` elimina un carácter. `CTRL-F` o `FLECHA A LA DERECHA` lo mueve hacia adelante un carácter y `CTRL-B` o `FLECHA A LA IZQUIERDA`, lo mueve hacia atrás un carácter. Para agregar texto, coloque el cursor y escriba los caracteres que quiere.

Si quiere editar un evento mediante el editor estándar, necesita referirlo mediante el comando `fc` y un evento de referencia específico, como su número. El editor a tomarse en cuenta para la shell

es el especificado en la variable **FCEDIT** o **EDITOR**. Esto determina el tipo de editor preestablecido para el comando **fc**. Puede asignar a la variable **FCEDIT** o **EDITOR** un editor diferente si así lo desea, como Emacs en vez de Vi. En el siguiente ejemplo se editará el cuarto evento, **cd informes**, con el editor estándar y después se ejecutará el evento ya editado:

```
$ fc 4
```

Puede seleccionar más de un comando a la vez para edición y ejecución, si hace referencia a un rango de comandos. Podría elegir un rango de comandos al indicar un identificador para el primer comando seguido por otro identificador para el último comando dentro del rango. Un identificador puede ser el número de comando o el inicio de los caracteres en el comando. En el siguiente ejemplo, el rango de comandos del 2 al 4 se edita y ejecuta, primero usando números de evento y después el inicio de los caracteres en esos eventos:

```
$ fc 2 4
$ fc vi c
```

El comando **fc** recurre al editor predeterminado especificado en la variable especial **FCEDIT** (si **FCEDIT** no está definido, busca la variable **EDITOR**; si ninguno está definido, use Vi). Generalmente, éste es el editor Vi. Si quiere usar el editor Emacs en lugar de Vi, recurra a la opción **-e** y el término **emacs** al invocar **fc**. En el siguiente ejemplo se editará el cuarto evento, **cd reports**, con el editor Emacs y después se ejecutará el evento editado:

```
$ fc -e emacs 4
```

Configuración del historial: HISTFILE e HISTSAVE

El número de eventos guardados por su sistema se almacena en una variable especial del sistema denominada **HISTSIZE**. Como opción predeterminada, suele ser 500. Puede cambiar esto a otro número, asignando simplemente un nuevo valor a **HISTSIZE**. En el siguiente ejemplo, el usuario cambia el número de eventos de historial guardados a 10:

```
$ HISTSIZE=10
```

Los eventos reales del historial se guardan en un archivo cuyo nombre se conserva en una variable especial, denominada **HISTFILE**. Como opción predeterminada, éste es el archivo **.bash_history**. Sin embargo, puede cambiar el archivo en que se guarda el historial de eventos, asignando el nombre a la variable **HISTFILE**. En el siguiente ejemplo, se despliega el valor de **HISTFILE**. Después, se asigna un nuevo nombre de archivo, **newhist**. El historial de eventos se guarda en el archivo **newhist**.

```
$ echo $HISTFILE
.bash_history
$ HISTFILE="newhist"
$ echo $HISTFILE
newhist
```

Expansion del nombre de archivo: *, ?, []

Los nombres de archivo son los argumentos más comunes usados en un comando. En ocasiones, tal vez sólo conozca parte del nombre del archivo o sólo quiera referir varios nombres de archivo con la misma extensión o comenzando con los mismos caracteres. La shell proporciona un conjunto de

caracteres especiales que buscan, relacionan y generan una lista de nombres de archivo. Estos son asteriscos, signos de interrogación y corchetes (*, ?, []). Dado un nombre de archivo parcial determinado, la shell utiliza estos operadores de relación para buscar archivos y expandir una lista de nombres de archivo encontrados. La shell reemplaza el argumento del nombre de archivo con la lista expandida de nombres de archivo coincidentes. Estos nombres de archivo pueden convertirse en argumentos para comandos como `ls`, que pueden operar en muchos archivos. En la tabla 3-5 se muestra una lista de caracteres de expansión de archivos de shell.

Símbolos de shell comunes	Ejecución
ENTER	Ejecuta una línea de comandos.
;	Separa comandos en la misma línea de comandos.
<code>`comando`</code>	Ejecuta un comando.
<code>\$ (comando)</code>	Ejecuta un comando.
[]	Coincide con una clase de caracteres posibles en nombres de archivo.
\	Cita el siguiente carácter. Usado para citar caracteres especiales.
	Canaliza la salida estándar de un comando, como la entrada para otro comando.
&	Ejecuta un comando en segundo plano o instancia.
!	Comando de historial.
Símbolos de expansión de archivos	Ejecución
*	Coincide con cualquier conjunto de caracteres en nombres de archivo.
?	Coincide con cualquier carácter en nombres de archivo.
[]	Coincide con una clase de caracteres en nombres de archivo.
Símbolos de redireccionamiento	Ejecución
>	Redirige la salida estándar a un archivo o dispositivo, creando el archivo cuando no existe y sobrescribiéndolo, si existe.
>!	Obliga la sobrescritura de un archivo, si ya existe. Esto invalida la opción <code>noclobber</code> .
<	Redirige la entrada estándar desde un archivo o dispositivo a un programa.
>>	Redirige la salida estándar de un archivo o dispositivo, adjuntando la salida al final del archivo.
Símbolos de redireccionamiento de errores estándar	Ejecución
2>	Redirige el error estándar a un archivo o dispositivo.
2>>	Redirige y adjunta el error estándar a un archivo o dispositivo.
2>&1	Redirige un error estándar a la salida estándar.
>&	Redirige el error estándar a un archivo o dispositivo.
&	Canaliza el error estándar como entrada de otro comando.

TABLA 3-5 Símbolos de shell

Relación de varios caracteres

El asterisco (*) hace referencia a los archivos que comienzan o terminan con un conjunto específico de caracteres. Coloque el asterisco antes o después de un conjunto de caracteres formando un patrón, para buscar nombres de archivos. Si el asterisco se coloca antes del patrón, los nombres de archivo que terminan en ese patrón son buscados. Si el asterisco se coloca después del patrón, se buscan los nombres de archivo iniciando con ese patrón. Cualquier archivo coincidente se copia a una lista de nombres de archivos generados por esta operación. En el siguiente ejemplo, todos los nombres de archivo comenzando con el patrón “doc” se buscan y se genera una lista. Todos los nombres de archivo terminando con el patrón “nes” se buscan y se genera una lista. En el último ejemplo, se muestra cómo el * puede usarse en cualquier combinación de caracteres.

```
$ ls
doc1 doc2 documentos docs luisdocs lunes viernes
$ ls doc*
doc1 doc2 documento docs
$ ls *nes
lunes viernes
$ ls l*n*
lunes
$
```

Los nombres de archivo a menudo incluyen una extensión especificada con un punto, seguida por una cadena de caracteres denotando el tipo de archivo, como .c para archivos C, .ccp para archivos C++, incluso .jpg para archivos de imagen JPEG. La extensión no tiene un rango especial y sólo es parte de los caracteres que hacen el nombre de archivo. El uso del asterisco hace más sencilla la selección de archivos con una extensión dada. En el siguiente ejemplo, el asterisco se usa para mostrar una lista específicamente de archivos con extensión .c. El asterisco se coloca antes de que .c constituya el argumento para **ls**.

```
$ ls *.c
calc.c principal.c
```

Puede utilizar el * con el comando **rm** para eliminar varios archivos a la vez. El asterisco al principio selecciona una lista de archivos con una extensión dada, el principio o final de un conjunto de caracteres y después presenta una lista de archivos que eliminará el comando **rm**. En el siguiente ejemplo, el comando **rm** elimina todos los archivos que comienzan con el patrón “doc”:

```
$ rm doc*
```

SUGERENCIA Use el carácter de expansión de archivo * con cuidado y moderación con el comando **rm**. La combinación puede ser peligrosa. Un * colocado equivocadamente en un comando **rm** sin la opción **-i**, puede eliminar fácilmente todos los archivos del directorio actual. La opción **-i** pedirá al usuario primero que confirme si quiere eliminar los archivos.

Coincidencia de un solo carácter

El signo de interrogación (?) sirve de comodín para un solo carácter en los nombres de archivo. Suponga que quiere relacionar los archivos **doc1** y **docA**, pero no el archivo **documento**. Mientras con el asterisco se buscarán coincidencias entre nombres de archivo de cualquier tamaño, el signo de interrogación limita la coincidencia a un solo carácter extra.

En el siguiente ejemplo se relacionan los archivos que comienzan con la palabra “doc” seguida por una sola letra diferente:

```
$ ls
doc1 docA documento
$ ls doc?
doc1 docA
```

Relación de un rango de caracteres

Mientras los caracteres de expansión de archivo `*` y `?` especifican porciones incompletas de un nombre de archivo, los corchetes (`[]`) permiten especificar un conjunto de caracteres válidos de búsqueda. Se rastreará una coincidencia de cualquier carácter entre los corchetes con el nombre de archivo. Suponga que quiere mostrar una lista con los archivos comenzando con “doc”, pero sólo terminan con `1` o `A`. No está interesado en nombres de archivos terminando en `2` o `B` u otro carácter. Aquí se muestra cómo:

```
$ ls
doc1 doc2 doc3 docA docB docD documento
$ ls doc[1A]
doc1 docA
```

También puede especificar un conjunto de caracteres como rango, en vez de mostrarlos uno por uno. Un guión colocado entre los límites superior e inferior de un conjunto de caracteres, selecciona todos los caracteres en del rango. El rango suele determinarse por el conjunto de caracteres en uso. En un conjunto de caracteres ASCII, el rango “a-g” seleccionará todos los caracteres en minúscula alfabéticamente desde la `a` hasta la `g`. En el siguiente ejemplo, se eligen archivos que comienzan con el patrón “doc” y terminan en los caracteres del `1` al `3`. Después, se relacionan aquellos terminando entre `B` y `E`.

```
$ ls doc[1-3]
doc1 doc2 doc3
$ ls doc[B-E]
docB docD
```

Puede combinar corchetes con otros caracteres de expansión de archivo, para formar operadores de relación flexibles. Suponga que quiere mostrar una lista mostrando sólo nombres de archivo cuya extensión termine con `.c` u `.o`, pero no otra extensión. Puede usar una combinación de asteriscos y corchetes: `*.[co]`. El asterisco relaciona todos los nombres de archivo, mientras los corchetes relacionan sólo archivos con extensión `.c` u `.o`.

```
$ ls *.[co]
principal.c principal.o calc.c
```

Coincidencia de símbolos shell

A veces, un carácter de expansión de archivos es en realidad parte de un nombre de archivo. En tales casos, necesita citar el carácter al colocar una diagonal invertida antes para referir el archivo. En el siguiente ejemplo, el usuario necesita hacer referencia a un archivo terminando con el carácter `?`, ¿respuestas? Sin embargo, el signo `?` es un carácter de expansión de archivo y relacionará cualquier archivo que inicie con “¿respuestas”, poseyendo uno o más caracteres. En este caso, el usuario cita el signo `?` con una diagonal invertida inicial, para referir el archivo.

```
$ ls ¿respuestas\?
¿respuestas?
```

Al colocar el nombre de archivo entre comillas también citará el carácter.

```
$ ls "¿respuestas?"
¿respuestas?
```

Es por igual cierto para nombres de archivo o directorios con espacios en blanco entre caracteres, como el carácter de espacio. En este caso, es posible usar la diagonal invertida para citar el carácter de espacio en el nombre de archivo o directorio, así como colocar todo el nombre entre comillas.

```
$ ls Mis\ documentos
Mis documentos
$ ls "Mis documentos"
Mis documentos
```

Generación de patrones

Aunque no es una operación de expansión de archivo, {} suelen ser útiles para generar nombres susceptibles de usarse para crear o modificar archivos y directorios. La operación de llaves sólo genera una lista de nombres. No busca coincidencias con nombres de archivo existentes. Los patrones se colocan en llaves y separados por comas. Cualquier patrón colocado entre llaves generará una versión del patrón, al emplear el patrón anterior, el siguiente o ambos. Suponga que quiere generar una lista de nombres iniciando con “doc”, pero sólo termine en los patrones “umento”, “final” y “borrador”. Así se hace:

```
$ echo doc{umento,final,borrador}
documento docfinal docborrador
```

Puesto que los nombres generados no tienen porque estar presentes, puede usar la operación {} en un comando para crear directorios, como se muestra aquí:

```
$ mkdir informe{invierno,verano,primavera}
$ ls
informeinvierno informeverano informeprimavera
```

Entrada y salida estándar y redireccionamiento

Los datos en operaciones de entrada y salida se organizan como archivo. La entrada de datos con el teclado se coloca en un flujo de datos ordenados, como un conjunto de bytes continuos. La salida de datos desde un comando o programa también se coloca en un flujo de datos y se ordena como un conjunto de bytes continuos. A esta entrada de flujo de datos en Linux se denomina *entrada estándar*, mientras al flujo de datos de salida se le llama *salida estándar*. También existe un flujo de datos de salida separado, reservado únicamente para mensajes de error, conocidos bajo el nombre *error estándar* (consulte la sección “Redireccionamiento y canalización del error estándar: <&, 2” más adelante, en este capítulo).

Debido a que entrada y salida estándares tienen la misma organización que un archivo, pueden interactuar de manera sencilla con archivos. Linux tiene una capacidad de redireccionamiento permitiendo ingresar y extraer datos de manera sencilla de los archivos. Puede redirigir una salida estándar para que, en vez de desplegar la salida en la pantalla, se almacene en un archivo. También

puede redirigir la entrada estándar de un teclado a un archivo, para que la entrada se lea desde un archivo en vez de su teclado.

Cuando se ejecuta un comando de Linux produciendo salida, ésta se coloca en el flujo de datos de salida estándar. La ubicación predeterminada para este flujo es un dispositivo (en este caso, la pantalla). Los *dispositivos*, como teclado y pantalla, se tratan como archivos. Éstos reciben y mandan flujos de bytes con la misma organización que el flujo de bytes de un archivo. La pantalla es un dispositivo para desplegar un flujo continuo de bytes. Como opción predeterminada, la salida estándar enviará datos al dispositivo de la pantalla, que después desplegará datos.

Por ejemplo, el comando **ls** genera una lista de todos los nombres de archivo y la envía a la salida estándar. Después, este flujo de bytes en la salida estándar se dirige al dispositivo de la pantalla. Esta lista de nombres de archivo entonces se imprime en la pantalla. El comando **cat** también envía la salida a la salida estándar. El contenido de un archivo se copia a la salida estándar, cuyo destino predeterminado es la pantalla. Entonces el contenido del archivo se despliega en la pantalla.

Redireccionamiento de la salida estándar: > y >>

Suponga que en vez de desplegar una lista de archivos en la pantalla, le gustaría guardar esta lista en un archivo. En otras palabras, le gustaría dirigir la salida estándar a un archivo, en lugar de la pantalla. Para ello, necesita colocar el operador de redirección de salida, el signo mayor que (>), seguido por el nombre de un archivo en la línea de comandos, tras el comando de Linux. En la tabla 3-6 se muestra una lista con diferentes maneras en que puede usar los operadores de redirección. En el siguiente ejemplo, la salida del comando **ls** se redirige del dispositivo de pantalla a un archivo:

```
$ ls -l *.c > listaprogramas
```

La operación de redireccionamiento crea el nuevo archivo de destino. Si el archivo ya existe, se sobrescribirá con los datos de la salida estándar. Puede configurar la característica **noclobber** para evitar la sobreescritura de un archivo existente con la operación de redireccionamiento. En este caso, el redireccionamiento a un archivo existente fallará. Puede pasar por alto la característica **noclobber**, colocando un signo de admiración tras el operador de redirección. Puede colocar el comando **noclobber** en un archivo de configuración shell para generar una operación automática predeterminada (consulte el capítulo 5). En el siguiente ejemplo se configura la característica **noclobber** para la shell BASH y después se obliga a sobrescribir el archivo **cartaanterior**, si ya existe:

```
$ set -o noclobber
$ cat micarta >! cartaanterior
```

A pesar de que el operador de redirección y el nombre de archivo se colocan tras el comando, la operación de redireccionamiento no se ejecuta después del comando. En realidad, se ejecuta antes. La operación crea el archivo y configura la redirección, antes de recibir cualquier dato de la salida estándar. Si el archivo ya existe, será destruido y reemplazado por un archivo del mismo nombre. En efecto, el comando que genera la salida sólo se ejecuta después de creado el archivo redirigido.

En el siguiente ejemplo, la salida del comando **ls** se redirige del dispositivo de la pantalla a un archivo. Primero el comando **ls** muestra una lista de archivos y, en el siguiente comando, **ls** redirige su lista al archivo **listarch**. El comando **cat** despliega la lista de archivos guardados en **listarch**. Observe que la lista de archivos en **listarch** incluye el nombre de archivo **listarch**. La lista de nombres de archivos generados por el comando **ls** incluye el nombre del archivo creado por la

Comando	Ejecución
ENTER	Ejecuta una línea de comandos.
;	Separa comandos en la misma línea de comandos.
<i>comando\</i> <i>argumentos opcionales</i>	Inserte una diagonal invertida antes de oprimir ENTER para continuar insertando un comando en la siguiente línea.
<code>`comando`</code>	Ejecuta un comando.
<code>\$(comando)</code>	Ejecuta un comando.
Caracteres especiales para expansión de nombres de archivo	Ejecución
*	Coincide con cualquier conjunto de caracteres.
?	Coincide con cualquier carácter.
[]	Coincide con posibles clases de caracteres.
\	Cita el siguiente carácter. Se utiliza para citar caracteres especiales.
Redirección	Ejecución
<i>comando > nombre de archivo</i>	Redirige la salida estándar a un archivo o dispositivo, creando el archivo si no existe y sobrescribiendo el archivo, si existe.
<i>comando < nombre de archivo</i>	Redirige la entrada estándar desde un archivo o dispositivo a un programa.
<i>comando >> nombre de archivo</i>	Redirige la salida estándar de un archivo o dispositivo, adjuntando la salida al final del archivo.
<i>comando >! nombre de archivo</i>	En las shells C y Korn, el signo de exclamación obliga a sobrescribir un archivo, si ya existe. Esto predomina sobre la opción noclobber .
<i>comando 2> nombre de archivo</i>	Redirige el error estándar a un archivo o dispositivo en la shell Bourne.
<i>comando 2>> nombre de archivo</i>	Redirige y adjunta el error estándar a un archivo o dispositivo en la shell Bourne.
<i>comando 2>&1</i>	Redirige un error estándar a la salida estándar en la shell Bourne.
<i>comando >& nombre de archivo</i>	Redirige el error estándar a un archivo o dispositivo en la shell C.
Canalizaciones	Ejecución
<i>comando comando</i>	Canaliza la salida estándar de un comando como entrada hacia otro comando.
<i>comando & comando</i>	Canaliza el error estándar como entrada de otro comando en shell C.

TABLA 3-6 Las operaciones de shell

operación de redireccionamiento (en este caso, **listarch**). El archivo **listarch** se crea primero por la operación de redireccionamiento y después el comando **ls** muestra el archivo en la lista, junto con otros archivos.

```
$ ls
misdatos intro prefacio
$ ls > listarch
$ cat listarch
misdatos intro listarch prefacio
```

SUGERENCIA Los errores ocurren cuando intenta usar el mismo nombre de archivo para un archivo de entrada para comando y archivo de destino redireccionado. En este caso, debido a que la operación de redireccionamiento se ejecuta primero, puesto que el archivo de entrada existe, se destruye y reemplaza por un archivo con el mismo nombre. Cuando el comando se ejecuta, encuentra un archivo de entrada vacío.

También puede adjuntar la salida estándar a un archivo existente usando el operador de redirección >>. En vez de sobrescribir el archivo, los datos en la salida estándar se agregan al final del archivo. En el siguiente ejemplo, los archivos **micarta** y **cartaanterior** se adjuntaron al archivo **cartascompletas**. El archivo **cartascompletas** entonces incluirá el contenido de ambos archivos, **micarta** y **cartaanterior**.

```
$ cat micarta >> cartascompletas
$ cat cartaanterior >> cartascompletas
```

La entrada estándar

Muchos comandos de Linux pueden recibir datos desde la entrada estándar. La entrada estándar por sí sola recibe datos desde un dispositivo o archivo. El dispositivo predeterminado para la entrada estándar es el teclado. Los caracteres escritos en el teclado se colocan en la entrada estándar, dirigida al comando de Linux. Así como en la salida estándar, también puede redirigir la entrada estándar, recibiendo entrada desde un archivo en vez del teclado. El operador para redirigir la entrada estándar es el signo menor que (<). En el siguiente ejemplo, la entrada estándar se redirige para recibir entrada desde el archivo **micarta**, en vez del dispositivo del teclado (utilice CTRL-D para terminar la entrada escrita). La operación de redireccionamiento lee el contenido de **micarta** en la entrada estándar. El comando **cat** lee la entrada estándar y despliega el contenido de **micarta**.

```
$ cat < micarta
hola Christopher
¿Cómo estás hoy?
$
```

Puede combinar operaciones de redireccionamiento para entrada y salida estándar. En el siguiente ejemplo, el comando **cat** no tiene argumentos de nombre de archivo. Sin los argumentos de nombre de archivo, el comando **cat** recibe la entrada desde la entrada estándar y envía la salida a la salida estándar. Sin embargo, en el siguiente ejemplo, la entrada estándar ha sido redirigida para recibir datos desde el archivo, mientras la salida estándar se ha redirigido para colocar los datos en un archivo.

```
$ cat < micarta > nuevacarta
```

Canalizaciones: |

Tal vez se encuentre en situaciones que necesita enviar datos de un comando a otro. En otras palabras, tal vez quiera enviar la salida estándar de un comando a otro, no a un archivo destino. Suponga que quiere enviar una lista de nombres de archivo a la impresora. Necesita dos comandos

para esto: **ls** para generar una lista de nombres de archivo y **lpr** para enviar la lista a la impresora. En efecto, necesita tomar la salida del comando **ls** y usarla como entrada del comando **lpr**. Considere que los datos son como un flujo de un comando a otro. Para formar dicha conexión en Linux, necesita emplear lo que se denomina *canalización*. El *operador de canalización* (el carácter de barra vertical, |) colocado entre dos comandos, forma una conexión entre éstos. La salida estándar de un comando se convierte en la entrada estándar del otro. El operador de canalización recibe la salida desde el comando colocado antes de la canalización, enviando los datos como entrada al comando colocado tras la canalización. Como se muestra en el siguiente ejemplo, puede conectar los comandos **ls** y **lpr** con un canalización. La lista de nombres de archivo producida por el comando **ls** se canaliza al comando **lpr**.

```
$ ls | lpr
```

Puede combinar la operación de **canalización** con otras características shell, como caracteres de expansión de archivo, para realizar operaciones especializadas. En el siguiente ejemplo se imprimen sólo los archivos con extensión **.c**. El comando **ls** se usa con el asterisco y **" .c "** para generar una lista de archivos con la extensión **.c**. Después la lista se canaliza al comando **lpr**.

```
$ ls *.c | lpr
```

En el ejemplo anterior, se ha empleado una lista de nombres de archivo como entrada, pero es importante observar que las canalizaciones operan en la salida estándar de un comando, cualquiera que sea. El contenido de todos los archivos, incluso varios archivos, pueden canalizarse de un comando a otro. En el siguiente ejemplo, el comando **cat** lee y saca el contenido del archivo **misdatos**, después canalizado al comando **lpr**:

```
$ cat misdatos | lpr
```

Linux tiene muchos comandos que generan una salida modificada. Por ejemplo, el comando **sort** toma el contenido de un archivo y genera una versión, en el que cada línea es ordenada alfabéticamente. El comando **sort** funciona mejor con archivos conteniendo listas de elementos. A los comandos como **sort**, que producen una salida modificada de su entrada, se les conoce como *filtros*. Los filtros suelen usarse con canalizaciones. En el siguiente ejemplo, se genera una versión ordenada de **milista**, para canalizarse al comando **more** y así desplegarse en pantalla. Observe que el archivo original, **milista**, no ha cambiado ni se ordena por sí mismo. Sólo se ordena la salida **sort** en la salida estándar.

```
$ sort milista | more
```

La entrada estándar canalizada en un comando puede controlarse con más cuidado a través del argumento de entrada estándar (**-**). Cuando usa un guión como argumento para un comando, representa la entrada estándar.

Redirección de un error estándar: **2>**, **>>**

Al ejecutar comandos, podría ocurrir un error. Tal vez dé el número equivocado de argumentos o se presente algún tipo de error del sistema. Cuando ocurre un error, el sistema envía un mensaje de error. Generalmente, dicho mensaje de error se despliega en la pantalla, junto con la salida estándar. Sin embargo, Linux distingue entre salida estándar y mensajes de error.

Los mensajes de error se colocan en otro flujo de datos estándar, denominado *error estándar*. En el siguiente ejemplo, al comando **cat** se da como argumento el nombre de un archivo que no existe, **miintro**. En este caso, el comando **cat** sólo envía el error:

```
$ cat miintro
cat : miintro not found
$
```

Debido a que los mensajes de error se encuentran en un flujo de datos separados de la salida estándar, aparecen en la pantalla para que los vea, aunque haya redirigido la salida estándar a un archivo. En el siguiente ejemplo, la salida estándar del comando **cat** es redirigida al archivo **misdatos**. Sin embargo, el error estándar, conteniendo los mensajes de error, todavía se dirige a la pantalla.

```
$ cat miintro > misdatos
cat : miintro not found
$
```

Puede redirigir el error estándar, como haría con la salida estándar. Esto significa que puede guardar sus mensajes de error en un archivo para referencias futuras. Es útil si necesita un registro de mensajes de error. Al igual que la salida estándar, el error estándar tiene el dispositivo de pantalla como destino predeterminado. Sin embargo, puede redirigir el error estándar a cualquier archivo o dispositivo a elegir, usando operadores de redirección. En este caso, el mensaje de error no se desplegará en la pantalla.

La redirección del error estándar depende de una característica especial de redireccionamiento de la shell. Puede hacer referencia a todos los flujos de byte estándar en las operaciones de redirección con números. Los números 0, 1 y 2 refieren entrada estándar, salida estándar y error estándar, respectivamente. Como opción predeterminada, una redirección de salida, **>**, opera en la salida estándar 1. Sin embargo, puede modificar la redirección de salida para operar en un error estándar, anteponiendo el número 2 al operador de redirección. En el siguiente ejemplo, el comando **cat** generará un error nuevamente. El mensaje de error se redirige al flujo de bytes estándar representado por el número 2, el error estándar.

```
$ cat nodatos 2> miserrores
$ cat miserrores
cat : nodatos not found
$
```

También puede adjuntar el error estándar a un archivo usando el número 2 y el operador de redirección adjunto (**>>**). En el siguiente ejemplo, el usuario adjunta el error estándar al archivo **miserrores**, que después funciona como registro de errores:

```
$ cat nodatos 2>> miserrores
```

Trabajos: segundo plano, cancelaciones e interrupciones

En Linux, no sólo tiene control sobre entrada y salida del comando, también sobre su ejecución. Puede realizar un trabajo en segundo plano conforme ejecuta otros comandos. También puede cancelarlos antes de que terminen de ejecutarse. Incluso puede interrumpir un comando, para volver a iniciarlo después, desde donde se quedó. Las operaciones en segundo plano son muy útiles para

trabajos largos. En vez de esperar en la terminal hasta que el comando termine su ejecución, puede colocarlo en segundo plano. Después puede continuar la ejecución de otros comandos de Linux. Por ejemplo, editar un archivo mientras se imprimen otros. Los comandos de segundo plano, además de aquellos para cancelar e interrumpir trabajos, se muestran en la lista de la tabla 3-7.

Ejecución de trabajos en segundo plano

Puede ejecutar un comando en segundo plano colocando el signo `&` en la línea de comandos, al final de la instrucción. Cuando coloca un trabajo en segundo plano, se despliega un número de trabajo de usuario, así como un número de proceso de sistema. El número de trabajo de usuario, colocado entre corchetes, es el número con que el usuario puede aludir ese trabajo. El número de proceso de sistema es con que el usuario puede identificar el trabajo. En el siguiente ejemplo, el comando para imprimir el archivo `misdatos` está colocado en segundo plano:

```
$ lpr misdatos &  
[1] 534  
$
```

Trabajos en segundo plano	Ejecución
<code>%númdetrabajo</code>	Hace referencia al trabajo por un número de trabajo, utilice el comando <code>jobs</code> para desplegar números de trabajo.
<code>%</code>	Hace referencia al trabajo reciente.
<code>%cadena</code>	Hace referencia al trabajo por una cadena que concuerde exactamente.
<code> \$?cadena?</code>	Hace referencia al trabajo que contiene la cadena de comandos.
<code>%--</code>	Hace referencia al penúltimo trabajo.
<code>&</code>	Ejecuta un comando en segundo plano.
<code>fg %númdetrabajo</code>	Lleva un comando del segundo plano al frente o reanuda un programa interrumpido.
<code>bg</code>	Coloca un comando al frente en segundo plano.
<code>CTRL-Z</code>	Interrumpe y detiene el programa actualmente en ejecución. El programa permanece detenido y en espera en segundo plano para ser reanudado.
<code>notify %númdetrabajo</code>	Notifica cuando un trabajo termina.
<code>kill %númdetrabajo</code> <code>kill númdeproceso</code>	Cancela y termina un trabajo ejecutándose en segundo plano.
<code>jobs</code>	Muestra una lista de todos los trabajos en segundo plano.
<code>ps -a</code>	Muestra una lista de los procesos actualmente en ejecución, incluidos aquellos en segundo plano.
<code>at hora fecha</code>	Ejecuta comandos en una hora y fecha específica. El tiempo puede insertarse con horas y minutos, y habilitarse como AM O PM.

TABLA 3-7 Operaciones de administración de trabajo

Puede colocar más de un comando en segundo plano. Cada uno se clasifica como trabajo, recibiendo nombre y número de trabajo. El comando **jobs** muestra una lista de trabajos ejecutándose en segundo plano. Cada entrada en la lista consta de un número de trabajo entre corchetes (ya sea que esté detenido o en ejecución) y el nombre del trabajo. El signo **+** indica el trabajo que se está procesando y el signo **-** indica el trabajo a ejecutar después. En el siguiente ejemplo, se han colocado dos comandos en segundo plano. El comando **jobs** muestra entonces la lista de esos trabajos, indicando cuál está en ejecución.

```
$ lpr intro &
[1] 547
$ cat *.c > misprogramas &
[2] 548
$ jobs
[1] + Running lpr intro
[2] - Running cat *.c > misprogramas
$
```

Referencia a trabajos

En general se hace referencia a trabajos usando el número de trabajo, con un símbolo **%** antes. Puede obtener este número con el comando **jobs**, que mostrará una lista de todos los trabajos en segundo plano, como se presenta en el ejemplo anterior. Además, puede referir un trabajo al usar una cadena de identificación (véase la tabla 3-7). La cadena de comandos debe ser una coincidencia exacta o parcial única. Si no existe una coincidencia exacta o única, recibirá un mensaje de error. Además, el símbolo **%** por sí sólo, sin número de trabajo, hace referencia al trabajo actual en segundo plano. Seguido por **--**, refiere el segundo trabajo en segundo plano previo. En el siguiente ejemplo, se lleva el trabajo 1 del ejemplo anterior al frente.

```
fg %lpr
```

Notificación de trabajo

Después de ejecutar cualquier comando en Linux, el sistema indica cuáles trabajos se han completado en segundo plano, si tiene alguno, hasta ahora. El sistema no interrumpe ninguna operación, como la edición, para notificarle acerca de un trabajo completo. Si quiere que se le notifique de inmediato respecto al término de una tarea, no importa que esté haciendo en el sistema, puede usar el comando **notify** para instruir al sistema que le avise. El comando **notify** toma el número de trabajo como argumento. Cuando un trabajo termina, el sistema interrumpe lo que está haciendo para notificarle que el trabajo ha terminado. En el siguiente ejemplo se le dice al sistema que notifique al usuario cuando el trabajo 2 termine:

```
$ notify %2
```

Paso de trabajos al frente

Puede traer un trabajo del segundo plano con el comando, **fg**. Si sólo hay un trabajo en segundo plano, bastará el comando **fg** para pasarlo al frente. Si más de un trabajo está en segundo plano, deberá usar el nombre del trabajo con el comando. Debe colocar el número de trabajo luego del comando **fg**, con un signo por ciento (%) antes. Un comando **bg**, generalmente usado para interrumpir trabajos, coloca un trabajo en segundo plano. En el siguiente ejemplo, el segundo trabajo es llevado al frente. Tal vez no reciba un aviso inmediato otra vez, debido a que el segundo

comando ahora está adelante y en ejecución. Cuando el comando termina de ejecutarse, aparece el indicador de comandos y entonces puede proceder con otro comando.

```
$ fg %2
cat *.c > misprogramas
$
```

Cancelación de trabajos

Si quiere cancelar un trabajo ejecutándose en segundo plano, puede forzarlo a que termine con el comando **kill**. Este toma como argumento, ya sea el número de trabajo del usuario o el número de proceso de sistema. El número de trabajo del usuario debe antecederse con un signo de porcentaje (%). Encontrará el número de trabajo con el comando **jobs**. En el siguiente ejemplo, el comando **jobs** muestra una lista de trabajos en segundo plano; después el trabajo 2 se cancela:

```
$ jobs
[1] + Running lpr intro
[2] - Running cat *.c > misprogramas
$ kill %2
```

Suspensión y detención de trabajos

Puede suspender un trabajo y detenerlo con las teclas CTRL-Z. Esto coloca el trabajo de lado hasta que se reinicie. El trabajo no está terminado; sólo permanece suspendido hasta que quiera continuar. Cuando esté listo, puede continuar con el trabajo al frente o en segundo plano al utilizar el comando **fg** o **bg**. El comando **fg** reinicia un trabajo suspendido al frente. El comando **bg** coloca el trabajo suspendido en segundo plano.

Algunas veces, tal vez necesite colocar en segundo plano un trabajo ejecutándose al frente. Sin embargo, no puede mover el trabajo en ejecución directamente al segundo plano. Primero debe suspenderlo con CTRL-Z y después colocarlo en segundo plano con el comando **bg**. En el siguiente ejemplo, primero se suspende con CTRL-Z el comando actual para mostrar una lista y redirigir los archivos **.c**. Después, el trabajo se envía a segundo plano.

```
$ cat *.c > misprogramas
^Z
$ bg
```

NOTA También puede usar ctrl-z para detener trabajos en ejecución como Vi, suspendiéndolos en segundo plano hasta estar listo para reanudarlos. La sesión Vi permanece como un trabajo detenido en segundo plano hasta reanudarlo con el comando bg.

Finalizar procesos: ps y kill

También puede cancelar un trabajo usando el número de proceso de sistema, que puede obtener con el comando **ps**. El comando **ps** desplegará sus procesos y puede usar un número de proceso para finalizar cualquier proceso en ejecución. El comando **ps** despliega mucha información, más que el comando **jobs**. En el siguiente ejemplo se muestra una lista de procesos ejecutándose por un usuario. PDI es el número de proceso de sistema, también conocido como ID de proceso. TTY es el identificador de terminal. Time es cuánto tiempo lleva el proceso hasta ahora. COMMAND es el nombre del proceso.

```
$ ps
PID      TTY      TIME    COMMAND
523      tty24    0:05    sh
567      tty24    0:01    lpr
570      tty24    0:00    ps
```

Después puede hacer referencia al número de proceso de sistema en el comando **kill**. Use el número de proceso sin signo de porcentaje antes. En el siguiente ejemplo se elimina el proceso 567:

```
$ kill 567
```

Consulte la página Man de **ps**, para información más detallada acerca de la detección y despliegue de información de procesos. Para desplegar el número PID, use la opción de salida **-o pid=**. Combinado con la opción de comando **-c**, puede desplegar el PID de un comando en particular. Si existe más de un proceso para ese comando, como varias shells bash, entonces todos los PID se desplegarán.

```
$ ps -C lpr -o pid=
567
```

En el caso de comandos únicos, los que sabe tienen sólo un proceso en ejecución, puede combinar de manera segura el comando previo con **kill**, para terminar el proceso en una línea. Esto evita desplegar e insertar de manera interactiva el PID para finalizar un proceso. La técnica puede ser útil para operaciones no interactivas como **cron** (consulte el capítulo 27) y para terminar operaciones indefinidas, como la grabación de un video. En el siguiente ejemplo, un comando que emplea sólo un proceso, **getatse**, se termina en un sola operación **kill**. El comando **getatssc** es un comando de grabación de hdtv. Las comillas invertidas se utilizan para ejecutar el primer comando **ps** y obtener el PID (consulte “Valores de comandos de Linux”, en el capítulo 4).

```
kill `ps -C getatssc -o pid=`
```

La shell C: edición e historial de la línea de comandos

La shell C se desarrolló originalmente para usarse en BSD Unix. Con Linux, está disponible como shell alternativa, junto con Korn y Bourne. shell C incorpora todos los comandos raíz usados en Bourne, pero difiere significativamente en características más complejas, como la programación de shell. La shell C fue desarrollada tras Bourne y la primera en introducir nuevas características como edición en la línea de comandos y la utilidad historial. La shell Korn después incorporó muchas de las mismas características. Asimismo, la shell bash, a cambio, incorporó muchas características de todas estas shell. Sin embargo, las respectivas implementaciones difieren significativamente. La shell C tiene ediciones de línea de comandos limitadas que permiten realizar algunas operaciones de edición básicas. La edición de línea de comandos de shell C no es tan poderosa como Korn. La utilidad de historial permite ejecutar y editar comandos previos. La utilidad historial trabaja de manera muy similar a Korn, BASH, Z y C. Sin embargo, sus nombres de comandos difieren radicalmente, la shell C tiene un conjunto de operaciones de edición de historial muy diferente.

En casi todas las distribuciones de Linux, se utiliza una versión mejorada de la shell C, denominada TCSH. Casi todos los comandos son similares. Puede acceder a la shell C con el comando **csh**, un vínculo a la shell TCSH. El indicador de comandos tradicional para la shell C es

el símbolo %. En algunas distribuciones de Linux el indicador de comandos puede permanecer con \$, sin cambio.

```
$ csh
%
```

El comando para entrar a la shell TCSH es **tcsh**.

Edición en la línea de comandos shell C

Al igual que BASH, la shell C sólo tiene límites en capacidades de edición de la línea de comandos. Sin embargo, son más poderosas que las de Bourne. En vez de eliminar sólo un carácter, puede eliminar toda la palabra. También puede realizar operaciones de edición, limitadas al uso de substitución de patrones.

La tecla CTRL-W elimina una palabra insertada recientemente. El término “palabra” es más un concepto técnico denotando la manera en que la shell analiza un comando. Una palabra se analiza hasta un espacio o tabulación. Cualquier carácter o conjunto de caracteres rodeados de espacios o tabulaciones se considera una palabra. Con CTRL-W puede eliminar texto insertado, de palabra en palabra.

```
% date quién
% date
```

Otras veces, tal vez necesite cambiar parte de una palabra o varias palabras en una línea de comando. La shell C tiene un comando de substitución de patrones para reemplazar patrones en la línea de comandos. Este comando de substitución se representa con un patrón encerrado entre los símbolos ^. El patrón a reemplazar se encierra entre dos ^. El texto reemplazado sigue inmediatamente.

```
% ^patrón^nuevotexto
```

La operación de substitución de patrón no es sólo un comando de edición. También es de ejecución. En vez de reemplazar el patrón, el comando correcto se desplegará y después ejecutará. En el siguiente ejemplo, el comando date se ha escrito mal. La shell despliega un mensaje de error diciendo que no pudo encontrarse el comando. Puede editarlo usando los símbolos ^ para reemplazar el texto incorrecto. Entonces el comando se ejecuta.

```
% dte
dte: not found
% ^dt^dat
date
Sun July 5 10:30:21 PST 1992
%
```

Utilería de historial de la shell C

Como en BASH, la utilidad de historial de la shell C mantiene un registro de comandos más recientes ejecutados. En la tabla 3-8 se muestra una lista de comandos del historial de la shell C. La utilidad del historial lleva el seguimiento de un número limitado de comandos más recientes, numerados desde 1. Historial no está activada automáticamente. Primero debe definir el historial con un comando **set** y asignarle el número de comandos que desea registrar. Esto, generalmente,

Referencias a eventos de shell C	
<code>!event num</code>	Hace referencia a un evento por su número de evento.
<code>!caracteres</code>	Hace referencia a un evento que comienza con caracteres especificados.
<code>!?patrón?</code>	Hace referencia a un evento que contiene el patrón específico.
<code>!-núm evento</code>	Hace referencia a un evento con un desplazamiento desde el primer evento.
<code>!núm-núm</code>	Hace referencia a un rango de eventos.
Referencias a palabra de evento de shell C	
<code>!núm evento:núm palabra</code>	Hace referencia a una palabra en particular en un evento.
<code>!núm evento:^</code>	Hace referencia al primer argumento (la segunda palabra) en un evento.
<code>!núm evento:\$</code>	Hace referencia al último argumento en el evento.
<code>!núm evento:^-</code>	Hace referencia a todos los argumentos en el evento.
<code>!núm evento:*</code>	Hace referencia a todos los argumentos en el evento.
Sustitución de edición de evento de shell C	
<code>!núm evento:s/patrón/nuevotexto/</code>	Edita un evento con una sustitución de patrón. Hace referencia a una palabra particular en un evento.
<code>!núm evento:sg/patrón/nuevotexto/</code>	Realiza una sustitución global en todas las instancias de un patrón en el evento.
<code>!núm evento:s/patrón/nuevotexto/p</code>	Suprime la ejecución de un evento editado.

TABLA 3-8 Comandos de historial de la shell C

se hace como parte de su configuración shell. En este ejemplo, la utilería historial se define y configura para recordar los últimos cinco comandos.

```
% set history=5
```

Como en la shell BASH, se hace referencia a comandos recordados como eventos. Para ver el conjunto de sus eventos más recientes, inserte la palabra **history** en la línea de comandos y oprima ENTER. Se desplegará una lista de comandos recientes, con un número antes de cada evento.

```
% history
1 ls
2 vi misdatos
3 mv misdatos informes
4 cd informes
5 ls -F
```

Se puede hacer referencia a cada uno de estos eventos por su número, el comienzo de los caracteres de un evento o un patrón de caracteres en el evento. Una referencia al patrón se encierra entre signos de interrogación, ? Puede volver a ejecutarlos utilizando el comando de historial !.

Los signos de exclamación son seguidos por una referencia al evento, como un número de evento, el comienzo de los caracteres o un patrón. En el siguiente ejemplo, primero se hace referencia al segundo comando en la lista del historial por su número de evento, después por el comienzo de los caracteres de un evento y luego por un patrón en el evento.

```
% !2
vi misdatos

% !vi
vi misdatos

% !?misd?
vi misdatos
```

También puede hacer referencia a un comando mediante un desplazamiento desde el final de la lista. Al poner un signo menos antes del número de desplazamiento, desde el final de la lista a ese comando. En el siguiente ejemplo, se hace referencia al segundo comando, **vi misdatos**, usando un desplazamiento.

```
% !-4
vi misdatos
```

Un signo de admiración también se utiliza para identificar el último comando ejecutado. Es equivalente a un desplazamiento de -1. En los siguientes ejemplos, ambos, el desplazamiento de 1 y el signo de admiración, hacen referencia al último comando, **ls -F**.

```
% !!
ls -F
misdatos /informes

% !-1
ls -F
misdatos /informes
```

Sustituciones de eventos del historial en la shell C

Una referencia a un evento debe considerarse como una representación de caracteres creando el evento. La referencia a evento **!1** representa realmente los caracteres "ls". Como tal, puede usar una referencia a un evento como parte de otro comando. La operación del historial puede pensarse como una sustitución. Los caracteres creando el evento reemplazan los signos de admiración y referencias de eventos insertadas en la línea de comandos. En el siguiente ejemplo, la lista de eventos se despliega primero. Después, una referencia al primer evento se usa como parte de un nuevo comando. La referencia al evento **!1** evalúa a **ls**, que se vuelve parte del comando **ls > misarchivos**.

```
% history
1 ls
2 vi misdatos
3 mv misdatos informes
4 cd informes
5 ls -F

% !1 > misarchivos
ls > misarchivos
```

También puede referir palabras en particular en un evento. Este se analiza en palabras separadas; cada palabra es identificada en secuencia por un número, a partir del 0. Una referencia a evento, seguida por dos puntos y un número, refiere una palabra en el evento. La mención al evento **13:2** hace referencia a la segunda palabra en el tercer evento. Primero indica el tercer evento, **mv misdatos informes**, y la segunda palabra en ese evento **misdatos**. Puede utilizar tales referencias de palabra como parte de un comando. En el siguiente ejemplo, **2:0** hace referencia a la primera palabra en el segundo evento, **vi**, y los reemplaza con prefacio.

```
% !2:0 prefacio
vi prefacio
```

Al utilizar un rango de números, puede hacer referencia a varias palabras en un evento. Los números de la primera y última palabra en el rango se separan por un guión. En el siguiente ejemplo, **3:0-1** hace referencia a las dos primeras palabras del tercer evento, **mv misdatos**.

```
% !3:0-1 cartasanteriores
```

Los caracteres meta **^** y **\$** representan segunda y última palabras de un evento. Se utilizan para referir argumentos del evento. Si sólo necesita el primer argumento de un evento, entonces **^** hace referencia a éste; **\$** hace referencia al último argumento. El rango **^-\$** hace referencia a todos los argumentos. (La primera palabra, el nombre de comando, no se incluye.) En el siguiente ejemplo, se refieren y usan como argumentos los utilizados en eventos anteriores en el comando actual. Para empezar, el primer argumento (la segunda palabra) del segundo evento, **misdatos**, se utiliza como un argumento en un comando **lp**, para imprimir un archivo. Después, el último argumento en el tercer evento, **informes**, se usa como argumento en el comando **ls**, para mostrar una lista de nombres de archivos en informes. Por último, los argumentos usados en el tercer evento **misdatos** e **informes**, se manejan como argumentos en el comando **copy**.

```
% lpr !2:^
lpr misdatos

% ls !3:$
ls informes

% cp !3:^-$
cp misdatos informes
```

El asterisco es un símbolo especial representando todos los argumentos en un comando antiguo. Es equivalente al rango **^-\$**. El último ejemplo puede reescribirse usando el asterisco, **!3***.

```
% cp !3*
cp misdatos informes
```

En la shell C, cuando un signo de admiración se utiliza en un comando, se interpreta como referencia de comando del historial. Si necesita un signo de admiración por otras razones, como símbolo de una dirección de correo electrónico, debe citar el signo de admiración colocando una diagonal invertida antes de éste.

```
% mail garnet\!chris < misdatos
```

Edición de eventos del historial en la shell C

Puede editar los comandos del historial con un comando de sustitución. Éste opera de la misma forma que el comando `^` para edición de línea de comandos. Coloca un patrón en un comando con el mismo texto. Para cambiar un comando de historial específico, inserte un signo de admiración y el número de evento de ese comando, seguido por dos puntos y el comando de sustitución. Éste último empieza con un carácter `s`, seguido por un patrón encerrado entre dos diagonales. El texto de reemplazo sigue inmediatamente, terminando con una diagonal.

```
% !num:s/patrón/nuevotexto/
```

En el siguiente ejemplo, el patrón “mis” en el tercer evento es cambiado por “tus”. Después se despliega el evento cambiado y se ejecuta.

```
% history
1 ls
2 vi misdatos
3 mv misdatos informes
4 cd informes
5 ls -F
% !3:s/mis/tus/
mv tusdatos informes
%
```

Al poner una `g` antes del comando `s` se realizará la sustitución global de un evento. Todas las instancias del patrón en el evento cambiarán. En el siguiente ejemplo, la extensión de cada nombre de archivo, en el primer evento, cambiarán de `.c` a `.p` y después se ejecutarán.

```
% lpr calc.c lib.c
% !1:gs/.c/.p/
lpr calc.p lib.p
%
```

El comando `&` repetirá la sustitución previa. En el siguiente ejemplo la misma sustitución se realiza en dos comandos, al cambiar el nombre de archivo **misdatos** a **tusdatos** en el segundo y tercer eventos.

```
% !3:s/mis/tus/
mv tusdatos informes
% !2:&
vi tusdatos
```

Cuando realiza una operación de historial en un comando, ésta se ejecuta automáticamente. Después, puede suprimir la ejecución con el calificador `p`. Este sólo desplegará el comando modificado, no lo ejecutará. Proceder así permite realizar varias operaciones en un comando antes de ejecutarlo. En el siguiente ejemplo, dos comandos de sustitución se efectúan en el tercer comando antes de ejecutarse.

```
% !3:s/mv/cp/:p          No ejecuta el comando
cp misdatos informes
% !3:s/informes/libros/  Cambia y ejecuta el comando
cp misdatos libros
%
```

La shell TCSH

La shell TCSH es, en esencia, una versión de la shell C, con características añadidas. Es totalmente compatible con la shell C estándar e incorpora todas las capacidades, incluido el lenguaje shell y la utilería historial. TCSH tiene características de edición de línea de comandos e historial más avanzadas que las encontradas en la shell C original. Puede usar combinaciones de teclas Vi o Emacs para editar comandos o eventos de historial. La shell TCSH también soporta el remate de líneas de comandos, llenando un comando al utilizar sólo los primeros caracteres que escriba. La shell TCSH tiene soporte para lenguaje nativo, administración de terminal extensa, nuevos comandos incluidos y variables de sistema. Consulte la página man de TCSH para conocer información más detallada.

Terminación de palabras en línea de comandos de TCSH

La línea de comandos tiene una característica integrada que completa comandos y nombres de archivos. Si inserta un patrón incompleto, como un argumento de nombre de archivo, puede oprimir la tecla TAB para activar esta característica, que completará el patrón para generar un nombre de archivo. Para utilizar esta característica, escriba el nombre parcial de un archivo en la línea de comandos y oprima TAB. La shell buscará automáticamente el archivo con el prefijo parcial y completará los caracteres faltantes en la línea de comandos por usted. En el siguiente ejemplo, el usuario envía el comando **cat** con un nombre de archivo incompleto. Cuando el usuario oprime TAB, el sistema buscará coincidencias; después de encontrar una, rellena el nombre de archivo.

```
> cat pre TAB
> cat prefacio
```

Si más de un archivo tiene el mismo prefijo, la shell encontrará el nombre hasta donde los nombres de archivo acepten y después hará beep. Puede entonces agregar más caracteres o seleccionar uno u otro.

Por ejemplo:

```
> ls
documento docudrama
> cat doc TAB
> cat docu beep
```

Si, en vez de eso quiere una lista de todos los nombres con los que su archivo incompleto concuerde, puede oprimir CTRL-D en la línea de comandos. En el siguiente ejemplo, al oprimir CTRL-D después del nombre de archivo, se genera una lista de posibles nombres de archivo.

```
> cat doc Ctrl-d
documento
docudrama
> cat docu
```

La shell vuelve a dibujar la línea de comandos, y puede escribir entonces el resto del nombre de archivo, caracteres diferentes u oprimir TAB para que el nombre de archivo se complete.

```
> cat docudrama
```

Edición de historial TCSH

Como en la shell C, la utilería de historial de la shell TCSH mantiene un registro de comandos recientes ya ejecutados. La utilería historial es como un tipo de memoria a corto plazo, llevando

seguimiento de un número limitado de comandos recientes. La utilidad `history` le permite referir a un evento antiguo colocándolo en la línea de comandos, permitiendo ejecutarlo. Sin embargo, no necesita desplegar primero la lista con el historial. La manera más sencilla de hacer esto es usar las teclas `FLECHA HACIA ARRIBA` o `FLECHA HACIA ABAJO` para colocar los eventos del historial en su línea de comandos, de uno en uno. Oprimir la tecla de nuevo coloca el siguiente evento del historial en su línea de comandos. La `FLECHA HACIA ABAJO` colocará el siguiente comando en la línea de comandos.

También es posible editar en la línea de comandos. Las teclas `FLECHA HACIA ARRIBA` o `FLECHA HACIA ABAJO` lo llevan por la línea de comandos. Después, puede insertar el texto donde sea que deje el cursor. Con las teclas `RETROCESO` y `SUPR` se eliminan caracteres. `CTRL-A` mueve su cursor al principio de la línea de comandos y `CTRL-E` al final. `CTRL-K` elimina los sobrantes de la línea desde la posición del cursor y `CTRL-U` elimina toda la línea.

La shell Z

La shell Z incluye todas las características de la shell Korn, agregando características de línea de comando y eventos de historial. La shell Z realiza expansiones automáticas en la línea de comandos, después de efectuar el análisis sintáctico. Las expansiones se realizan en nombres de archivos, procesos, parámetros, comandos, expresiones aritméticas, corchetes y generación de nombres de archivo.

La shell Z soporta el uso de combinaciones de teclas Vi y Emacs para referir eventos del historial, muy parecido a como hace BASH. `FLECHA HACIA ARRIBA` y `CTRL-P` lo llevan al evento anterior, mientras `FLECHA HACIA ABAJO` y `CTRL-N` lo llevan abajo, al siguiente. `ESC <` lo dirige al primer evento y `ESC >` lo lleva al último evento. `FLECHA A LA DERECHA` y `FLECHA A LA IZQUIERDA` lo conducen a lo largo de la línea de eventos. `CTRL-R` `CTRL-X` efectúa una búsqueda de eventos del historial.

También puede hacerse referencia a eventos del historial usando el símbolo `!`, de manera muy similar a como se hace en el historial de la shell C. Cuando inserta el comando del historial, una lista de comandos previos (llamados eventos) se desplegará, cada uno con un número. Para hacer referencia a un evento, ingrese el símbolo `!` y su número. En el siguiente ejemplo se hace referencia al tercer evento.

```
! 3
```

Hay más de una forma para hacer referencia a un evento. Puede usar un desplazamiento del comando actual, emplear un patrón para identificar un evento o especificar el comienzo de los caracteres de un evento. En la tabla 3-9 se muestra una lista de estas opciones.

Puede manejar designadores de palabra para incluir sólo segmentos de un evento de historial en su comando. Un designador de palabra indica qué palabra o palabras de un comando dado en la línea de comandos, se incluirá en la referencia del historial. Dos puntos separan el número de evento desde el designador de palabra. Puede omitirse si éste empieza con `^`, `$`, `*`, `-` o `%`. Las palabras se numeran desde el 0; esto hace referencia a la primera palabra de un evento y 1 a la segunda palabra. `$` refiere la última palabra. `^` indica el primer argumento, la primera palabra después de la palabra de comando (igual a 1). Puede hacer referencia a un rango de palabras o, con `*`, las palabras sobrantes de un evento. Para referir todas las palabras desde la tercera hasta el final, use `3*`. El `*`, por sí solo, indica todos los argumentos (desde 1). En el siguiente ejemplo se hace referencia a la segunda, tercera y cuarta palabra del sexto evento.

```
! 6:2-4
```

Comandos de historial de la shell Z	
!	Inicia una sustitución de historial, excepto cuando va después de un espacio, nueva línea, = o (.
!!	Hace referencia a comandos previos. Por sí solo, repite el comando previo.
!núm	Refiere el <i>número</i> de línea de comandos.
!-núm	Hace referencia a la línea de comandos actual menos número.
!cad	Hace referencia a los comandos más recientes empezando con cad.
!?cad[?]	Hace referencia a comandos recientes conteniendo cad.
!#	Hace referencia a la línea de comandos escrita hasta el momento.
!{...}	Aisla una referencia de historial a partir de caracteres adyacentes (si es necesario).
Designadores de palabra de la shell Z	
0	La primera palabra de entrada (comando).
núm	El enésimo argumento.
^	El primer argumento (es decir, 1).
\$	El primer argumento (es decir, 1).
%	La palabra coincidente con la búsqueda ?cad (la más reciente).
cad-cad	Un rango de palabras; -cad abrevia 0-cad.
*	Todos los argumentos o un valor nulo, si sólo existe una palabra en el evento.
cad*	Abrevia cad-\$
cad-	Cómo cad* pero omite la palabra \$.

TABLA 3-9 Historial de la shell Z

Secuencias de comandos y programación de la shell

Las secuencias de comandos de la shell combinan comandos de Linux de manera tal que realizan tareas específicas. Los diferentes tipos de shell proporcionan muchas herramientas de programación para crear programas de shell. Puede definir variables y asignar valores a éstas. También definir variables en un archivo de secuencia de comandos y hacer que un usuario inserte valores de manera interactiva para éstos cuando ejecute la secuencia de comandos. La shell proporciona bucles y estructuras de control condicionales que repiten expresiones de construcción realizando operaciones aritméticas o de comparación. Todas estas herramientas de programación operan en forma similar a las encontradas en otros lenguajes de programación, de modo que si está familiarizado con la programación, puede encontrar que la programación en shell es fácil de aprender.

Las shells BASH, TCSH y Z, descritas en el Capítulo 3, son tipos de shell. Puede tener muchos ejemplos de tipos particulares de shell. Una *shell*, por definición, es un entorno interpretativo en el que puede ejecutar comandos. Puede tener diferentes entornos ejecutándose simultáneamente, ya sean los mismos o diferentes tipos de shell; por ejemplo, puede tener varias shells ejecutándose al mismo tiempo del tipo BASH.

En este capítulo, se cubrirán las bases para crear un programa en shell utilizando BASH y TCSH, las empleadas en casi todos los sistemas Linux. Aprenderá a crear sus propias secuencias de comandos, a definir variables shell y desarrollar interfaces de usuario, además de tareas más difíciles de combinación de estructuras de control para crear programas complejos. En las tablas del capítulo se mostrarán listas de comandos y operadores de shell, igualmente, numerosos ejemplos que muestran cómo implementarlas.

En general, las instrucciones para crear un programa de shell se insertan en un archivo de secuencia de comandos que después puede ejecutarse. Incluso puede distribuir su programa junto con varios archivos de secuencia de comandos, uno de los cuales contendrá instrucciones para ejecutar otros. Puede pensar en variables, expresiones y estructuras de control como herramientas que puede utilizar para juntar varios comandos Linux en una operación. En este sentido, un programa de shell es un nuevo y complejo comando creado en Linux.

La shell BASH tiene un conjunto de comandos de programación flexibles y poderosos para construir secuencias de comandos complejos. Soporta variables tanto locales en la shell dada, como exportadas a otras shell. Puede pasar argumentos de una secuencia de comandos a otra. La shell

BASH tiene un conjunto completo de estructuras de control, incluidos bucles y afirmaciones `if`, además de estructuras `case`, de las que aprenderá conforme lea este libro. Todos los comandos de shell interactúan de manera sencilla con operaciones de redireccionamiento y canalización que permiten alimentación de una entrada estándar o enviarla a una salida estándar. A diferencia de la shell Bourne, la primera shell utilizada para Unix, BASH incorpora muchas características de las shell TCSH y Z. Las operaciones aritméticas en particular son más sencillas en BASH.

La shell TCSH, al igual que BASH, también tiene capacidades para lenguaje de programación. Puede definir variables y asignar valores a éstas. Puede colocar definiciones de variables y comandos Linux en un archivo de secuencia de comandos y después ejecutar esa secuencia. Puede usar un bucle y estructuras de control condicional para repetir comandos de Linux o tomar decisiones sobre cuáles comandos quiere ejecutar. También puede colocar trampas en su programa para manejar interrupciones.

La shell TCSH difiere de otras en que sus estructuras de control se ajustan más a un formato de lenguaje de programación. Por ejemplo, la condición de prueba para una estructura de control de la shell TCSH es una expresión arrojando como resultado cierto o falso, no un comando de Linux. Una expresión de shell TCSH usa los mismos operadores encontrados en el lenguaje de programación C. Puede realizar diversas operaciones de asignación, aritméticas, relacionales y de bits. La shell TCSH también permite declarar variables numéricas para usar de manera sencilla en dichas operaciones.

Variables de shell

En cada shell, puede insertar y ejecutar comandos. Puede incrementar las capacidades de una shell mediante variables. Con una variable de shell, puede almacenar datos a que hacer referencia una y otra vez mientras ejecuta diferentes comandos en una shell dada. Por ejemplo, puede definir una variable de shell almacenando el nombre de un nombre de archivo complejo. Después, en lugar de escribir el nombre del archivo de nuevo en comandos diferentes, puede hacer referencia a éste con la variable de shell.

Se tiene capacidad para generar variables dentro de una shell, y dichas variables se denominan *variables de shell*. Algunas utilerías, como Mail, tienen sus shell correspondientes con variables de shell propias. También puede crear su propia shell usando lo que se llama *secuencia de comandos de shell*. Tiene una shell de usuario activa en cuanto inicia sesión. A esto suele conocerse como *shell de inicio de sesión*. Variables de parámetros especiales en el nivel sistema se definen en esta shell de inicio de sesión. Las variables de shell también pueden utilizarse para definir entornos shell.

NOTA Las variables shell existen mientras una esté activa (es decir, hasta salir de esa shell). Por ejemplo, al salir de la sesión terminará la shell de inicio de sesión. Cuando vuelva a iniciar sesión, debe definirse de nuevo cualquier variable necesaria en su shell de inicio de sesión.

Definición y evaluación de variables: `=`, `$`, `set`, `unset`

Se define una variable en una shell cuando utiliza por primera vez el nombre de la variable. Este puede ser cualquier conjunto de caracteres alfabéticos, incluidos los caracteres en minúscula.

El nombre puede también incluir un número, pero el número no puede ser el primer carácter del nombre. Es posible que un nombre no tenga otro tipo de carácter, como un signo de admiración, unión o incluso un espacio. Tales símbolos están reservados por la shell para su propio uso. También, un nombre de variable puede incluir más de una palabra. La shell usa espacios en la línea de comandos para distinguir diferentes componentes de un comando como opciones, argumentos y el nombre del comando.

Se asigna un valor a una variable con el operador de asignación (=). Debe escribir el nombre de la variable, operador de asignación y después el valor asignado. No coloque espacios alrededor del operador de asignación. Por ejemplo, el operador de asignación `poeta = Virgilio`, fallará. (La shell C tiene un tipo de operación de asignación diferente.) Puede asignar un conjunto de caracteres a una variable. En el siguiente ejemplo, la variable `poeta` se asigna a la cadena de comandos `Virgilio`:

```
$ poeta=Virgilio
```

Una vez haya asignado valor a la variable, puede emplear el nombre de la misma para referir su valor. En general, utiliza valores de variables como argumentos para un comando. Puede hacer referencia al valor de una variable empleando su nombre precedido por el operador `$`. El signo de pesos es un operador especial que utiliza el nombre de la variable para referirse al valor de la variable, que en efecto revisará. La evaluación recupera un valor de dicha variable, generalmente un conjunto de caracteres. El conjunto de caracteres reemplaza después el nombre en la línea de comandos. Dondequiera que se coloque un signo `$` antes del nombre de variable, éste se reemplaza con el valor de la misma. En el siguiente ejemplo, la variable de shell `poeta` se evalúa y su contenido, `Virgilio`, se usa como argumento para un comando `echo`. El comando `echo` sólo hace un envío o imprime un conjunto de caracteres en la pantalla.

```
$ echo $poeta
Virgilio
```

Debe tener cuidado para distinguir entre la evaluación de una variable y su nombre solo. Si deja fuera el operador `$` antes del nombre de variable, todo lo que tiene es la variable por sí sola. En el siguiente ejemplo, el operador `$` está ausente desde el nombre de variable. En tal caso, el comando `echo` tiene como argumento la palabra “poeta”, y también imprime “poeta”:

```
$ echo poeta
poeta
```

El contenido de una variable suele usarse como argumento de comando. Un argumento de comando común es una ruta de directorio. Es tedioso tener que escribir de nuevo la ruta de un directorio en uso una y otra vez. Si asigna la ruta de directorio a una variable, simplemente puede utilizar la variable evaluada en su lugar. La ruta de directorio asignada a la variable se recupera cuando esta se evalúa con el operador `$`. En el siguiente ejemplo se asigna una ruta de directorio a una variable y después se utiliza la variable evaluada en un comando de copiado. La evaluación de `ldir` (que es `$ldir`) da como resultado la ruta `/home/chris/cartas`. El comando copiar evalúa a `cp micarta /home/chris/cartas`.

```
$ ldir=/home/chris/cartas
$ cp micarta $ldir
```

Obtendrá una lista de todas las variables con el comando `set`. Si decide que no quiere cierta variable, puede quitarla con el comando `unset`. El comando `unset` elimina la definición de una variable.

Valores de variable: cadenas

Los valores asignados a las variables pueden incluir cualquier conjunto de caracteres. Estos pueden ser una cadena de caracteres escritos explícitamente o el resultado obtenido de ejecutar un comando

Linux. En casi todos los casos, debe citar los valores utilizando comillas sencillas, comillas, diagonales invertidas o comillas invertidas. Las comillas sencillas, comillas y diagonales invertidas permiten hacer referencia a cadenas de comandos de diferentes formas. Las comillas invertidas tienen la función especial de ejecutar un comando de Linux y utilizar sus resultados como argumentos en la línea de comandos.

Conversión de cadenas en citas: comillas, comillas sencillas y diagonales invertidas

Los valores de variable pueden crearse con cualquier tipo de caracteres. Sin embargo, los problemas ocurren cuando quiere incluir caracteres que la shell también maneja como operadores. Su shell tiene cierto tipo de meta caracteres empleados para evaluar la línea de comandos. Se utiliza un espacio para analizar argumentos en la línea de comandos. Asterisco, signos de interrogación y corchetes son meta caracteres utilizados para generar listas de nombres de archivos. El punto representa el directorio actual/activo. El signo de pesos, \$, se utiliza para evaluar variables, y los caracteres mayor a (>) y menor que (<), son operadores de redirección. El signo ampersand (&, unión, manejado como “y” en español) se emplea para ejecutar comandos en segundo plano y la barra (|) canaliza la salida. Si quiere usar cualquiera de estos caracteres como parte del valor de una variable, primero necesita llamarlos. Al citar un meta carácter en una línea de comandos, hace del meta carácter sólo otro carácter. No lo evalúa la shell.

Puede utilizar comillas dobles, comillas sencillas y diagonales invertidas para convertir en cita dichos meta caracteres. Las comillas sencillas y dobles facilitan convertir en cita varios meta caracteres a la vez. Cualquier meta carácter con comillas dobles o sencillas se convierte en cita. Una diagonal invertida vuelve cita el carácter que le sigue.

Si quiere asignar más de una palabra a una variable, necesita convertir en cita los espacios separando las palabras. Puede hacerlo encerrando todas las palabras en comillas. Considere esto como crear una cadena de comandos de caracteres para ser asignados a la variable. Por supuesto, cualquier otro meta carácter encerrado entre comillas también se convierte en cita.

De los ejemplos siguientes, en el primero las comillas encierran palabras separadas por espacios. Debido a que los espacios se encierran en comillas, se tratan como caracteres, no como delimitadores usados para analizar argumentos de línea de comandos. En el segundo ejemplo, las comillas también encierran un punto, tratándolo como carácter. En el tercer ejemplo, un asterisco también se encierra en comillas. El asterisco se considera otro carácter en la cadena de comandos y no se evalúa.

```
$ noticia="La reunión será mañana"
$ echo $noticia
La reunión será mañana

$ mensaje="El proyecto va a tiempo."
$ echo $mensaje
El proyecto va a tiempo.

$ noticia="Puede obtener una lista de archivos con ls *.c"
$ echo $noticia
Puede obtener una lista de archivos con ls *.c
```

Sin embargo, las comillas no convierten en cita el signo de pesos, el operador evaluando las variables. Todavía se evaluará un operador \$ tras un nombre de variable encerrado en comillas, reemplazando el nombre de la variable con su valor. El valor de la variable después se volverá parte de la cadena de comandos, no el nombre de la variable. Puede haber ocasiones cuando quiera una variable entre comillas sea evaluada. En el siguiente ejemplo, las comillas se utilizan para que el nombre del ganador se incluya en la noticia.

```
$ ganador=daniel
$ noticia="La persona que ganó es $ganador"
$ echo $noticia
La persona que ganó es daniel
```

Por otra parte, en ocasiones querrá se evalúe una variable entre comillas. En ese caso debe usar comillas sencillas. Estas suprimen cualquier evaluación de variable y tratan al signo de moneda como otro carácter. En el siguiente ejemplo, las comillas sencillas evitan la evaluación de la variable ganador.

```
$ ganador=daniel
$ resultado='El nombre está en la variable $ganador'
$ echo $resultado
El nombre está en la variable $ganador
```

Si, en este caso, se utilizaran comillas doble en vez de sencillas, se daría una evaluación involuntaria de la variable. En el siguiente ejemplo, los caracteres "\$ganador" se interpretan como evaluación de variable.

```
$ ganador=daniel
$ resultado="El nombre está en la variable $ganador"
$ echo $resultado
El nombre está en la variable daniel
```

Siempre tiene la opción de convertir en cita cualquier meta carácter, incluido el operador \$, colocando una diagonal invertida antes. El uso de una diagonal invertida es para convertir en cita las teclas ENTER (nuevas líneas). La diagonal invertida es útil cuando quiere evaluar ambas variables en una cadena de comandos e incluir el carácter \$. En el siguiente ejemplo, la diagonal invertida se coloca antes de \$ para tratarlo como carácter de signo de pesos: \\$. Al mismo tiempo, se evalúa la variable \$ganador porque las comillas dobles usadas no convierten en cita el operador \$.

```
$ ganador=daniel
$ resultado="$ganador ganó \$1,000.00"
$ echo $resultado
daniel ganó $1,000.00
```

Citación de comandos: comillas sencillas

Sin embargo, hay ocasiones en que quizás quiera usar comillas sencillas para encerrar un comando de Linux. Las comillas sencillas permiten asignar el comando escrito a una variable. Si lo hace, puede utilizar el nombre de variable como otro nombre para el comando de Linux. Al insertar el nombre de variable antecedido con un operador \$ en la línea de comandos, se ejecutará el comando. En el siguiente ejemplo, una variable de shell se asigna a caracteres creando un comando de Linux para mostrar una lista de archivos, `ls -F`. Observe las comillas sencillas alrededor del comando. Cuando la variable de shell se evalúa en la línea de comandos, el comando de Linux contenido se convertirá en un argumento de línea de comandos y se ejecutará por la shell.

```
$ lsf='ls -F'
$ $lsf
misdatos /informes /cartas
$
```

En realidad está creando otro nombre para el comando, como alias.

Valores de comandos de Linux: comillas invertidas

A pesar de que puede crear valores de variables escribiendo caracteres o cadenas de caracteres, también puede obtener valores desde otros comandos Linux. Para asignar el resultado de los comandos de Linux a una variable, primero necesita ejecutar el comando. Si coloca un comando de Linux entre comillas invertidas en la línea de comandos, este comando se ejecutará primero y su resultado se convertirá en argumento en la línea de comandos. En el caso de asignaciones, el resultado de un comando puede asignarse a una variable colocando el comando entre comillas invertidas para ejecutarlo primero. Puede pensar en las comillas invertidas como una expresión que consiste de un comando para ejecución, cuyo resultado después se asigna a una variable. Los caracteres componiendo el comando por sí solo no se asignan. En el siguiente ejemplo, el comando `ls *.c` se ejecuta y su resultado se asigna a la variable `listc`. `ls *.c`, que genera una lista de todos los archivos con extensión `.c`. Esta lista de archivos después se asigna a la variable `listc`.

```
$ listc=`ls *.c`
$ echo $listc
principal.c prog.c lib.c
```

Necesita tener en mente la diferencia entre comillas sencillas e invertidas. Las comillas sencillas tratan a un comando de Linux como un conjunto de caracteres. Las comillas invertidas fuerzan la ejecución de un comando de Linux. Tal vez en ocasiones ingrese por accidente comillas sencillas cuando en realidad quiere usar comillas invertidas. De los siguientes ejemplos, en el primero la asignación para la variable `lssc` tiene comillas sencillas, no invertidas, colocadas alrededor del comando `ls *.c`. En este caso, `ls *.c` se coloca alrededor del comando `ls *.c`, forzando la evaluación del comando. Se genera una lista de nombres de archivos terminando en `.c` y se asignan al valor de `lssc`.

```
$ lssc='ls *.c'
$ echo $lssc
ls *.c

$ lssc=`ls *.c`
$ echo $lssc
principal.c prog.c
```

Secuencia de comandos de shell: comandos definidos por el usuario

Puede colocar comandos en un archivo y después hacer que la shell los lea y ejecute. En este sentido, el archivo funciona como un programa de shell, ejecutando los comandos igual que si fueran instrucciones de un programa. A un archivo conteniendo comandos de shell se le denomina *secuencia de comandos de shell*.

Puede insertar comandos de shell en un archivo de secuencia de comandos, al utilizar un editor de texto estándar, por ejemplo el editor Vi. El comando `sh` ó `.`, usado con el nombre de archivo de la secuencia de comandos, leerá el archivo de la secuencia y ejecutará los comandos. En el siguiente ejemplo, el archivo de texto `lsc` contiene un comando `ls` que sólo despliega archivos con la extensión `.c`:

```
lsc
ls *.c
```

Aquí se muestra una ejecución para la secuencia de comandos **lsc**:

```
$ sh lsc
principal.c calc.c
$ . lsc
principal.c calc.c
```

Ejecución de las secuencias de comandos

Puede prescindir de los comandos **sh** y **.** configurando los permisos ejecutables de un archivo de secuencia de comandos. Cuando el archivo se crea con su editor de texto, se dan permisos de sólo lectura y escritura. El comando **chmod** con la opción **+x** le dará al archivo de secuencia de comandos permisos de ejecución. Una vez que sea ejecutable, al ingresar el nombre del archivo de la secuencia de comandos en el indicador de comandos de la shell y oprimir ENTER, se ejecutará el archivo de secuencia de comandos y los comandos de shell contenidos. En realidad, el nombre del archivo de la secuencia de comandos se convierte en un comando nuevo de shell. De esta forma, puede usar las secuencias de comandos shell para designar y crear sus propios comandos de Linux. Sólo es necesario configurar el permiso una vez. En el siguiente ejemplo, está activado el permiso de ejecución del archivo **lsc** para el dueño. Después, la secuencia de comandos de shell de **lsc** se ejecuta directamente como cualquier otro comando de Linux.

```
$ chmod u+x lsc
$ lsc
principal.c calc.c
```

Tal vez deba especificar la secuencia de comandos en uso en su directorio de trabajo actual. Se hace esto poniendo un prefijo al nombre de la secuencia de comandos con una combinación de punto y diagonal, **./**, como en **./lsc**. Este punto es un carácter especial representando el nombre de su directorio de trabajo actual. La diagonal es un separador de ruta del directorio. En el siguiente ejemplo se muestra como ejecutar una secuencia de comandos **lsc**:

```
$ ./lsc
principal.c calc.c
```

Argumentos de secuencia de comandos

Al igual que cualquier comando de Linux puede tomar argumentos, también es posible en la secuencia de comandos de shell. Se hace referencia a los argumentos en la línea de comandos de manera secuencial, iniciando con 1. Se hace referencia a un argumento usando el operador **\$** y el número de su posición. Se hace referencia al primer argumento con **\$1**, al segundo con **\$2**, etc. En el siguiente ejemplo, la secuencia de comandos **lsetxt** imprime archivos con una extensión especificada. El primer argumento es la extensión. La secuencia de comandos se ejecuta entonces con el argumento **c** (por supuesto, debió establecer el permiso de ejecución).

```
lsetxt
ls *. $1
```

Aquí se muestra una ejecución de la secuencia de comandos **lsetxt** con un argumento:

```
$ lsetxt c
principal.c calc.c
```

En el siguiente ejemplo, los comandos para imprimir un archivo con números de líneas se han colocado en un archivo ejecutable denominado **lpnum**, que toma un nombre de archivo como

argumento. El comando **cat** con la opción **-n** saca primero el contenido del archivo con los números de línea. Después, esta salida es canalizada hacia el comando **lpr**, que lo imprime. El comando para imprimir los números de línea se ejecuta en segundo plano

```
lpnum
cat -n $1 | lpr &
```

Aquí se muestra una ejecución para la secuencia de comandos **lpnum** con un argumento:

```
$ lpnum misdatos
```

Tal vez necesite hacer referencia a más de un argumento simultáneamente. El número de argumentos utilizados puede variar. En **lpnum**, tal vez quiera imprimir tres archivos a la vez y cinco archivos en algún otro momento. El operador **\$** con el asterisco, **\$***, hace referencia a todos los argumentos en la línea de comandos. En el siguiente ejemplo, **lpnum** se vuelve a escribir utilizando **\$***, para que tome un número de argumentos diferente cada vez que lo use.

```
lpnum
cat -n $* | lpr &
```

Aquí se muestra una ejecución de la secuencia de comandos **lpnum** con varios argumentos:

```
$ lpnum misdatos prefacio
```

Conjunto de argumentos TCSH: **argv**

Las shells TCSH y C emplean un conjunto diferente de variables de argumentos para referir los argumentos. Son muy similares a las usadas en el lenguaje de programación C. Cuando se invoca la secuencia de comandos de la shell TCSH, todas las palabras de la línea de comandos se analizan y colocan en elementos de un conjunto denominado **argv**. El conjunto **argv[0]** almacenará el nombre de la secuencia de comandos de shell y, a partir de **argv[1]**, cada elemento almacenará un argumento insertado en la línea de comandos. En el caso de las secuencias de comandos de shell, **argv[0]** siempre contendrá el número de la secuencia de comandos de shell. Al igual que cualquier elemento del conjunto, puede acceder al contenido de un elemento de conjunto de argumentos colocando un operador **\$** antes. Por ejemplo, **\$argv[1]** accede al contenido del primer elemento en el conjunto **argv**, el primer argumento. En la secuencia de comandos **saludoarg**, un saludo se pasa al primer argumento en la línea de comandos. Accederá a este primer argumento con **\$argv[1]**.

```
saludoarg
#
echo "El saludo que ingresó fue: $argv[1]"
```

A continuación se muestra la ejecución de la secuencia de comandos **saludoarg**:

```
% saludoarg Hola
El saludo que ingresó fue: Hola
```

Cada palabra se analiza en la línea de comandos, a menos que se encuentre entre comillas. En el siguiente ejemplo, la secuencia de comandos **saludoarg** se invoca con una cadena de caracteres sin comillas y después con una que sí las incluye. Observe que la cadena de caracteres citada, "Hola, cómo estás", fue tratada como argumento.

```
% saludoarg Hola, cómo estás
El saludo que ingresó fue: Hola
% saludoarg "Hola, como estas"
El saludo que ingresó fue: Hola, cómo estás
```


Si más de un argumento es insertado, se puede hacer referencia a cada argumento con un elemento correspondiente en el conjunto **argv**. En el siguiente ejemplo, la secuencia de comandos **misargs** imprime cuatro argumentos. Se insertan cuatro argumentos en la línea de comandos.

```
misargs
#
echo "El primer argumento es: $argv[1]"
echo "El segundo argumento es: $argv[2]"
echo "El tercer argumento es: $argv[3]"
echo "El cuarto argumento es: $argv[4]"
```

Aquí se muestra la ejecución de la secuencia de comandos **misargs**:

```
% misargs Hola Saludos hey "Cómo estás"
El primer argumento es: Hola
El segundo argumento es: Saludos
El tercer argumento es: hey
El cuarto argumento es: Cómo estás
```

Variables de entorno y subshells: export y setenv

Cuando inicia sesión en su cuenta, el sistema Linux genera su shell de usuario. Dentro de esta shell, puede enviar comandos y declarar variables. También crear y ejecutar secuencias de comandos de shell. Sin embargo, cuando ejecuta una secuencia de comandos de shell, el sistema genera una subshell. Entonces tendrá dos shells: en la que inició sesión y la generada con la secuencia de comandos. Dentro de la secuencia de comandos de shell puede ejecutar otra, que después tendrá su propia shell. Cuando una secuencia de comandos ha terminado la ejecución, su shell termina y vuelve a entrar a la shell desde la que se ejecutó. En este sentido, puede tener muchas shells, cada una anidada dentro de otra.

Las variables que definidas en una shell son locales a ésta. Si define una variable en una secuencia de comandos de shell, entonces, al ejecutar la secuencia de comandos, la variable se define con esa secuencia de comandos de shell y es local a ella. Ninguna otra shell puede hacer referencia a ésta. En ese sentido, la variable se esconde en su shell.

Para ilustrar esta situación con mayor claridad, en el siguiente ejemplo se emplearán dos secuencias de comandos; a una de ellas se le llama desde el interior de la otra. Cuando se ejecuta la primera secuencia de comandos, genera su propia shell. En el interior de esta shell, se ejecuta otra secuencia de comandos que, a su vez, genera su propia shell. En el siguiente ejemplo, el usuario ejecuta primero la secuencia de comandos **dispprimera**, que despliega el primer nombre. Cuando se ejecuta la secuencia de comandos **dispprimera**, genera su propia shell y después, dentro de esa shell, define la variable **nombre**. Después de desplegar el contenido de **nombre**, la secuencia de comandos ejecuta otra secuencia de comandos: **dispúltima**. Durante la ejecución, genera su propia shell. Define la variable **apellido** dentro de su shell y después despliega el contenido de **apellido**. Entonces trata de referirse a **nombre** y desplegar su contenido. No puede hacerlo porque **nombre** es local a la shell **dispprimera** y no puede hacerse referencia afuera de ella. Un mensaje de error se despliega indicando que, para la shell **dispúltima**, **nombre** es una variable no definida.

```
dispprimera
nombre="Charles"

echo "El nombre es $nombre"

dispúltima
```

```
dispúltima
apellido="Dickens"

echo "El apellido es $apellido"
echo "$nombre $apellido"
```

Aquí se muestra la ejecución de la secuencia de comandos **dispprimera**:

```
$ dispprimera
El nombre es Charles
El apellido es Dickens
Dickens
sh: nombre: not found
$
```

```
dispfile
miarchivo="Lista"

echo "Desplegando $miarchivo"
pr -t -n $miarchivo

imprimirarchivo
imprimirarchivo

miarchivo="Lista"

echo "Imprimiendo $miarchivo"
lp $miarchivo &
```

Aquí se muestra la ejecución de la secuencia de comandos **disparchivo**:

```
$ disparchivo
Desplegando Lista
1 pantalla
2 modem
3 papel
Imprimiendo Lista
$
```

Si quiere que el mismo valor de una variable se utilice en una shell y una subshell de secuencia de comandos, simplemente puede definir la variable dos veces, una vez en cada secuencia de comandos, y asignarle el mismo valor. En el ejemplo anterior, existe una variable **miarchivo** definida en **disparchivo** e **imprimirarchivo**. El usuario ejecuta la secuencia de comandos **b**, para desplegar primero el archivo de lista con números de línea. Cuando la secuencia de comandos ejecuta **disparchivo**, genera su propia shell y después, dentro de esa shell, define la variable **miarchivo**. Después de desplegar el contenido del archivo, la secuencia de comandos luego ejecuta otra secuencia de comandos, **imprimirarchivo**. Al ejecutarse, genera su propia shell. Define su propia variable **miarchivo** en su shell y después envía un archivo a la impresora.

¿Qué pasa si quiere definir una variable en una shell y tiene su valor referido en cualquier subshell? ¿Por ejemplo, qué pasa si quiere definir la variable **miarchivo** en la secuencia de comandos **disparchivo** y que se haga referencia a su valor, **Lista**, desde de la secuencia de comandos **imprimirarchivo**, en lugar de definir explícitamente otra variable en ésta última? Ya que

las variables son locales a la shell que definen, no hay forma de que haga esto con variables ordinarias. Sin embargo, existe un tipo de variable denominada *variable de entorno* que permite se haga referencia a su valor por cualquier subshell. Las variables de entorno constituyen un entorno para shell y cualquier subshell generada, sin importar que tan anidada esté.

Puede definir las variables de entorno en tres principales tipos de shell: Bourne, Korn y C. Sin embargo, la estrategia utilizada para implementar variables de entorno en las shells Bourne y Korn es muy diferente a C. En Bourne y Korn, las variables de entorno se exportan. Es decir, se hace una copia de una variable de entorno en cada subshell. En cierto sentido, si la variable **miarchivo** se exporta, una copia se define automáticamente en cada subshell. En la shell C, por otra parte, una variable de entorno se define sólo una vez y puede hacer referencia directamente a cualquier subshell.

Variables de entorno de shell

En las shell Bourne, BASH y Korn, puede considerarse que una variable de entorno es una variable regular con capacidades agregadas. Para hacer una variable de entorno, se aplica el comando **export** a una variable ya definida. El comando **export** instruye al sistema para definir una copia de esa variable en cada nueva shell generada. Cada nueva shell tendrá su propia copia de la variable de entorno. A este proceso se denomina *exportación de variables*.

En el siguiente ejemplo, la variable **miarchivo** se define en la secuencia de comandos **disparchivo**. Después se convierte en una variable de entorno usando el comando **export**. Por tanto, la variable **miarchivo** se exportará a cualquier subshell, como la generada cuando se ejecuta **imprimirarchivo**.

```
disparchivo
miarchivo="Lista"
export miarchivo

echo "Desplegando $miarchivo"
pr -t -n $miarchivo

imprimirarchivo

imprimirarchivo
echo "Imprimiendo $miarchivo"
lp $miarchivo &
```

Aquí se muestra la ejecución de la secuencia de comandos **disparchivo**:

```
$ disparchivo
Desplegando Lista
1 pantalla
2 módem
3 papel
Imprimiendo Lista
$
```

Cuando se ejecute **imprimirarchivo**, se le asignará su propia copia de **miarchivo** y puede hacer referencia a esa copia en su propia shell. No necesita definir explícitamente otra variable **miarchivo** en **imprimirarchivo**.

Es un error pensar en variables de entorno exportadas como variables globales. Una nueva shell nunca puede hacer referencia a una variable fuera de sí misma. En cambio, se genera una copia de la variable con su valor para la nueva shell. Puede considerar que las variables exportadas envían sus valores a una shell, no a sí mismas. Para quienes están familiarizados con estructuras de programación, las variables exportadas pueden considerarse una forma de llamada por valor.

Variables de entorno de las shells TCSH y C

En las shell TCSH y C, una variable de entorno se define usando un comando de definición separada, **setenv**. En ese sentido, una variable de entorno es realmente un tipo de variable muy diferente al de una variable local regular. Una variable de entorno de la shell C opera de manera más parecida a una variable global. Cualquier subshell puede referirse a ella. Esto difiere de Bourne, BASH y Korn, en las que sólo se pasa una copia de la variable de entorno, para usarse como subshell.

Para definir una variable de entorno primero inserte el comando **setenv**, seguido por el nombre de la variable y después el valor. No hay operador de asignación. En el siguiente ejemplo, se define la variable de entorno **miarchivo** y se asigna el valor **Lista**.

```
% setenv miarchivo lista

disparchivo
setenv miarchivo "Lista"

echo "Desplegando $miarchivo"
cat -n $miarchivo

imprimirarchivo

imprimirarchivo
echo "Imprimiendo $miarchivo"
lpr $miarchivo &
```

Aquí se ejecuta la secuencia de comandos **disparchivo**:

```
$ disparchivo
Desplegando List
1 pantalla
2 módem
3 papel
Imprimiendo List
$
```

En el ejemplo previo, la variable **miarchivo** se define como variable de entorno en la secuencia de comandos **disparchivo**. Observe que se usa el comando **setenv** en vez de **set**. Ahora puede hacerse referencia a la variable **miarchivo** en cualquier subshell, como la generada cuando se ejecuta **imprimirarchivo**.

Al ejecutar **imprimirarchivo**, podrá acceder directamente a la variable **miarchivo** definida en la shell de la secuencia de comandos **disparchivo**.

Estructuras de control

Puede controlar la ejecución de los comandos de Linux en la secuencia de comandos de shell con estructuras de control. Las estructuras de control permiten repetir comandos y seleccionar ciertos comandos en lugar de otros. Una estructura de control consta de dos componentes principales: una prueba y los comandos. Si la prueba tiene éxito, entonces se ejecutan los comandos. De esta forma, puede usar estructuras de control para tomar decisiones sobre la ejecución de los comandos.

Existen dos tipos diferentes de estructuras de control: *bucles* y *condiciones*. Un bucle repite los comandos, si una condición ejecuta un comando mientras ciertas condiciones se cumplan. La shell BASH tiene tres estructuras de control de bucle: **while**, **for** y **for-in**. Existen dos estructuras de condición: **if** y **case**. Las estructuras de control tienen como prueba la ejecución de un comando de Linux. Todos los comandos de Linux regresan un estado de salida después de terminada la ejecución. Si un comando tiene éxito, su estado de finalización será 0. Si el comando falla por cualquier razón, su estado de finalización será un valor positivo que refiere el tipo de falla ocurrida. Las estructuras de control revisan si el estado de finalización de un comando Linux es 0 u otro valor. En el caso de las estructuras **if** y **while**, si el estado de finalización es un valor 0, entonces el comando tuvo éxito y la estructura continúa.

Operaciones de prueba

Con el comando **test**, puede comparar enteros y cadenas de comandos, incluso realizar operaciones lógicas. El comando consta de la palabra clave **test**, seguida por valores que se comparan, separados por una opción especificando el tipo de comparación que se hace. La opción puede considerarse como el operador, pero se escribe, como otras opciones, con un signo de menos y códigos de letras. Por ejemplo, **-eq** es representa la opción comparación de igualdad. Sin embargo, existen dos operaciones de cadena de comandos que realmente usan un operador en vez de una opción. Cuando compara la igualdad de dos cadenas de comandos, se usa el signo igual (=). En el caso de la desigualdad, se maneja el comando **!=**. En la Tabla 4-1 se presenta una lista de opciones y operadores utilizados comúnmente por **test**. Aquí se muestra la sintaxis del comando **test**:

```
test valor -option valor
test cadena = cadena
```

En el siguiente ejemplo, el usuario compara dos valores enteros para ver si son iguales. En este caso, necesita usar la opción igualdad, **-eq**. El estado de finalización del comando **test** se examina para encontrar el resultado de la operación de prueba. La variable especial de shell **\$?** almacena el estado de finalización de los comandos de Linux de ejecución más reciente.

```
$ núm=5
$ test $núm -eq 10
$ echo $?
1
```

En vez de usar la palabra clave **test** para el comando **test**, puede recurrir a las llaves. El comando **test \$saludo = "hola"** puede escribirse

```
$ [ $saludo = "hola" ]
```

De manera similar, el comando de prueba **test \$num -eq 10** puede escribirse

```
$ [ $núm -eq 10 ]
```

Comparaciones de enteros	Función
-gt	Mayor que
-lt	Menor que
-ge	Mayor que o igual a
-le	Menor que o igual a
-eq	Igual a
-ne	No igual
Comparaciones de cadenas de comandos	
-z	Pruebas para cadenas de comandos vacías
=	Pruebas para cadenas de comandos iguales
!=	Pruebas para cadenas de comandos desiguales
Operadores lógicos	
-a	Y lógico
-o	O lógico
!	NO lógico
Pruebas de archivo	
-f	El archivo existe y es un archivo regular
-s	El archivo no está vacío
-r	El archivo puede leerse
-w	El archivo puede ser modificado y se puede escribir en él
-x	El archivo es ejecutable
-d	El nombre de archivo es un nombre de directorio

TABLA 4-1 Operadores de prueba de la shell BASH

Las propias llaves deben rodearse por un espacio en blanco: un espacio, TAB o ENTER. Sin el espacio, son inválidas.

Estructuras de control condicionales

La shell BASH tiene un conjunto de estructuras de control condicionales que permiten seleccionar cuáles comandos de Linux ejecutar. Muchas de éstas similares a las estructuras de control condicionales encontradas en lenguajes de programación, pero con varias diferencias. La condición **if** prueba el éxito de un comando de Linux, no una expresión. Además, el final de un comando **if-then** debe indicarse con la palabra clave **fi**, y el final de un comando **case** debe indicarse con la palabra clave **esac**. Las estructuras de control condicionales se muestran en la lista de la tabla 4-2.

La estructura **if** coloca una condición en los comandos. La condición es el estado de finalización de un comando específico de Linux. Si un comando es exitoso, al regresar un estado de finalización de 0, entonces se ejecutan los comandos dentro de la estructura **if**. Si el estado de finalización es otro, distinto de 0, entonces el comando ha fallado y no se ejecutarán los comandos en la estructura

Estructuras de control de condición: if, else, elif, case	Función
if comando then comando fi	if ejecuta una acción si su comando de prueba es cierto.
if comando then comando else comando fi	if-else ejecuta una acción si el estado de finalización de su comando de prueba es cierto; si es falso, entonces se ejecuta la acción else .
if comando then comando elif comando then comando else comando fi	elif permite anidar estructuras if , facilitando la selección entre varias opciones; en la primera estructura cierta if , sus comandos se ejecutan y el control deja la estructura entera elif .
case cadena in patrón) comando ; ; esac	case relaciona el valor de la cadena de comandos con cualquiera de los varios patrones; si su patrón se relaciona, sus comandos asociados se ejecutan.
comando && comando	Una condición Y lógica regresa un valor cierto 0, si uno o de los otros comandos regresa un valor 0 cierto; si ambos comandos regresan un valor distinto de cero, entonces la condición Y es falsa y también regresa un valor distinto de cero.
comando comando	Una condición O lógica regresa un valor cierto 0 si uno o de los otros comandos regresa(n) un valor 0 cierto; si ambos comandos regresan un valor distinto de cero, entonces la condición O es falsa y también regresa un valor distinto de cero.
! comando	La condición lógica NO invierte el valor devuelto por el comando.
Estructuras de control de bucle: while, until, for, for-in, select	
while comando do comando done	while ejecuta una acción siempre y cuando el comando de prueba sea cierto.
until comando do comando done	until ejecuta una acción siempre y cuando el comando de prueba sea falso.

TABLA 4-2 Estructuras de control de la shell BASH (continúa)

Estructuras de control de bucle: while, until, for, for-in, select	
for <i>variable in valores de lista</i> do <i>comando</i> done	for-in está diseñado para usarse con listas de valores; a la variable operando se le asignan valores de la lista consecutivamente.
for <i>variable</i> do <i>comando</i> done	for está diseñado para argumentos de secuencia de comandos referidos; a la variable operando se asigna cada valor de argumento consecutivamente.
select <i>cadena in lista de elementos</i> do <i>comando</i> done	select crea un menú basado en elementos en la lista de elementos; después ejecuta el comando; este suele ser case .

TABLA 4-2 Estructuras de control de la shell BASH (*continuación*)

if. El comando **if** comienza con la palabra clave **if**, seguida por un comando de Linux cuya condición de finalización será evaluada. La palabra clave **fi** termina el comando. La secuencia de comandos **elsels**, en el siguiente ejemplo, ejecuta el comando **ls** para mostrar una lista de archivos con dos opciones diferentes posibles, ya sea por tamaño o con toda la información del archivo. Si el usuario inserta una **s**

```
elsels
echo Enter s to list file by sizes
echo      otherwise all file information is listed.
echo -n "Please enter option: "
read choice
if [ "$choice" = s ]
    then
        ls -s
    else
        ls -l
fi
echo Good-bye
```

Una ejecución del programa sigue:

```
$ elsels
Enter s to list file by sizes,
otherwise all file information is listed.
Please enter option: s
total 2
    1 lunes      2 hoy
$
```


Estructuras de control bucle

El bucle **while** repite comandos. Un bucle **while** comienza con la palabra clave **while** seguida por un comando Linux. La palabra clave **do** sigue en la línea que continúa. El final del bucle se especifica con la palabra **done**. El comando de Linux utilizado en las estructuras **while**, tiende a ser un comando de prueba indicado entre corchetes.

La estructura **for-in** está diseñada para referir una lista de valores de manera secuencial. Toma dos operandos: una variable y una lista de valores. Los valores en la lista se asignan uno por uno a la variable en la estructura **for-in**. Como el comando **while**, la estructura **for-in** es un bucle. Cada vez que pasa el bucle, el siguiente valor en la lista se asigna a la variable. Cuando se llegue al final de la lista, el bucle se detiene. Como el bucle **while**, el cuerpo de un bucle **for-in** comienza con la palabra clave **do** y termina con la palabra clave **done**. La secuencia de comandos **cbackup** hace una copia de seguridad de cada archivo y lo coloca en un directorio denominado **csorigen**. Observe el uso del carácter especial ***** para generar una lista de todos los nombres de archivo con extensión **.c**.

```
cbackup
for archivocs in *.c
do
    cp $archivocs csorigen/$archivocs
    echo $archivocs
done
```

He aquí una ejecución del programa:

```
$ cbackup
io.c
lib.c
principal.c
$
```

Si no especifica una lista de valor a una estructura **for**, tomará como lista de valores los argumentos de la línea de comandos. Los argumentos especificados en la línea de comandos cuando se invoca el archivo de shell, se convierten en una lista de valores a los que hace referencia el comando **for**. La variable usada por el comando **for** automáticamente va tomando cada valor de argumento en secuencia. La primera vez que pasa el bucle, la variable toma el valor del primer argumento. La segunda vez, toma el valor del segundo argumento.

Estructuras de control de las shell TCSH y C

Como en otras shell, TCSH tiene un conjunto de estructuras de control que le permiten manipular la ejecución en una secuencia de comandos. Existen bucles y estructuras de control condicionales con las que puede repetir comandos de Linux o tomar decisiones acerca de cuáles comandos quiere ejecutar. Las estructuras de control **while** e **if** tienen un propósito más general, realizan iteraciones y toman decisiones recurriendo a varias pruebas diferentes. Las estructuras de control **switch** y **foreach** son operaciones más especializadas. La estructura **switch** es una forma restringida de condición **if** que revisa si un valor es igual a otro, entre un conjunto de valores posibles. La estructura **foreach** es un tipo limitado de bucle ejecutado a través de una lista de valores, asignando un valor nuevo a una variable con cada repetición.

La shell TCSH difiere de otras en que sus estructuras de control se parecen más a un formato de lenguaje de programación. La condición de prueba para la estructura de control de TCSH es una expresión asignando un valor cierto o falso, no un comando de Linux. Una diferencia clave entre las estructuras de control de las shell BASH y TCSH es que las de TCSH no pueden redirigir ni canalizar su salida. De manera estricta, son estructuras de control, controlando la ejecución de comandos.

Expresiones de prueba

Las estructuras de control **while** e **if** usan una expresión como prueba. Una prueba cierta es una expresión que da como resultado un valor distinto de cero. Una prueba falsa es cualquier expresión que da como resultado un valor de 0. En la shell TCSH, las expresiones relacionales e iguales pueden usarse de manera sencilla como expresiones de prueba, porque dan como resultado 1 cuando son ciertas y 0 si son falsas. Existen muchos operadores posibles para usarse en una expresión. Puede utilizar un número de operadores en una expresión, como se muestra en la tabla 4-3. La expresión de prueba también puede ser aritmética o una cadena de comandos de comparación, pero las cadenas de comandos sólo pueden ser comparadas para igualdad o desigualdad.

A diferencia de la shell BASH, debe encerrar las expresiones de prueba **if** y **while** de la shell TCSH entre paréntesis. En el siguiente ejemplo se muestra una expresión de prueba simple, para revisar si dos cadenas de comando son iguales.

```
if ( $saludo == "hola" ) then
    echo Saludo informal
endif
```

La shell TCSH tiene un conjunto de operadores separados para probar las cadenas de comandos contra otras cadenas de comandos o expresiones regulares. Los operadores **==** y **!=** prueban igualdad o desigualdad de las cadenas de comandos. Los operadores **=~** y **!~** prueban una cadena de comandos contra una expresión regular y si hay coincidencias con un patrón. La expresión regular puede contener cualquiera de los caracteres especiales de shell. En el siguiente ejemplo, cualquier valor de **saludo** que empezando con una **h** mayúscula o minúscula relacionará la expresión regular **[Hh]***.

```
if ( $saludo =~ [Hh]* ) then
    echo Saludo informal
endif
```

Cómo en la shell BASH, TCSH tiene varios operadores especiales para probar el resultado de los archivos. Muchos de estos operadores son iguales. En el siguiente ejemplo, el comando **if** prueba si puede leerse el archivo **misdatos**.

```
if ( -r misdatos ) then
    echo Saludo informar
endif
```

Condiciones de la shell TCSH: if-then, if-then-else, switch

La shell TCSH tiene un conjunto de estructuras de control condicionales con las que puede tomar una decisión acerca de cuáles comandos de Linux se ejecutarán. Muchas de estas estructuras de control condicional son similares a las encontradas en la shell BASH. Sin embargo, existen diferencias clave. La estructura **if** de la shell TCSH termina con la palabra clave **endif**. La estructura **switch** utiliza la palabra **case** de manera diferente. Termina con la palabra clave **endsw**

Comparación de cadenas de comandos	Función y Descripción
==	Prueba la igualdad en cadenas de comandos
!=	Prueba la desigualdad en cadenas de comandos
=~	Compara la cadena con un patrón para probar si es igual; el patrón puede ser cualquier expresión regular
!~	Compara la cadena con un patrón para probar si no es igual; el patrón puede ser cualquier expresión regular
Operadores lógicos	
&&	Y lógico
 	O lógico
!	No lógico
Pruebas de archivo	
-e	El archivo existe
-r	El archivo puede leerse
-w	El archivo puede ser modificado y puede escribirse en él
-x	El archivo es ejecutable
-d	El nombre de archivo es un nombre de directorio
-f	El archivo existe y es un archivo regular
-o	El archivo es propiedad del usuario
-z	El archivo está vacío
Operadores relacionales	
>	Mayor que
<	Menor que
>=	Mayor que o igual a
<=	Menor que o igual a
!=	No es igual
==	Es igual

TABLA 4-3 Operadores de expresión de prueba TCSH

y maneja la palabra clave **breaksw** en vez de dos puntos y coma. Además, existen dos estructuras de control **if**: una versión simple para ejecutar sólo un comando y una versión más compleja ejecutando varios comandos, además de comandos alternos. La versión simple **if** consta de la palabra clave **if** seguida por una prueba y un solo comando de Linux. La versión más compleja termina con la palabra clave **endif**. Las estructuras condicionales de la shell TCSH se muestran en la lista de la tabla 4-4.

Estructuras de control	Descripción
if (<i>expresión</i>) then <i>comandos</i> endif	Si la expresión es cierta, los siguientes comandos se ejecutan. Puede especificar más de un comando de Linux.
if (<i>expresión</i>) then <i>comando</i> else <i>comando</i> endif	Si la expresión es cierta, se ejecuta el comando después de then . Si la expresión es falsa, se ejecuta el comando siguiente de else .
switch (<i>cadena</i>) case <i>patrón</i> : <i>comando</i> breaksw default : <i>comando</i> endsw	Le permite seleccionar entre varios comandos alternos.

TABLA 4-4 Estructuras de control condicional TCSH

La estructura if-then

La estructura **if-then** coloca una condición en varios comandos de Linux. Esa condición es una expresión. Si la presión da como resultado un valor distinto de 0, la expresión es cierta y se ejecutan estos comandos en la estructura **if**. Si la expresión da como resultado un valor de 0, la expresión es falsa y no se ejecutan los comandos de la estructura **if**.

La estructura **if-then** comienza con la palabra clave **if** y es seguida por una expresión encerrada entre paréntesis. La palabra clave **then** sigue a la expresión. Después puede especificar cualquier número de comandos de Linux en las siguientes líneas. La palabra clave **endif** termina el comando **if**. Observe que, en la shell BASH la palabra clave **then** está en una línea por sí sola, en TCSH, **then** está en la misma línea que la expresión de prueba. Aquí se muestra la sintaxis para la estructura **if-then**:

```
if ( Expresión ) then
    Comandos
endif
```

La secuencia de comandos **ifls**, mostrada a continuación permite presentar una lista de archivos por tamaño. Si inserta una **s** en el indicador de comandos, se presenta una lista con todos los archivos del directorio actual, seguida por el número de bloques que utiliza. Si inserta cualquier otra cosa en la petición, la prueba **if** falla y la secuencia de comandos hace nada.

```
ifls
#
echo -n "Por favor inserte una opción: "
set opción = $<

if ($opción == "s") then
    echo Presentando archivos por tamaño
    ls -s
endif
```

Aquí se muestra una ejecución de la secuencia de comandos **ifls**:

```
% ifls
Por favor inserte una opción: s
Presentando archivos por tamaño
total 2
  1 lunes      2 hoy
```

A menudo, necesita seleccionar entre dos opciones, considerando que una expresión es cierta. La palabra clave **else** permite una estructura **if** para seleccionar entre dos comandos opcionales. Si la expresión es cierta, se ejecutan los comandos inmediatamente después de la expresión de prueba. Si la expresión es falsa, los que se ejecutan son los que siguen tras la palabra clave **else**. Aquí se muestra la sintaxis para el comando **if-else**:

```
if ( expresión ) then
    comandos
else
    comandos
endif
```

La secuencia de comandos **elsels**, en el siguiente ejemplo, ejecuta el comando **ls** para mostrar una lista de archivos con dos posibles opciones: por tamaño o con toda la información de los archivos. Si el usuario inserta una **s**, los archivos se muestran en una lista por tamaño; de otra forma, toda la información de los archivos se muestra en la lista.

```
elsels
#
echo Ingrese s para presentar los archivos por tamaño.
echo de otra manera, se presenta toda la información del archivo.
echo -n "Por favor ingrese una opción : "
set opción = $<

if ($opción == "s") then
    ls -s
else
    ls -l
endif
echo Adiós
```

He aquí una ejecución de la secuencia de comandos **elsels**:

```
> elsels
Ingrese s para presentar los archivos por tamaño,
De otra manera, se presenta toda la información del archivo.
Por favor, ingrese una opción: s
total 2
  1 lunes      2 hoy
Adiós
```

La estructura switch

La estructura **switch** selecciona entre varios comandos opcionales posibles. Es similar a la estructura **case** de la shell **BASH**, en que la elección se hace al comparar cadenas de comandos con varios patrones posibles. Cada patrón se asocia con un conjunto de comandos. Si se encuentra una relación, los comandos asociados se realizan.

La estructura **switch** comienza con la palabra clave **switch**, seguida por una cadena de comandos de prueba entre paréntesis. La cadena de comandos a menudo se deriva de la evaluación de una variable. Luego sigue un conjunto de patrones —cada patrón con la palabra clave **case** antes y finalizado con dos puntos. Los comandos asociados con esta elección se muestran en la lista luego de los dos puntos. El comando se termina con la palabra clave **breaksw**. Después de todos los patrones de la lista, la palabra clave **endsw** termina la estructura switch. Aquí se muestra la sintaxis de la estructura switch:

```
switch (cadena de comandos de prueba)
  case patrón:
    comandos
    breaksw
  case patrón:
    comandos
    breaksw
  default:
    comandos
    breaksw
endsw
```

Bucles de la shell TCSH: while, foreach, repeat

La shell TCSH tiene su propio conjunto de estructuras de control de bucle que permiten repetir los comandos de Linux: **while**, **foreach** y **repeat**. Las estructuras de control de bucle de la shell TCSH se muestran en la lista de la tabla 4-5.

La estructura **while** opera de forma muy similar a las estructuras correspondientes encontradas en lenguajes de programación. Al igual que la estructura **if** de la shell TCSH, la estructura **while** verifica el resultado de una expresión. La estructura **foreach** de la shell TCSH, como las estructuras **for** y **for-in** en la shell BASH, no realiza tarea. Sólo recorre una lista de valores, asignando cada valor a cambio a una variable específica. En ese sentido, la estructura **foreach** es muy diferente a estructuras correspondientes encontradas en lenguajes de programación. La estructura **repeat** es una estructura de control simple y limitada. Repite un comando determinado número de veces. No tiene expresión de prueba y tampoco puede repetir más de un comando.

Estructuras de control de bucle	Descripción
while (expresión) comando end	Ejecuta de comandos siempre y cuando la expresión sea cierta.
foreach variable (lista de argumentos) comando end	Repite el bucle para todos los argumentos en la lista. Cada vez que pasa el bucle, la variable se configura para el siguiente argumento en la lista; opera como for-in en la shell BASH.
repeat número de comando	Repite un comando determinado número de veces.
continue	Salta a la siguiente repetición, pasando por alto el recordatorio de los comandos de bucle.
break	Cancela un bucle.

TABLA 4-5 Las estructuras de control de bucle TCSH

La estructura while

El bucle **while** repite comandos. Un bucle **while** comienza con la palabra clave **while** y seguido por una expresión encerrada en paréntesis. El final del bucle se especifica con la palabra clave **end**. Aquí se muestra la sintaxis para el bucle **while**:

```
while ( expresión )
    comandos
end
```

La estructura **while** puede combinarse de manera sencilla con una estructura **switch** para dirigir un menú.

La estructura foreach

La estructura **foreach** está diseñada para hacer referencia, secuencialmente, a una lista de valores. Es muy similar a la estructura **for-in** de la shell BASH. La estructura **foreach** toma dos operandos: una variable y una lista de valores encerrados en paréntesis. Cada valor de la lista está asignado a la variable en la estructura **foreach**. Como la estructura **while**, **foreach** es un bucle. Cada vez que pasa por un bucle, se asigna a la variable el siguiente valor en la lista. Cuando se llega al final de la lista, el bucle se detiene. Como el bucle **while**, el cuerpo del bucle **foreach** termina con la palabra clave **end**. Aquí se muestra la sintaxis para el bucle **foreach**:

```
foreach variable ( lista de valores )
    comandos
end
```

En la secuencia de comandos **milista**, en el siguiente ejemplo, la secuencia de comandos sólo da salida a una lista de cada elemento con la fecha actual. La lista de elementos crea la lista de valores leídos por el bucle **foreach**. Cada elemento se asigna consecutivamente a la variable **grocery**.

```
milista
#
set fechaactual=`date +%D`
foreach abarrotes( leche galletas manzanas queso )
    echo "$abarrotes    $fechaactual"
end
$ milista
leche      12/23/96
galletas   12/23/96
manzanas   12/23/96
queso      12/23/96
$
```

El bucle **foreach** es útil para administrar archivos. En la estructura **foreach**, puede usar caracteres especiales de shell en un patrón para generar una lista de nombres de archivo a ser usados como lista de valores. Esta lista generada de nombres de archivo se convierte después en una lista a la que hace referencia la estructura **foreach**. Un asterisco por sí solo genera una lista de todos los archivos y directorios. ***.c** muestra una lista de archivos con la extensión **.c**. Generalmente son archivos de código fuente C. En el siguiente ejemplo se crea un respaldo de cada archivo y coloca en un directorio de respaldo denominado **csorigen**. El patrón ***.c** genera una lista de nombres de archivo donde puede operar la estructura **foreach**.

```

cbackup
#
foreach archivocs ( *.c )
    cp $archivocs csorigen/$archivocs
    echo $archivocs
end

% cbackup
io.c
lib.c
principal.c

```

La estructura **foreach** sin lista específica de valores toma como su lista de valores los argumentos de la línea de comandos. Los argumentos especificados en la línea de comandos cuando se invocó el archivo de shell, se convirtieron en una lista de valores referidos por la estructura **foreach**. A la variable usada en la estructura **foreach** se asigna automáticamente cada valor de argumento en secuencia. La primera vez que pasa por el bucle, la variable toma el valor del primer argumento. La segunda vez, toma el del segundo argumento, y así sucesivamente.

En la secuencia de comandos **milistarg** del siguiente ejemplo, no hay una lista de valores especificado para el bucle **foreach**. En cambio, el bucle **foreach** lee consecutivamente los valores de los argumentos de la línea de comandos en la variable **comestibles**. Cuando todos los argumentos se han leído, el bucle termina.

```

milistarg
#
set fechaactual=`date '+%D'`
foreach comestibles ( $argv[*] )
    echo "$comestibles      $fechaactual"
end

$ milistarg leche galletas manzanas queso
leche      12/23/96
galletas   12/23/96
manzanas   12/23/96
queso      12/23/96
$

```


Configuración de la shell

Son cuatro las principales shell en sistemas Linux: Bourne Again (BASH), AT&T Korn, TCSH y Z. BASH es la versión avanzada de la shell Bourne, incluyendo muchas características avanzadas desarrolladas por las shell Korn y C. TCSH es una versión mejorada de la shell C, originalmente desarrollada para versiones BSD de Unix. La shell AT&T Unix Korn es de fuente abierta. La shell Z es una versión mejorada de la shell Korn. A pesar de que sus contrapartes de Unix difieren bastante, las shell de Linux mantienen muchas de las mismas características. En Unix, la shell Bourne carece de numerosas capacidades encontradas en otras shell de Unix. Sin embargo, en Linux, la shell BASH incorpora todas las características avanzadas de las shell Korn y C, además de TCSH. Las cuatro shell están disponibles para su uso, aunque la shell BASH es la empleada por omisión.

La shell BASH es la predeterminada para casi todas las distribuciones de Linux. Si está iniciando sesión en una interfaz de línea de comandos, se le colocará en la shell predeterminada automáticamente y dará un indicador de comandos de shell, donde ingresar comandos. El indicador de comandos para la shell BASH es un signo de pesos (\$). En una interfaz GUI, como GNOME o KDE, puede abrir una ventana Terminal que desplegará una interfaz de línea de comandos con un indicador de comandos para la shell predeterminada (BASH). A pesar de que inicie sesión en la shell predeterminada o se despliegue automáticamente en la ventana Terminal, puede cambiar a otra shell introduciendo el nombre. `tcsh` invoca a la shell TCSH, `bash` a la shell BASH, `ksh` a la shell Korn y `zsh` a la shell Z. Puede salir de una shell pulsando CTRL-D o usando el comando `exit`. Sólo necesita un tipo de shell para hacer su trabajo. En la tabla 5-1 se muestran los diferentes comandos que puede utilizar para invocar diferentes shell. Algunas han agregado vínculos que puede usar para invocar la misma shell, como `sh` y `bsh`, que vincula e invoca el comando `bash` para la shell BASH.

En este capítulo se describen las características más comunes de la shell BASH, como alias, al igual que la manera de configurar la shell para sus propias necesidades, mediante variables de shell y archivos de inicialización. Las otras shell comparten muchas características, manejando variables y archivos de inicialización similares.

A pesar de que aquí se muestran características y configuraciones básicas de shell, debe consultar sus respectivos manuales y FAQ en línea para cada shell, a fin de conocer explicaciones y ejemplos más detallados (consulte la tabla 3-1 en el capítulo 3 para conocer los sitios Web de cada shell).

Shells	Descripción
bash	Shell BASH, /bin/bash
bsh	Shell BASH, /bin/bsh (se vincula a /bin/bash)
sh	Shell BASH, /bin/sh (se vincula a /bin/bash)
tcsh	Shell TCSH, /usr/tcsh
csh	Shell TCSH, /bin/csh (se vincula a /bin/tcsh)
ksh	Shell Korn, /bin/ksh (también el vínculo agregado /usr/bin/ksh)
zsh	Shell Z, /bin/zsh

TABLA 5-1 Nombres de comandos para invocar shell

Inicialización de la shell y archivos de configuración

Cada tipo de shell tiene su propio conjunto de archivos de inicialización y configuración. Ya se analizaron los archivos de configuración de la shell BASH. La shell TCSH utiliza archivos **.login**, **.tcshrc** y **.logout** en lugar de **.bash_profile**, **.bashrc** y **.bash_logout**. En vez de **.bash_profile**, algunas distribuciones emplean el nombre **.profile**. La shell Z tiene varios archivos de inicialización: **.zshenv**, **.zlogin**, **.zprofile**, **.zshrc** y **.zlogout**. Consulte la tabla 5-2 para conocer una lista. Revise las páginas Man de cada shell para ver cómo suelen configurarse. Cuando instala una shell, las versiones predeterminadas de esos archivos se colocan automáticamente en los directorios home del usuario. Excepto la shell TCSH, todas las demás utilizan la sintaxis de manera muy parecida para definiciones de variable y asignación de valores (TCSH utiliza una sintaxis diferente, descrita en las páginas Man).

Directorios y archivos de configuración

En general, las aplicaciones instalan archivos de configuración en un directorio home del usuario cuyo contenido es información específica a la configuración, para amoldar la aplicación según las necesidades de cada usuario. Esto puede tomar la forma de un solo archivo de configuración que comienza con un punto o un directorio que contiene varios archivos de configuración. El nombre del directorio también comenzará con un punto. Por ejemplo, Mozilla instala un directorio denominado **.mozilla**, dentro del directorio home del usuario, conteniendo archivos de configuración. Por otra parte, muchas aplicaciones de correo sólo usan un archivo denominado **.mailrc** para almacenar alias, presentando configuraciones elegidas por el usuario, aunque otros como Evolution también tienen uno propio, **.evolution**. La mayoría de archivos de configuración terminan en las letras **rc**. FTP usa un archivo denominado **.netrc**. Casi todos los lectores de noticias manejan un archivo denominado **.newsrsc**. Las entradas en archivos de configuración suelen configurarse por aplicación, aunque generalmente pueden crear entradas directamente al editar el archivo. Las aplicaciones tienen su propio conjunto de variables especiales a las que puede definir y asignar valores. Es posible listar los archivos de configuración en su directorio de inicio con el comando **ls -a**.

Nombre de archivo	Función
Shell BASH	
.bash_profile	Archivo de inicialización de inicio de sesión
.profile	Archivo de inicialización de inicio de sesión (igual que .bash_profile)
.bashrc	Archivo de configuración de la shell BASH
.bash_logout	Nombre de terminación de sesión
.bash_history	Archivo de historial
/etc/profile	Archivo de inicialización de inicio de sesión en el sistema
/etc/bashrc	Archivo de configuración de la shell BASH del sistema
/etc/profile.d	Directorio para archivos de configuración especializados de la shell BASH
Shell TCSH	
.login	Archivo de inicialización de inicio de sesión.
.tcshrc	Archivo de configuración de la shell TCSH
.logout	Archivo de terminación de sesión
Shell Z	
.zshenv	Archivo de inicio de sesión de shell (primera lectura)
.zprofile	Archivo de inicialización de inicio de sesión
.zlogin	Archivo de inicio de sesión de la shell
.zshrc	Archivo de configuración de la Z
.zlogout	Archivo de terminación de sesión
Shell Korn	
.profile	Archivo de inicialización de inicio de sesión
.kshrc	Archivo de configuración de la shell Korn

TABLA 5-2 Archivos de configuración de shell

Alias

Usted utiliza el comando **alias** para crear otro nombre para un comando. El comando **alias** opera como una macro que se expande al comando que representa. El alias no reemplaza literalmente el nombre del comando; sólo da otro nombre a ese comando. Un comando **alias** comienza con la palabra clave **alias** y el nuevo nombre para el comando, seguido por un signo igual y el comando al que hará referencia el alias.

NOTA No puede haber espacios alrededor del signo igual utilizando el comando **alias**.

En el siguiente ejemplo, `list` se convierte en otro nombre para `ls`:

```
$ alias lista=ls
$ ls
misdatos hoy
$ lista
misdatos hoy
$
```

Uso de alias en comandos y opciones

También puede usar un alias para sustituir un comando y su opción, pero necesita encerrar ambos comandos y la opción entre comillas sencillas. Cualquier comando en el que utilice alias con espacios también debe encerrarse entre comillas simples. En el siguiente ejemplo, el alias `lss` hace referencia al comando `ls` con su opción `-s` y el alias `lsa` refiere el comando `ls` con la opción `-F`. El comando `ls` con la opción `-s` muestra una lista de archivos y sus tamaños en bloques, mientras `ls` con la opción `-F` coloca una diagonal tras los nombres de directorio. Observe cómo las comillas simples encierran el comando y su opción.

```
$ alias lss='ls -s'
$ lss
misdatos 14      hoy 6      informes 2
$ alias lsa='ls -F'
$ lsa
misdatos hoy informes/
$
```

Los alias son útiles para simplificar operaciones complejas. En el siguiente ejemplo, `largolista` se convierte en otro nombre para el comando `ls` con la opción `-l` (el formato largo mostrando la lista de toda la información de archivo), además de la opción `-h` para usar una presentación de tamaño de archivos legible para un ser humano. Asegúrese de encerrar el comando y sus argumentos dentro de comillas sencillas para ser tomadas como argumento sin análisis de la shell.

```
$ alias largolista='ls -lh'
$ largolista
-rw-r--r--    1 root    root    51k Sep  18  2003 misdatos
-rw-r--r--    1 root    root    16k Sep  27  2003 hoy
```

Uso de alias en comandos y argumentos

A menudo utilizará un alias para incluir un nombre de comando con un argumento. Si ejecuta un comando con argumento combinado además con caracteres especiales de manera regular, tal vez quiera utilizar un alias. Por ejemplo, suponga que a menudo hace una lista de sólo sus archivos de código fuente y código de objeto (archivos que terminan ya sea en `.c` o `.o`). Tal vez necesite un argumento para la combinación `ls` de un carácter especial como `*.[co]`. En cambio, puede utilizar un alias `ls` con el argumento `*.[co]`, dándole un nombre simple. En el siguiente ejemplo, el usuario crea un alias denominado `lsc` para el comando `ls.[co]`:

```
$ alias lsc='ls *.[co]'
$ lsc
principal.c principal.o lib.c lib.o
```

Comandos de alias

También puede utilizar el nombre de un comando como alias. Esto puede ser práctico en casos que necesita usar un comando sólo con una opción específica. En el caso de comandos **rm**, **cp** y **mv**, siempre debe manejarse la opción **-i** para asegurar que un archivo existente no se sobrescriba. En vez de siempre cuidar el uso de la opción **-i** cada vez que emplee uno de estos comandos, puede recurrir a un alias en el nombre del comando para incluir la opción. En el siguiente ejemplo, a los comandos **rm**, **cp** y **mv** se les ha puesto un alias para incluir la opción **-i**:

```
$ alias rm='rm -i'
$ alias mv='mv -i'
$ alias cp='cp -i'
```

El comando **alias** por sí solo lista todos los alias que han sido definidos, mostrando a su vez los comandos que representa. Puede quitar un alias mediante el comando **unalias**. En el siguiente ejemplo, el usuario muestra una lista de alias actuales y después quita el alias **lsa**:

```
$ alias
lsa=ls -F
list=ls
rm=rm -i
$ unalias lsa
```

Control de operaciones de shell

La shell BASH tiene varias características que permiten controlar la manera en que trabajan diferentes operaciones de shell. Por ejemplo, configurar la característica **noclobber** evita que el redireccionamiento sobrescriba archivos. Puede activar y desactivar estas características activando o desactivando un interruptor, a través del comando **set**. Éste toma dos argumentos: una opción que especifica activado o desactivado y el nombre de la característica. Para activar una característica, use la opción **-o** y para desactivarla puede recurrir a la opción **+o**. Aquí se muestra la forma básica:

```
$ set -o característica activa la característica
$ set +o característica desactiva la característica
```

Tres de las características más comunes son **ignoreeof**, **noclobber** y **noglob**. En la tabla 5-3 se muestra una lista de características diferentes, además del comando **set**. Al configurar **ignoreeof**

Características	Descripción
\$ set -+o característica	Las características de la shell BASH se activan y desactivan con el comando set ; -o activa la característica y +o la desactiva: \$ set -o noclobber activa <i>noclobber</i> \$ set +o noclobber desactiva <i>noclobber</i>
ignoreeof	Desabilita el cierre de sesión con CTRL-D
noclobber	No sobrescribe archivos al redireccionar
noglob	Deshabilita los caracteres especiales empleados para las expansiones de nombres de archivo: * , ? , ~ , y []

TABLA 5-3 Características especiales de la shell BASH

se habilita una característica que evita la salida de una sesión de usuario de shell con CTRL-D. CTRL-D no sólo se utiliza para salir de la sesión de usuario de shell, sino también para terminar la entrada del usuario introducida directamente en la entrada estándar. CTRL-D suele emplearse para el programa Mail o utilerías como **cat**. Fácilmente podría ingresar un CTRL-D adicional en tales circunstancias y terminar su sesión por accidente. La característica **ignoreeof** se activa utilizando el comando **set** con la opción **-o**. El usuario puede salir de su sesión con sólo insertar el comando **logout**.

```
$ set -o ignoreeof
$ CTRL-D
Use exit to logout
$
```

Variables de entorno y subshells: export

Cuando inicia sesión en su cuenta, Linux genera su shell de usuario. En esta shell, puede enviar comandos y declarar variables. También es capaz de crear y ejecutar secuencias de comandos de shell. Sin embargo, cuando ejecuta una secuencia de comandos de shell, el sistema genera una subshell. Después tiene dos shells, con la que inició sesión y la generada para la secuencia de comandos. Dentro de la shell secuencia de comandos, puede ejecutar otra secuencia de comandos shell, que a su vez tiene shell propia. Cuando una secuencia de comandos ha terminado su ejecución, su shell termina y regresa a la shell desde donde se ejecutó. En este sentido, puede tener muchas shells, cada una anidada dentro de otra. Las variables definidas en una shell son locales a ésta. Si define una variable en una secuencia de comandos shell, entonces al ejecutar la secuencia de comandos, la variable se define con esa secuencia de comandos de shell y es local a ésta. Ninguna otra shell puede hacer referencia a esa variable. En ese sentido, la variable está escondida dentro de su shell.

Puede definir variables de entorno en todos los tipos de shell, incluidas BASH, Z y TCSH. Sin embargo, la estrategia empleada para implementar variables de entorno en BASH es diferente de TCSH. En la primera, las variables de entorno se exportan. Es decir, la copia de una variable de entorno se crea en cada subshell. Por ejemplo, si la variable **EDITOR** se exporta, una copia se define automáticamente en cada subshell. En la shell TCSH, por otra parte, una variable de entorno se define sólo una vez y cualquier subshell puede referirse a ella.

En la shell BASH, puede pensarse en una variable de entorno como variable regular con capacidades extra. Para crear una variable de entorno, aplique el comando **export** a una variable ya definida. El comando **export** instruye al sistema para definir una copia de esa variable en cada nueva shell generada. Cada nueva shell tendrá su propia copia de variable de entorno. A este proceso se denomina *exportación de variables*. Es un error pensar en variables de entorno exportadas como variables globales. Una nueva shell nunca puede hacer referencia a una variable fuera de sí misma. En cambio, la nueva shell genera una copia de la variable con su valor.

Configuración de su shell con parámetros shell

Cuando inicie sesión, Linux configurará ciertos parámetros para su shell de inicio de sesión. Estos parámetros pueden tomar la forma de variables o características. Consulte la sección anterior, “Control de operaciones shell”, para conocer una descripción de la forma en que se activan estas características. Linux reserva un conjunto de variables predefinidas para uso de la shell y el sistema. Se trata de valores asignados por el sistema, con lo que se configuran los parámetros. Linux configura variables de parámetros de shell usados para definir su shell de usuario. El sistema

establece muchas de estas variables de parámetros de shell al iniciar sesión. Algunos parámetros de variables de shell se determinan automáticamente por la shell, y otros se configuran por secuencias de comandos de inicialización, a ser descritos después. La shell configura directamente ciertas variables shell y otras sólo son manejadas por scripts de inicialización. Muchas de estas y otras variables son específicas de aplicaciones, utilizadas para tareas como correo, historial o edición. En el aspecto funcional, tal vez sea mejor pensar en éstas como variables al nivel del sistema, utilizadas para configurar todo el sistema, definiendo valores como la ubicación de comandos de ejecución en su sistema, o el número de comandos de historial permitidos. Consulte la tabla 5-4 para conocer una lista de tales variables configuradas por la shell para tareas específicas de ésta; en la tabla 5-5 se muestra una lista de variables utilizadas por la shell para dar soporte a otras aplicaciones.

Un conjunto reservado de palabras clave se utiliza para los nombres de esas variables de sistema. No debe utilizar estas palabras clave como nombre de alguna variable propia. Las variables de shell del sistema se especifican con letras mayúsculas, lo que permite su fácil identificación. Las variables presentadas por la shell van en minúsculas. Por ejemplo, el sistema utiliza la palabra clave **HOME** para definir la variable **HOME**. Se trata de una variable de entorno especial que almacena la ubicación del directorio inicio del usuario. Por otra parte, la palabra clave **noclobber** se utiliza para activar o desactivar la característica noclobber.

Variables de parámetros de shell

Si lo desea, puede cambiar muchas de las variables de parámetro de shell, definidas automáticamente y a las que el sistema asigna valores iniciales. Sin embargo, existen variables de parámetro cuyos valores no pueden cambiarse. Por ejemplo, la variable **HOME** almacena la ruta de su directorio

Variables de Shell	Descripción
BASH	Almacena la ruta completa de un comando BASH
BASH_VERSION	Despliega el número de versión actual de BASH
GROUPS	Grupos a los que pertenece el usuario
HISTCMD	Número de comandos actuales en la lista del historial
HOME	Ruta del directorio de inicio del usuario
HOSTNAME	El nombre del host
HOSTTYPE	Despliega el tipo de maquina en donde se ejecuta el host
OLDPWD	Directorio de trabajo anterior
OSTYPE	Sistema operativo en uso
PATH	Muestra una lista de las rutas para los directorios buscados de comandos ejecutables
PPID	ID del proceso de la shell padre de la shell
PWD	Directorio de trabajo del usuario
RANDOM	Genera un número aleatorio cuando se hace referencia
SHLVL	Nivel de la shell actual, número de shell invocadas
UID	ID de usuario del usuario actual

TABLA 5-4 Variables de shell configuradas por la shell

Variables de shell	Descripción
BASH_VERSION	Despliega el número de versión actual de BASH
CDPATH	Busca la ruta para el comando cd
EXINIT	Comandos de inicialización para el editor Ex/Vi
FCEDIT	Editor utilizado por el comando de historial fc
GROUPS	Grupos a los que pertenece el usuario
HISTFILE	Despliega la ruta del archivo de historial
HISTSIZE	Número de comandos permitidos por el historial
HISTFILESIZE	Tamaño del archivo de historial en líneas
HISTCMD	Número de comandos actuales en la lista del historial
HOME	Ruta del directorio de inicio del usuario
HOSTFILE	Configura el nombre del archivo host, si es diferente de /etc/hosts
IFS	Símbolo de delimitador entre campos
IGNOREEOF	Si no está configurado, el carácter EOF cerrará la shell. Puede configurarse de acuerdo con el número de caracteres EOF que se ignorarán antes de aceptar uno para cerrar la shell (como opción predeterminada es 10)
INPUTRC	Establece el archivo de configuración inputrc para Readline (línea de comandos). La opción predeterminada es el directorio actual, .inputrc . La mayoría de distribuciones Linux configuran éste como /etc/inputrc
KDEDIR	La ubicación de la ruta para el escritorio KDE
LOGNAME	Nombre de inicio de sesión
MAIL	Nombre de un archivo de correo específico, revisado por la utilidad Mail para mensajes recibidos, si MAILPATH no está configurado
MAILCHECK	Intervalo para revisar correo recibido
MAILPATH	Muestra una lista de archivos de correo en que Mail revisará si hay mensajes recibidos
HOSTTYPE	Plataformas de Linux, como i686, x86_64 o ppc
PROMPT_COMMAND	Comando a ejecutarse antes del indicador de comandos, integrando el resultado como parte del indicador de comandos
HISTFILE	La ruta del archivo de historial
PS1	Indicador de comandos de shell primaria
PS2	Indicador de comandos de shell secundaria
QTDIR	Ubicación de la biblioteca Qt (utilizada para KDE)
SHELL	Ruta del programa para el tipo de shell en uso
TERM	Tipo de terminal
TMOUT	El tiempo que la shell permanece activa esperando una entrada
USER	Nombre de usuario
UID	ID de usuario real (numérico)
EUID	ID de usuario efectivo (EUID, numérico). Suele ser el mismo que UID pero puede ser diferente cuando el usuario cambia de ID, que sucede con el comando su , para convertirse en un usuario root efectivo

TABLA 5-5 Variables de entorno de sistema utilizadas por la shell

inicial. Comandos como `cd` hacen referencia a la ruta de la variable de shell **HOME** para localizar su directorio de inicio. Algunas variables de parámetro más comunes se describen en esta sección. El sistema define otras y les da un valor inicial que puede cambiarse con libertad. Para ello, debe redefinirlos y asignar un nuevo valor. Por ejemplo, el sistema define la variable **PATH**, asignándole un valor inicial; contiene la ruta de directorios donde se ubican los comandos. En el momento que quiera ejecutar un comando, la shell lo buscará en estos directorios. Puede agregar uno nuevo en el que desea se hagan búsquedas y redefinir la variable **PATH**, para de ese modo incluir la nueva ruta del directorio. Aún así existen variables de parámetro indefinidas por el sistema. Suele tratarse de características opcionales, como la variable **EXINIT**, para configurar opciones del editor Vi. Cada vez que inicia sesión, debe definir y asignar un valor a dichas variables. Algunas variables de parámetro comunes son **SHELL**, **PATH**, **PS1**, **PS2** y **MAIL**. La variable **SHELL** almacena la ruta del programa para el tipo de shell donde inicia sesión. La variable **PATH** muestra una lista de diferentes directorios en que buscará un comando de Linux. Las variables **PS1** y **PS2** almacenan los símbolos de indicador de comandos. La variable **MAIL** almacena la ruta de su archivo de bandeja de entrada. Puede modificar los valores de cualquiera de estas variables para personalizar su shell.

NOTA También puede obtener una lista de variables de shell definidas mediante el comando `env`. El comando `env` opera como `set`, pero muestra una lista únicamente de las variables de parámetro.

Uso de archivos de inicialización

Puede definir automáticamente variables de parámetro a través de secuencias de comandos de shell, denominadas archivos de inicialización. Un *archivo de inicialización* es una secuencia de comandos de shell denominada especialmente para ejecutarse cuando entra a cierta shell. Puede editar el archivo de inicialización, colocar definiciones en él y asignar variables de parámetros. Apenas ingrese en la shell, el archivo de inicialización ejecutará estas definiciones y asignaciones, activando variables de parámetro con sus propios valores. Por ejemplo, el archivo **.bash_profile** de la shell BASH, es un archivo de inicialización que se ejecuta cada vez que inicia sesión. Contiene definiciones y asignaciones de variables de parámetro. Sin embargo, el archivo **.bash_profile** sólo es, en esencia, una secuencia de comandos shell, que puede editar con cualquier editor de texto como Vi, cambiando, si así lo desea, los valores asignados a las variables de parámetro.

En la shell BASH, todas las variables de parámetro están diseñadas para ser de entorno. Cuando define o redefine una variable de parámetro, también necesita exportarla para hacerla de entorno. Esto significa que cualquier cambio hecho a una variable de parámetro debe acompañarse por un comando **export**. Verá eso al final del archivo de inicialización de inicio de sesión, **.bash_profile**, usualmente hay un comando **export** para todas las variables de parámetro definidas en éste.

Su directorio inicio: HOME

La variable **HOME** contiene una ruta para su directorio de inicio. El administrador de parámetros es quien determina el directorio inicio cuando se crea su cuenta. La ruta para su directorio de inicio se lee automáticamente desde su variable **HOME** al iniciar sesión. En el siguiente ejemplo, el comando **echo** despliega el contenido de la variable **HOME**.

```
$ echo $HOME
/home/chris
```

La variable **HOME** a menudo se usa cuando necesita especificar la ruta absoluta para su directorio inicio. En el siguiente ejemplo, la ruta absoluta de **informes** se especifica mediante **HOME**, para la ruta del directorio inicio:

```
$ ls $HOME/informes
```

Ubicaciones de comando: PATH

La variable **PATH** contiene una serie de rutas de directorio separadas por dos puntos. Cada vez que ejecuta un comando, se busca éste en cada una de las rutas que mostradas en la lista de la variable **PATH**. Por ejemplo, el comando **cp** reside en su sistema en el directorio **/bin**. Esta ruta es uno de los directorios mostrados en la lista de la variable **PATH**. Cada vez que ejecuta el comando **cp**, se busca esta y localiza el comando **cp**. El sistema define y asigna **PATH** a un conjunto inicial de rutas. En Linux, las rutas iniciales son **/bin** y **/usr/bin**.

La shell puede activar cualquier archivo ejecutable, incluidos programas y secuencias de comandos creados. Por esta razón, la variable **PATH** también puede hacer referencia a su directorio de trabajo; de modo que si quiere ejecutar una de sus propias secuencias de comandos o programas en su directorio de trabajo, la shell podrá localizarla. No se permiten espacios entre rutas en la cadena de comandos. Los dos puntos, sin ruta especificada, hacen referencia a su directorio de trabajo. Generalmente, se colocan dos puntos al final de las rutas, como una entrada vacía especificando su directorio de trabajo. Por ejemplo, la ruta **//bin:/usr/bin:** refiere tres directorios: **/bin**, **/usr/bin** y su directorio de trabajo actual:

```
$ echo $ PATH
/bin:/usr/sbin:
```

Puede agregar cualquier nueva ruta de directorio que desee a una variable **PATH**. Esto es útil si ha creado varios comandos propios de Linux usando secuencias de comandos. Puede colocar estos nuevos comandos de la secuencia de comandos de shell en un directorio para crear y luego agregar ese directorio a la lista **PATH**. Después, sin importar en que directorio esté, puede ejecutar una de sus secuencias de comandos de shell. La variable **PATH** contendrá el directorio para esa secuencia de comandos, de modo tal que se busque en el directorio cada vez que envíe un comando.

Puede agregar un directorio a la variable **PATH** con una asignación de variable y así ejecutar esta asignación directamente en su shell. En el siguiente ejemplo, el usuario **chris** agrega un nuevo directorio, denominado **misbin** a **PATH**. Aunque puede escribir de manera cuidadosa las rutas completas mostradas en la lista **PATH** para la asignación, también puede emplear una evaluación de **PATH** (**\$PATH**) en su lugar. En el siguiente ejemplo, se utiliza también una evaluación de **HOME** para designar el directorio home del usuario en la ruta del nuevo directorio. Observe la entrada vacía entre los dos puntos, especificando el directorio de trabajo:

```
$ PATH=$PATH:$HOME/misbin:
$ export PATH
$ echo $PATH
/bin:/usr/bin:~/home/chris/misbin
```

Si agrega un directorio a **PATH** mientras está dentro de su sesión, el directorio se agregará sólo durante el tiempo que se encuentre en sesión. Cuando inicie sesión de nuevo, el archivo de inicialización para iniciar de sesión, **.bash_profile**, de nuevo volverá a inicializar su **PATH** con el conjunto de directorios original.

El archivo **.bash_profile** es descrito de manera detallada más adelante en este capítulo. Para agregar permanentemente un directorio a **PATH**, necesita editar su archivo **.bash_profile** y encontrar la asignación de la variable **PATH**. Luego, simplemente inserte el directorio, con dos puntos antes, en el conjunto de rutas asignadas a **PATH**.

Especificación del entorno BASH: BASH_ENV

La variable **BASH_ENV** almacena el nombre del archivo de inicialización de la shell BASH para ser ejecutado cuando genere una shell BASH. Por ejemplo, al ejecutar una secuencia de comandos de shell BASH, la variable **BASH_ENV** se revisa y el nombre de la secuencia de comandos almacenada se ejecuta antes de la secuencia de comandos de shell. La variable **BASH_ENV** generalmente incluye **\$HOME/.bashrc**. Éste es el archivo **.bashrc** en el directorio de inicio del usuario. (El archivo **.bashrc** se analizará más adelante en este capítulo.) Puede especificar un archivo diferente, si así lo desea, utilizándolo en vez del archivo **.bashrc** para las secuencias de comandos shell de BASH.

Configuración del indicador de comandos de shell

Las variables **PS1** y **PS2** contienen los símbolos de indicador de comandos primarios y secundarios, respectivamente. El símbolo del indicador de comandos primario para la shell BASH es un signo de moneda (**\$**). Puede cambiar el símbolo asignando un nuevo conjunto de caracteres a la variable **PS1**. En el siguiente ejemplo, el indicador de comandos de shell se cambia por el símbolo **->**:

```
$ PS1= '->'
-> export PS1
->
```

Puede cambiar el indicador de comandos a cualquier conjunto de caracteres, incluida una cadena de comandos, como se muestra en el siguiente ejemplo:

```
$ PS1="Por favor inserte un comando: "
Por favor inserte un comando: export PS1
Por favor inserte un comando: ls
misdatos /informes
Por favor inserte un comando:
```

La variable **PS2** almacena el símbolo de indicador de comandos secundario, usado para comandos que requieren varias líneas para completarse. El indicador de comandos secundario predeterminado es **>**. Las líneas de comandos agregadas empiezan con el indicador de comandos secundario en vez del indicador primario. Puede cambiar el indicador de comandos secundario tan fácil como el indicador de comandos primario, como se muestra aquí:

```
$ PS2="@@"
```

Al igual que la shell TCSH, BASH ofrece un conjunto predefinido de códigos que puede utilizar para configurar su indicador de comandos. Con éstos puede hacer que hora, nombre de usuario o ruta de directorio sean parte de su indicador de comandos. Puede incluso hacer que su indicador de comandos despliegue el número de evento de historial del comando actual por insertar. Cada código va precedido de un símbolo ****: **\w** representa el directorio de trabajo actual, **\t** hora y **\u** nombre de usuario; **\!** desplegará el siguiente número de evento de historial. En el siguiente ejemplo, el usuario agrega el directorio de trabajo actual para el indicador de comandos:

```
$ PS1="\w $"
/home/dylan $
```

Los códigos deben incluirse en una cadena con comillas. Si no existe, los caracteres de código no se evalúan y se usan como indicador de comandos. `PS1=\w` asigna los caracteres `\w` al indicador de comando, no el directorio de trabajo. En el siguiente ejemplo se incorporan hora y número de evento de historial con el nuevo indicador de comandos:

```
$ PS1="\t \! ->"
```

En la siguiente tabla se muestra una lista de códigos para configuración de su indicador de comandos:

Códigos de indicador de comando	Descripción
\!	Número de historial actual
\\$	Utiliza <code>\$</code> como indicador de comando para todos los usuarios excepto root, que tiene <code>#</code> como indicador de comando
\d	Fecha actual
\#	Número de comando de historial para la shell actual
\h	Nombre de host
\s	Tipo de shell activa
\t	Hora en horas, minutos y segundos
\u	Nombre de usuario
\v	Versión de shell
\w	Nombre de ruta completo del directorio de trabajo actual
\W	Nombre del directorio de trabajo actual
\\	Despliega un carácter de diagonal invertida
\n	Inserta una nueva línea
\[\]	Permite la entrada de caracteres de despliegue específicos de la terminal para características como color o negritas
\nnn	Carácter especificado en formato octal

El indicador de comandos predeterminado de BASH es `\s-\v\$` para desplegar el tipo de shell, versión y el símbolo `$` como indicador de comandos. Algunas distribuciones, como Fedora y Red Hat, han cambiado esto a un comando más complejo constando de usuario, nombre de host y nombre del directorio de trabajo actual. La operación actual se lleva a cabo en el archivo `/etc/bashrc`, analizada más adelante en la sección “La secuencia de comandos de sistema BASH `/etc/bashrc` y el directorio `/etc/profile.d`”. Aquí se muestra un ejemplo de configuración. El archivo `/etc/bashrc` usa las variables de entorno `USER`, `HOSTNAME` y `PWD` para configurar dichos valores. Un equivalente simple se muestra aquí con un signo `@` en el nombre de host y `$` para el símbolo de indicador de comandos final. El directorio de inicio se representa con una tilde (`~`).

```
$ PS1="\u@\h:\w$"  
richard@turtle.com:~$
```

Especificación de su servidor de noticias

Diversas variables de parámetros de shell se usan para configurar valores manejados por las aplicaciones de red, como exploradores Web o lectores de noticias. `NNTPSERVER` se utiliza para

configurar el valor de un servidor de noticias remoto accesible en su red. Si está utilizando un ISP, éste proporciona un servidor de noticias Usenet, al que puede acceder a través de su aplicación de lector de noticias. Sin embargo, primero debe brindar a su lector la dirección Internet del servidor de noticias. Este es el papel de la variable **NNTPSERVER**. Los servidores de noticias en Internet suelen utilizar un protocolo **NNTPSERVER** debe almacenar la dirección de dicho servidor de noticias. En el caso de muchos ISP, la dirección del servidor de noticias es un nombre de dominio que da inicio con **nntp**. En el siguiente ejemplo se asigna la dirección del servidor de noticias **nntp.miservicio.com** a la variable shell **NNTPSERVER**. Las aplicaciones de lector de noticias obtienen automáticamente la dirección del servidor de noticias de **NNTPSERVER**. Generalmente, esta asignación se coloca en el archivo de inicialización de **shell**, **.bash_profile**, para activarse automáticamente siempre que un usuario inicia sesión.

```
NNTPSERVER=news.miservicio.com
export NNTPSERVER
```

Configuración de su shell de inicio de sesión: **.bash_profile**

El archivo **.bash_profile** es un archivo de inicialización de inicio de sesión de la shell BASH, que también puede llamarse **.profile** (como en Linux SUSE o UBUNTU). Es un archivo de secuencia de comandos en ejecución automática cuando un usuario inicia sesión. El archivo contiene comandos de shell definiendo las variables de entorno de sistema usadas para administrar su shell. Pueden ser redefiniciones de variables establecidas por el sistema o definiciones de variables asignadas por el usuario. Por ejemplo, cuando inicia sesión, su shell de usuario necesita saber en qué directorios se almacenan los comandos de Linux. Hará referencia a la variable **PATH** para encontrar nombres de ruta de esos directorios. Sin embargo, primero, la variable **PATH** debe asignarse a esos nombres de ruta. En el archivo **.bash_profile**, una operación de asignación hace esto. Debido a que está en el archivo **.bash_profile**, la asignación se ejecuta automáticamente cuando el usuario inicia sesión.

Exportación de variables

Las variables de parámetro también deben ser exportadas, mediante el comando **export**, para hacerlas accesibles a cualquier subshell que pueda insertar. Puede exportar varias variables en un comando **export** haciendo una lista de éstas como argumentos. generalmente, el archivo **.bash_profile** termina con **export** y una lista de todas las variables definidas en el archivo. Si falta una variable en esta lista, tal vez no sea capaz de acceder a ella. Observe el comando **export** al final del archivo **.profile** en el primer ejemplo de la siguiente sección. También puede combinar asignación y **export** en una operación, como se muestra aquí para **NNTPSERVER**:

```
export NNTPSERVER=news.miservicio.com
```

Asignación de variables

En la lista del siguiente ejemplo, se presenta una copia del archivo **.bash_profile** estándar, proporcionada cuando se crea una cuenta. Observe que **PATH** se asigna como si fuera el valor de **\$HOME**. Ambos, **PATH** y **HOME**, son variables de parámetro ya definidas por el sistema. **PATH** almacena los nombres de rutas de los directorios buscados por cualquier comando ingresado y **HOME** almacena el nombre de ruta para su directorio de inicio. La asignación **PATH=\$PATH:\$HOME/bin** tiene el efecto de redefinir **PATH** para incluir su directorio bin en el directorio de inicio, para también buscar en su directorio bin cualquier comando, incluidos los que usted mismo creó, como secuencias de comandos o programas. Observe que después se exporta **PATH**, para tener acceso a él desde cualquier subshell.

```
.bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User-specific environment and startup programs
PATH=$PATH:$HOME/bin
export PATH
```

La versión del usuario **root** de **.bash_profile** agrega una entrada para desactivar la variable **NOMBREUSUARIO**, conteniendo el nombre de texto del usuario.

```
unset NOMBREUSUARIO
```

Si desea que también se busque en su directorio de inicio, puede usar cualquier editor de texto para modificar esta línea en su archivo **.bash_profile** por **PATH=\$PATH:\$HOME/bin: \$HOME**, agregando **\$HOME** al final. En realidad, puede cambiar esta entrada para agregar los directorios en que quiere se hagan búsquedas. Si agrega dos puntos al final, entonces también se buscarán comandos en su directorio de trabajo actual. Crear comandos de ejecución automática en su directorio de trabajo actual puede ser un riesgo de seguridad, permitiendo que archivos en cualquier directorio se ejecuten, en vez de ciertos directorios especificados. En la siguiente sección se muestra un ejemplo de cómo modificar su archivo **.bash_profile**.

```
PATH=$PATH:$HOME/bin:$HOME:
```

Edición de la secuencia de comandos de su perfil de BASH

Su archivo de inicialización **.bash_profile** es un archivo de texto susceptible de modificarse con un editor de texto, como cualquier otro archivo. Puede agregar nuevos directorios de manera sencilla a su **PATH** editando **.bash_profile** y usar los comandos de edición para insertar un nuevo nombre de ruta del directorio en la lista de nombres asignados a la variable **PATH**. Incluso puede agregar nuevas definiciones de variables. Sin embargo, si lo hace, asegúrese de incluir el nuevo nombre de la variable en la lista de argumentos del comando **export**. Por ejemplo, si su archivo **.bash_profile** no tiene definición de la variable **EXINIT**, puede editar el archivo y agregar una nueva línea para asignar un valor a **EXINIT**. La definición **EXINIT='set nu ai'** configurará el editor Vi con numeración de línea y sangrías. Después necesita agregar **EXINIT** a la lista de argumentos del comando **export**. Cuando se ejecuta de nuevo el archivo **.bash_profile**, la variable **EXINIT** tendrá el valor del comando **set nu ai**. Cuando invoque el editor Vi, se ejecutará el comando en la variable **EXINIT**, configurando el número de línea y se agregarán sangrías automáticamente a las opciones.

En el siguiente ejemplo, se ha modificado el **.bash_profile** del usuario para incluir definiciones **EXINIT** y redefiniciones de **PATH**, **PS1** y **HISTSIZE**. La variable **PATH** tiene **\$HOME**: agregada a su valor. **\$HOME** es una variable que evalúa el directorio de inicio del usuario, y los dos puntos finales especifican el directorio de trabajo actual, permitiéndole ejecutar comandos localizados en el directorio de inicio o de trabajo. La redefinición de **HISTSIZE** reduce el número de eventos de historial guardados, desde 1000 definidos en el archivo **.profile** del sistema, hasta 30. La redefinición de la variable de parámetro **PS1** cambiará el indicador de comandos para incluir el nombre de ruta de su directorio de trabajo actual. Cualquier cambio hecho a estas variables de parámetro en el

archivo **.bash_profile** supera las hechas antes, en el archivo **.profile** del sistema. Después, todas las variables de parámetro se exportan con el comando **export**.

```
.bash_profile
# .bash_profile
# Get the aliases and functions
if [ -f ~/.bashrc ];
then
    . ~/.bashrc
fi
# User-specific environment and startup programs
PATH=$PATH:$HOME/bin:$HOME:
unset NOMBREUSUARIO
HISTSIZE=30
NNTPSERVER=news.miservidor.com
EXINIT='set nu ai'
PS1="\w \$"
export PATH HISTSIZE EXINIT PS1 NNTPSERVER
```

Nueva ejecución manual de la secuencia de comandos **.bash_profile**

A pesar de que **.bash_profile** se ejecuta siempre que inicia sesión, no se vuelve a ejecutar automáticamente después de hacer cambios a éste. El archivo **.bash_profile** es de inicialización que se ejecuta *sólo* al iniciar sesión. Si quiere aprovechar cualquier cambio hecho, sin salir de la sesión e iniciando sesión de nuevo, puede ejecutar otra vez **.bash_profile** con el comando punto (**.**). El archivo **.bash_profile** es una secuencia de comandos de shell y, como cualquier otra secuencia de comandos de shell, puede ser ejecutarse con el comando (**.**).

```
$ . .bash_profile
```

Como opción, puede usar el comando **source** para ejecutar el archivo de inicialización **.bash_profile** o cualquier archivo de inicialización como **.login** utilizado en la shell TCSH o **.bashrc**.

```
$ source .bash_profile
```

Secuencia de comandos de perfil de la shell del sistema

Su sistema Linux también tiene archivo de perfil propio, ejecutable cuando cualquier usuario inicia sesión. Este archivo de inicialización de sistema se denomina **profile** y se encuentra en el directorio **/etc**, **/etc/profile**. Contiene definiciones de variable de parámetros que el sistema necesita proporcionar a cada usuario. Una copia del archivo **profile** del sistema se presenta al final de esta sección. En algunas distribuciones, será un archivo realmente simple y en otras mucho más complejo. Algunas distribuciones como Fedora y Red Hat usan una función **pathmunge** para generar una lista de directorios para la variable **PATH**. Las rutas de usuario normal no tendrán directorios de sistema (las incluidas en la ruta **sbin**) pero comprendiendo el nombre de su directorio home, junto con **/usr/kerberos/bin**, para herramientas Kerberos. La ruta generada para el usuario root (EUID de 0) incluirá directorios de sistema y aplicación de usuario, agregando **/usr/kerberos/sbin**, **/sbin**, **/usr/sbin** y **/usr/local/sbin**, además del directorio de aplicación local del usuario **root**, **/root/bin**.

```
# echo $PATH
/usr/kerberos/bin/usr/local/bin:usr/sbin:/bin:/usr/X11R6/bin:/home/richard/bin
```


Un enfoque especial se incluye en la shell Korn para configurar los ID de User y Effective User IDs (**EUID** y **UID**).

Luego se establecen las variables **USER**, **MAIL** y **LOGNAME**, siempre y cuando **/usr/bin/id**, proporcionando el ID de usuario, sea ejecutable. El comando **id** con la opción **-un**, sólo despliega el nombre de texto del ID de usuario, como **chris** o **richard**.

También se redefine **HISTSIZE** para incluir mayor número de eventos de historial. Aquí se ha agregado una entrada para la variable **NNTPSERVER**. Generalmente, una dirección de servidor de noticias es un valor que necesita configurarse para todos los usuarios. El administrador de sistema debe hacer tales asignaciones en el archivo **/etc/profile** del sistema, en vez de hacerlas en cada archivo **.bash_profile** de usuario.

***NOTA** El archivo **/etc/profile** también ejecuta cualquier secuencia de comandos en el directorio **/etc/profile.d**. El diseño permite una estructura modular. En vez de crear entradas editando el archivo **/etc/profile**, puede agregar una secuencia de comandos al directorio **profile.d**.*

El archivo **/etc/profile** también ejecuta el archivo **/etc/inputrc**, que configura su editor de línea de comandos. Aquí encontrará asignaciones clave para tareas diferentes, como ir al final de la línea o eliminar caracteres. También se configuran las opciones globales. Las claves se representan en formato hexadecimal.

El número de configuraciones de alias y variables necesarias para diferentes aplicaciones, harán que el archivo **/etc/profile** sea muy largo y difícil de administrar. En vez de eso, se colocan alias específicos de aplicación y tareas, así como variables en archivos de configuración separados, localizados en el directorio **/etc/profile.d**. Existen secuencias de comandos correspondientes para las shell BASH y C. Las secuencias de comandos de la shell BASH se ejecutan mediante **/etc/profile**. Los nombres de secuencias de comandos se asignan acorde con los tipos de tareas y aplicaciones en configuración. Por ejemplo, para Red Hat, configura la codificación de archivo para tipo de color, cuando el comando **ls** despliega archivos y directorios. El archivo **vim.sh** configura el alias para el comando **vi**, ejecutando vim cuando el usuario sólo inserte **vi**. El archivo **kde.sh** maneja la variable de entorno global **KDEDIR**, especificando el directorio de aplicaciones de KDE, en este caso **/usr**. El archivo **krb5.sh** agrega nombres de rutas de Kerberos, **/usr/kerberos**, a la variable **PATH**. Los archivos ejecutados por la shell BASH terminan con la extensión **.sh**, y los ejecutados en la shell C tienen la extensión **.csh**.

```
/etc/profile
# /etc/profile

# Systemwide environment and startup programs, for login setup
# Functions and aliases go in /etc/bashrc

pathmunge () {
    if ! echo $PATH | /bin/egrep -q "(^|:)$1($|:)" ; then
        if [ "$2" = "after" ] ; then
            PATH=$PATH:$1
        else
            PATH=$1:$PATH
        fi
    fi
}

# ksh workaround (Atajo ksh)
if [ -z "$EUID" -a -x /usr/bin/id ]; then
```



```

        EUID=`id -u`
        UID=`id -ru`
    fi

    # Path manipulation
    if [ "$EUID" = "0" ]; then
        pathmunge /sbin
        pathmunge /usr/sbin
        pathmunge /usr/local/sbin
    fi

    # No core files by default
    ulimit -S -c 0 > /dev/null 2>&1

    if [ -x /usr/bin/id ]; then
        USER=`id -un`
        LOGNAME=$USER
        MAIL="/var/spool/mail/$USER"
    fi

    HOSTNAME=`/bin/hostname`
    HISTSIZE=1000

    if [ -z "$INPUTRC" -a ! -f "$HOME/.inputrc" ]; then
        INPUTRC=/etc/inputrc
    fi

    export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE INPUTRC

    for i in /etc/profile.d/*.sh ; do
        if [ -r "$i" ]; then
            . $i
        fi
    done

    unset i
    unset pathmunge

```

Configuración de la shell BASH: .bashrc

El archivo **.bashrc** es de configuración y se ejecuta siempre que ingresa en la shell BASH o genera una subshell. Si la shell BASH es su shell de inicio de sesión, **.bashrc** se ejecuta junto con su archivo **.bash_login** al iniciar sesión. Si inserta la shell BASH desde otra shell, el archivo **.bashrc** se ejecuta automáticamente, y establecerán las definiciones de variables y alias contenidas. Si inserta un tipo diferente de shell, se ejecutará el archivo de configuración para esa shell. Por ejemplo, si fuera a insertar la shell TCSH con el comando `tcsh`, se ejecutará el archivo de configuración **.tcshrc** en vez de **.bashrc**.

La secuencia de comandos BASH .bashrc de usuario

El archivo de configuración **shell .bashrc**, realmente se ejecuta cada vez que genera una shell BASH, al ejecutar una secuencia de comandos de shell. En otras palabras, cada vez que se crea una subshell, el archivo **.bashrc** se ejecuta. Esto tiene el efecto de exportar cualquier variable local o

alias definido en el archivo de inicialización de la shell **.bashrc**. El archivo **.bashrc** suele contener definición de alias y cualquier característica de variable empleada para activar determinadas funciones de shell. Los alias y variables de características se definen de manera local dentro de la shell. Pero el archivo **.bashrc** los define en cada shell. Por esta razón, el archivo **.bashrc** suele almacenar alias y opciones que quiere definir para cada shell. En este ejemplo, el **.bashrc** estándar para usuarios sólo incluye ejecución del archivo `/etc/bashrc` de sistema. Como ejemplo de la manera en que puede agregar sus propios alias y opciones, se han añadido alias de los comandos **rm**, **cp** y **mv** y las opciones **noclobber** y **ignoreeof** de shell. Para el usuario **root** **.bashrc**, ya se han incluido los alias **rm**, **cp** y **mv** en el archivo **.bashrc** de **root**.

```
.bashrc
# Source global definitions
if [ -f /etc/bashrc ];
then
    . /etc/bashrc
fi
set -o ignoreeof
set -o noclobber
alias rm='rm -i'
alias mv='mv -i'
alias cp='cp -i'
```

Puede agregar cualquier comando o definición propios a su archivo **.bashrc**. Si ha hecho cambios a **.bashrc** y quiere tomen efecto durante su sesión, necesita ejecutar de nuevo el archivo con el comando `. o source`.

```
$ . .bashrc
```

La secuencia de comandos de BASH /etc/bashrc del sistema y el directorio /etc/profile.d

En general, los sistemas Linux contienen un archivo **bashrc** para el sistema, ejecutado por todos los usuarios. Este contiene ciertos alias globales y parámetros necesarios para todos los usuarios, cuando ingresan en una shell BASH. Esto se ubica dentro del directorio **/etc**, en **/etc/bashrc**. Un archivo **.bashrc** propio del usuario, ubicado en el directorio home, contiene comandos para ejecutar este archivo **.bashrc** de sistema. El comando `./etc/bashrc` en el ejemplo previo de **.bashrc**, hace exactamente eso. Actualmente el archivo **/etc/bashrc** configura el indicador de comando shell predeterminado, uno para una ventana terminal y otro para una interfaz de pantalla. Después se configuran alias y variables especializados al utilizar archivos de configuración ubicados en el directorio **/etc/profile.d**. **/etc/bashrc** ejecuta estas secuencias de comandos, si la shell no es de inicio de sesión.

El archivo de cierre de sesión de la shell BASH: **.bash_logout**

El archivo **.bash_logout** también es de configuración, pero se ejecuta cuando el usuario cierra la sesión. Se diseña para efectuar cualquier operación que quiera se realice al cerrar sesión. En vez de las definiciones de variable, el archivo **.bash_logout** contiene comandos de shell que forman un tipo de procedimiento de apagado (acciones que siempre quiere se realicen antes de cerrar su sesión). Un comando de cierre de sesión común consiste en limpiar la pantalla y después enviar un mensaje de despedida.

Al igual que con **.bash_profile**, puede agregar comandos propios de shell a **.bash_logout**. En realidad, el archivo **.bash_logout** no se configura automáticamente cuando se crea su cuenta.

Necesita crearlo usted mismo, utilizando el editor Vi o Emacs. Luego, puede agregar un mensaje de despedida u otras aplicaciones. En el siguiente ejemplo, el usuario tiene los comandos **clear** y **echo** en el archivo **.bash_logout**. Cuando el usuario cierra su sesión, **clear** limpia la pantalla, y después **echo** despliega el mensaje “Adios por ahora.”

```
.bash_logout
# ~/.bash_logout
clear
echo "Adios por ahora"
```

La configuración de la shell TCSH

La shell TCSH es, en esencia, una versión de la shell C con características adicionales. Las operaciones de configuración se realizan de manera muy parecida las mismas tareas, pero con sintaxis ligeramente diferente. El comando **alias** opera de la misma forma, aunque usa un formato de comandos diferente. A las variables de sistema se les asignan valores usando operadores de asignación de la shell TCSH, así como los archivos de inicialización y configuración tienen nombres diferentes.

Alias de TCSH y C

El comando **alias** se utiliza para crear otro nombre para un comando. El alias opera como macro que expande al comando que representa. El alias no reemplaza literalmente el nombre del comando; sólo da otro nombre a ese comando.

Un comando **alias** comienza con la palabra clave **alias** y el nuevo nombre del comando, seguido por el comando al que hará referencia el alias. En el siguiente ejemplo, se crea un alias para el comando **ls** con el nombre **lista**, que se convierte en otro nombre para el comando **ls**.

```
> alias lista ls
> ls
misdatos intro
> lista
misdatos intro
>
```

En caso de que el comando al que está asignando un alias tenga opciones, necesita encerrar comando y opción entre comillas simples. Un comando con alias que tiene espacios, también necesitará comillas. En el siguiente ejemplo, a **ls** con la opción **-l** se le asigna el alias **largol**:

```
> alias largol 'ls -l'
> ls -l
-rw-r--r--  1  chris weather 207 Feb  20  11:55  misdatos
> largol
-rw-r--r--  1  chris weather 207 Feb  20  11:55  misdatos
>
```

También puede utilizar el nombre de un comando como alias. En el caso de los comandos **rm**, **cp** y **mv**, siempre debe manejarse la opción **-i**, para garantizar que el archivo existente no se sobrescriba. En vez de estar obligado a siempre usar la opción **-i** con todo cuidado, cada vez que emplea uno de estos comandos, tal vez deba asignar un alias al nombre del comando para

que incluya la opción. En el siguiente ejemplo, a los comandos **rm**, **cp** y **mv** se les ha dado alias para que incluyan la opción **-i**.

```
> alias rm='rm -i'
> alias mv='mv -i'
> alias cp='cp -i'
```

El comando **alias** por si sólo proporciona una lista de todos los alias activos y sus comandos. Un alias puede quitarse con el comando **unalias**.

```
> alias
lss    ls -s
list   ls
rm     ls -i
> unalias lss
```

Variables de características de las shell TCSH y C: características de shell

La shell TCSH tiene varias características que permiten controlar la manera en que trabajan diferentes operaciones de shell. Las características de TCSH incluyen las de la shell PDSKH y muchas propias. Por ejemplo, la shell TCSH tiene una opción **noclobber** para evitar que el redireccionamiento sobrescriba archivos. Algunas de las características más comunes utilizadas son **echo**, **noclobber**, **ignoreeof** y **noglob**. Las características de la shell TCSH se activan o desactivan definiendo o dejando sin definir una variable asociada con esa característica. Se asigna un nombre a una variable para cada característica; por ejemplo, **noclobber** se activa al definir la variable **noclobber**. Puede utilizar el comando **set** para definir una variable y el comando **unset** para dejar de definirla. Para activar la característica **noclobber** puede enviar el comando **set noclobber**. Apáguela mediante el comando **unset noclobber**.

```
> set variable-característica
> unset variable-característica
```

Estas variables también son conocidas como interruptores, porque se manejan para activar o desactivar características.

echo

Al configurar **echo** se activa una característica que despliega un comando antes de ser ejecutado. El comando **set echo** activa la característica **echo** y **unset echo** la desactiva.

ignoreeof

Al configurar **ignoreeof** se activa una característica que evita el usuario salga de la sesión de una shell de usuario con CTRL-D. Está diseñada para evitar cierres de sesión accidentales. Con esta característica desactivada, puede salir de su sesión al oprimir CTRL-D. Sin embargo, CTRL-D también se utiliza para terminar la entrada de usuario, ingresada directamente en la entrada estándar. A menudo se emplea para el programa Mail o utilerías como cat. Es fácil ingresar un CTRL-D adicional en dichas circunstancias, con lo que saldrá por accidente de su sesión. La característica **ignoreeof** evita dichos cierres de sesión accidentales. Cuando está activada, tiene que salir de su sesión explícitamente, al utilizar el comando **logout**:

```
$ set ignoreeof
$ ^D
Use logout to logout
$
```

noclobber

Al configurar **noclobber** se activa una característica que protege de manera segura los archivos existentes de la salida redireccionada. Con la característica **noclobber**, si redirecciona la salida a un archivo ya existente, no se sobrescribirá con la salida estándar. El archivo original se conservará. Se presentan situaciones en que utiliza un nombre ya ha dado a un archivo existente, como nombre para el archivo que almacenará la salida redireccionada. La característica **noclobber** evita la sobrescritura por accidente de un archivo original:

```
> set noclobber
> cat prefacio > miarchivo
miarchivo: file exists
$
```

En ocasiones querrá sobrescribir un archivo con la salida redireccionada. En este caso, puede colocar un signo de admiración tras el operador de redirección. Esto omite la característica **noclobber**, reemplazando el contenido del archivo con la salida estándar:

```
> cat prefacio >! miarchivo
```

noglob

Al configurar **noglob** se activa una característica que deshabilita caracteres especiales en la shell del usuario. Los caracteres ***, ?, [] y ~** ya no se expandirán a nombres de archivos relacionados. Esta característica es útil si, por alguna razón, tiene caracteres especiales como parte de un nombre de archivo. En el siguiente ejemplo, el usuario necesita referirse al archivo que termina con el carácter **¿**, **¿respuestas?**. En primer lugar, el usuario desactiva los caracteres especiales, utilizando la opción **noglob**. Ahora el signo de interrogación en la línea de comandos se toma como parte de un nombre de archivo, no como carácter especial y el usuario puede hacer referencia al archivo **¿respuestas?**

```
$ set noglob
$ ls ¿respuestas?
¿respuestas?
```

Variables de shell especiales de TCSH y C para configurar su sistema

Como en la shell BASH, puede utilizar variables de shell especiales en la shell TCSH para configurar su sistema. Su sistema define algunas inicialmente y después puede redefinirlas con un nuevo valor. Existen otras que usted debe definir inicialmente. Una de las variables especiales utilizadas con más frecuencia es **prompt**, para crear su propio indicador en la línea de comandos. Otra es **history**, para determinar a cuántos eventos de historial quiere dar seguimiento.

En la shell TCSH, muchas variables especiales tienen nombres y funciones parecidas a las de BASH o Public Domain Korn Shell (PDKSH). Algunas van en mayúsculas, pero casi todas se escriben en minúsculas. Las variables **EXINIT** y **TERM** retienen forma de mayúsculas. Sin embargo, **history** y **cdpath** van en minúsculas. Otras variables especiales pueden realizar funciones parecidas, pero tienen implementaciones muy diferentes. Por ejemplo, la variable **mail** almacena la misma información que BASH **MAIL**, **MAILPATH** y **MAILCHECK** juntas.

prompt, prompt2, prompt3

Las variables **prompt**, **prompt2** y **prompt3** almacenan indicadores de su línea de comandos. Puede configurar su indicador de comando para desplegar cualquier símbolo o cadena de comandos que quiera. Para que, en su línea de comandos, se despliegue un símbolo diferente como indicador

de comandos, sólo utilice el comando **set** para asignar ese símbolo a la variable **prompt**. En el siguiente ejemplo, el usuario asigna un signo + a la variable **prompt**, haciéndolo el nuevo signo de indicador de comandos.

```
> set prompt = "+"
+
```

Puede usar un conjunto de códigos predeterminado para que la configuración de su indicador de comandos sea más sencilla. Con este conjunto, puede hacer que hora, nombre de usuario o nombre de ruta de su directorio sea parte del indicador de comandos. Incluso puede hacer que su indicador de comandos despliegue el número de evento de historial de comandos que va a insertar. Cada código lleva un símbolo % antes; por ejemplo, %/ despliega el siguiente número de evento de historial. En el siguiente ejemplo, el usuario agrega su directorio de trabajo al indicador de comandos.

```
> set prompt = "%/ >"
/home/dylan >
```

En el siguiente ejemplo se incorporan hora y número de evento de historial, al nuevo indicador de comandos.

```
> set prompt = "%t %! $"
```

Aquí se muestra una lista de los códigos:

%/	Directorio de trabajo actual
%h, %!, !	Número de historial actual
%t	Hora
%n	Nombre de usuario
%d	Día de la semana
%w	Mes actual
%y	Año actual

La variable **prompt2** es utilizada en casos especiales cuando un comando utiliza varias líneas para salir. **prompt2** se despliega para las líneas extra requeridas por el comando. **prompt3** es el indicador de comando que se emplea si la característica spell check se encuentra activada.

cdpath

La variable **cdpath** almacena nombres de ruta de directorios en que habrán de buscarse subdirectorios específicos a los que se hace referencia con el comando **cd**. Estos nombres de ruta forman una matriz similar a la de nombres de ruta asignados a la variable **path** de la shell TCSH. Observe el espacio entre los nombres de ruta.

```
> set cdpath=(/usr/chris/informes /usr/chris/cartas)
```

history and savehist

Como ya aprendió, la variable **history** se usa para determinar el número de eventos de historial que quiere guardar. Sólo debe asignar el número máximo de eventos que registrará **history**. Cuando alcance el número máximo, la cuenta comienza de nuevo desde 1. Sin embargo, la variable **savehist** almacena el número de eventos que se guardarán en el archivo **.history** cuando salga de la sesión. Al iniciar sesión nuevamente, estos eventos se convertirán en la lista inicial.

En el siguiente ejemplo se guardarán hasta 20 eventos en la lista de historial, mientras esté en su sesión. Sin embargo, sólo se guardarán 5 en el archivo **.history** cuando salga de su sesión. Una vez vuelva a iniciar sesión, su lista de historial estará conformada por los últimos 5 comandos de la sesión previa.

```
> set history=20
> set savehist=5
```

mail

En la shell TCSH, la variable **mail** combina la característica de las variables **MAIL**, **MAILCHECK** y **MAILPATH**, de las shells BASH y PDKSH. La variable **mail** de la shell TCSH tiene asignado como valor una matriz cuyos elementos contienen el intervalo para revisar un correo y los nombres de ruta de directorio, para los archivos de buzón de correo que habrán de revisarse. A fin de asignar valores a esos elementos, debe establecer un conjunto de valores a la variable **mail**. La matriz de nuevos valores se especifica con una lista de palabras separadas por espacios y encerradas entre paréntesis. El primer valor es un número indicando la cantidad de segundos que debe esperar antes de revisar el correo de nueva cuenta. Este valor se compara con el usado en la variable **MAILCHECK** de la shell BASH. Los valores restantes incluyen nombres de ruta de directorio, en que se encuentran los archivos de bandeja de entrada en que se revisará su correo. Observe que estos valores combinan las funciones de las variables **MAIL** y **MAILPATH**, de las shells BASH y Korn.

En el siguiente ejemplo, la variable **mail** se configura para revisar el correo cada 20 minutos (1200 segundos), y el archivo de buzón de correo se revisa en **usr/mail/chris**. El primer valor de la matriz asignada a **mail** es 1200, mientras el segundo valor en la matriz es el nombre de ruta del archivo de buzón de correo que habrá de revisarse.

```
> set mail ( 1200 /usr/mail/chris )
```

Con la misma facilidad, puede agregar más nombres de rutas de archivos de buzón de correo para el conjunto **mail**. En el siguiente ejemplo, se asignan dos buzones. Observe los espacios rodeando cada elemento.

```
> set mail ( 1200 /usr/mail/chris /home/mail/chris )
```

Archivos de inicialización de la shell TCSH: .login, .tcshrc, .logout

La shell TCSH tiene tres archivos de inicialización: **.login**, **.logout** y **.tcshrc**. Los archivos reciben su nombre de acuerdo con la operación que ejecutan. El archivo **.login** es para inicializar sesión, que se ejecuta cada vez que inicia sesión. El archivo **.logout** se ejecuta siempre que cierra la sesión. El archivo **.tcshrc** es de inicialización de shell, que se ejecuta en cada ocasión que entra a la shell TCSH, ya sea iniciando sesión o al cambiar a la shell TCSH, desde otra shell con el comando **tcsh**.

.login

La shell TCSH tiene su propio archivo de inicialización de inicio de sesión denominado **.login**, conteniendo los comandos de shell y definiciones de variables especiales utilizadas para configurar su shell. El archivo **.login** corresponde al archivo **.profile**, utilizado en las shells BASH y PDKSH.

Un archivo **.login** contiene comandos **setenv** que asignan valores a variables de entorno especiales, como **TERM**. Puede cambiar estos valores asignados al editar el archivo **.login** con cualquier editor estándar. También puede agregar nuevos valores. Sin embargo, recuerde que en la shell TCSH, el comando para asignar un valor a una variable de entorno es **setenv**. En el siguiente ejemplo, se define la variable **EXINIT** y se le asignan las opciones de numeración de línea y sangría automática del editor Vi.

```
> setenv EXINIT 'set nu ai'
```

Tenga cuidado cuando edite su archivo **.login**. Los cambios de edición inadvertidos pueden causar que las variables de configuren incorrectamente o no se configuren. Es recomendable hacer un respaldo de su archivo **.login** antes de editarlo.

Si ha hecho cambios a su archivo **.login** y quiere que los cambios surtan efecto durante su sesión, necesita ejecutar de nuevo el archivo. Puede hacerlo utilizando el comando **source**. Este comando ejecutará cualquier archivo de inicialización, incluidos **.tcshrc** y **.logout**. En el siguiente ejemplo, el usuario vuelve a ejecutar el archivo **.login**.

```
> source .login
```

Si también está planeando utilizar la shell PDKSH en su sistema Linux, necesita definir una variable denominada **ENV** en su archivo **.login** y asignarle un nombre para el archivo de inicialización de la shell PDKSH. Si después debe ingresar en la shell PDKSH desde TCSH, se ubica y ejecuta automáticamente el archivo de inicialización de la shell PDKSH. En el ejemplo del archivo **.login** mostrado a continuación, verá que el último comando asigna al archivo de inicialización de la shell PDKSH de **.kshrc** la variable **ENV**: **setenv ENV \$HOME/.kshrc**.

```
.login
setenv term vt100
setenv EXINIT 'set nu ai'

setenv ENV $HOME/.kshrc
```

.tcshrc

El archivo de inicialización **.tcshrc** se ejecuta cada vez que ingresa en la shell TCSH o genera cualquier subshell. Si TCSH es su shell de inicio de sesión, entonces el archivo **.tcshrc** se ejecutará con el archivo **.login** cuando inicia sesión. Si inserta la shell TCSH desde otra shell, el archivo **.tcshrc** se ejecuta automáticamente, y establecerán las definiciones de variable y alias contenidas.

En realidad, el archivo de inicialización de la shell **.tcshrc** se ejecuta cada vez que genera una shell, como cuando ejecuta una secuencia de comandos de shell. En otras palabras, cada vez que crea una subshell, se ejecuta el archivo **.tcshrc**. Esto permite definir variables locales en el archivo de inicialización **.tcshrc** y hacer que éstas, en un sentido, se exporten a cualquier subshell. Aunque las variables especiales definidas por el usuario como **history** son locales, se definirán para cada subshell generada. De esta forma, se configura **history** para cada subshell. Sin embargo, cada subshell tiene su propia variable **history** local. Incluso puede cambiar la variable **history** local en una subshell sin afectar alguna de las variables de otras subshell. La definición de variables

especiales en el archivo de inicialización de la shell permite tratarlas como variables exportadas de la shell BASH. Una variable en una shell BASH o PDKSH sólo pasa una copia de sí misma a cualquier subshell. El cambio de la copia no afecta la definición original.

El archivo **.tcshrc** también contiene la definición de alias y cualquier variable utilizada para activar las características de shell. Alias y variables de característica se definen de manera local en la shell. Pero el archivo **.tcshrc** los definirá en cada shell. Por esta razón, **.tcshrc** suele almacenar ese tipo de alias, como los definidos para los comandos **rm**, **cp** y **mv**. En el siguiente ejemplo se muestra un archivo **.tcshrc** con muchas de las definiciones estándar.

```
.tcshrc
set shell=/usr/bin/csh
set path= $PATH (/bin /usr/bin . )
set cdpath= ( /home /chris/informes /home /chris/letters )

set prompt="! $cwd >"
set history=20

set ignoreeof
set noclobber

alias rm 'rm -i'
alias mv 'mv -i'
alias cm 'cm -i'
```

Las variables locales, a diferencia de las variables de entorno, se definen con el comando **set**. Cualquier variable local que defina en **.tcshrc** utiliza el comando **set**. Es necesario colocar en el archivo **.login** cualquier variable definida con **setenv** como variable de entorno (por ejemplo, **TERM**). En el siguiente ejemplo se muestran los tipos de definiciones encontrados en el archivo **.tcshrc**. Observe que las variables **history** y **noclobber** se definen utilizando el comando **set**.

```
set history=20
set noclobber
```

Es posible editar cualquiera de los valores asignados a esas variables. Sin embargo, cuando edite los nombres de ruta asignados a **path** o **cdpath**, tenga en cuenta que estos nombres de ruta se incluyen en una matriz. Cada elemento de una matriz está separado por un espacio. Si agrega un nuevo nombre de ruta, debe cerciorarse que hay un espacio separándolo de los demás parámetros.

Si ha hecho cambios a **.tcshrc** y quiere se apliquen durante su sesión actual, recuerde ejecutar el archivo **.tcshrc** con el comando **source**:

```
> source .tcshrc
```

.logout

El archivo **.logout** también es de inicialización, pero se ejecuta cuando el usuario cierra la sesión. Está diseñado para realizar cualquiera de las operaciones deseadas cuando cierre su sesión. En vez de las definiciones de variable, el archivo **.logout** suele contener comandos de shell que dan forma un procedimiento de apagado. Por ejemplo, un comando de cierre de sesión común es revisar si existen trabajos activos en el fondo; otro es limpiar la imagen y después enviar un mensaje de despedida.

Como con **.login**, puede agregar sus propios comandos de shell al archivo **.logout**. Con el editor Vi puede cambiar el mensaje de despedida o agregar otras operaciones. En el siguiente ejemplo, el usuario tiene los comandos **clear** y **echo** en el archivo **.logout**. Cuando el usuario sale de su sesión, el comando **clear** limpiará la pantalla y **echo** desplegará el mensaje “Adiós, por ahora”.

```
.logout
```

```
clear
```

```
echo "Adiós, por ahora"
```

Archivos, directorios y archiveros de Linux

En Linux, todos los archivos se organizan en directorios que, a cambio, se conectan jerárquicamente entre sí en una estructura general de archivos. Se hace referencia a un archivo, no sólo de acuerdo con su nombre, sino su ubicación en esta estructura. Puede crear todos los directorios nuevos que quiera, agregando más a la estructura de archivos. Los comandos de archivo de Linux pueden realizar operaciones complejas, como mover o copiar directorios completos junto con sus subdirectorios. Puede usar operaciones de archivo como **find**, **cp**, **mv** y **ln** para localizar archivos, copiarlos, moverlos o vincularlos de un directorio a otro. Los administradores de archivo, como Konqueror y Nautilus, utilizados en los escritorios KDE y GNOME respectivamente, proporcionan una interfaz de usuario gráfica para realizar las mismas operaciones mediante iconos, ventanas y menús (consulte los capítulos 8 y 9). En este capítulo nos concentraremos en comandos para administración archivos en la línea shell, como **cp** y **mv**. Sin embargo, ya sea que utilice la línea de comandos o el administrador de archivos GUI, la estructura fundamental de archivos es la misma.

En el capítulo 32 se analiza a fondo la organización de la estructura de archivos de Linux, en diferentes directorios de sistema y administración de red. Aunque no es parte de la estructura de archivos, también existen herramientas especiales para acceder a particiones de Windows y discos flexibles. Éstos siguen casi el mismo formato que los comandos de archivos de Linux.

Los archiveros se usan para respaldar archivos o combinarlos en un paquete, que luego puede transferir como un solo archivo en Internet o publicarlo en un sitio FTP, para una descarga sencilla. La utilería de archivo estándar empleada en sistemas Linux y Unix es Tar, para el que existen varias GUI. Cuenta con varios programas de compresión de donde escoger, incluidos GNU zip (gzip), Zip, bzip y compress.

NOTA Linux también permite montar y acceder a sistemas de archivos usados por otros sistemas operativos, como Unix o Windows. Linux por sí sólo da soporte a varios sistemas diferentes de archivos: *ext2*, *ext3* y *ReiserFS*, entre ellos.

Archivos de Linux

Puede asignar un nombre a un archivo recurriendo a cualquier letra, subrayado y números. También puede incluir puntos y comas. Excepto casos especiales, nunca debe iniciar un nombre de archivo con un punto. Otros caracteres, como diagonales, signos de interrogación o asteriscos, se reservan para uso del sistema como caracteres especiales que no deben ser parte de un nombre de archivo. Los nombres de archivo pueden tener hasta 256 caracteres. También pueden incluir espacios, aunque para hacer referencia a dichos archivos en la línea de comandos, debe asegurarse de encerrarlos entre comillas. En un archivo de escritorio como GNOME o KDE, no necesita comillas.

Puede incluir una extensión como parte de un nombre de archivo. Un punto se utiliza para distinguir el nombre de archivo de la extensión. Las extensiones son útiles para ordenar sus archivos en categorías. Tal vez esté familiarizado con ciertas extensiones estándar adoptadas por convención. Por ejemplo, los archivos de código fuente C siempre tiene la extensión `.c`. Los archivos conteniendo código de objeto compilado, tienen una extensión `.o`. Por supuesto, puede crear extensiones propias de archivo. En los siguientes ejemplos se muestran nombres de archivos válidos para Linux. Tenga en cuenta que para hacer referencia a los últimos de estos nombres en la línea de comandos, deberá encerrarlos entre comillas como “Revisión de libros recientes”:

```
prefacio
capítulo2
9700info
Nuevas_Revisiones
calc.c
intro.bk1
Revisión de libros recientes
```

Los archivos especiales de inicialización también se utilizan para almacenar comandos de configuración de shell. Estos son archivos ocultos o de punto, que empiezan con un punto. Los archivos de punto utilizados por comandos y aplicaciones tienen nombres predeterminados, como el directorio `.mozilla`, utilizado para almacenar sus datos y archivos de configuración de Mozilla. Recuerde que cuando utiliza `ls` para desplegar sus nombres de archivo, no se desplegarán los archivos de punto. Para incluir los archivos de punto, necesita utilizar `ls` con la opción `-a`. Los archivos de punto se discuten con más detalle en el capítulo 5.

El comando `ls -l` despliega información detallada acerca de un archivo. Primero se despliegan permisos, seguidos por un número de vínculos, propietario del archivo, nombre del grupo al que pertenece el usuario, tamaño del archivo en bytes, fecha y hora de la última modificación del archivo y el nombre de éste. Los permisos indican quién puede acceder al archivo: usuario, miembros del grupo o todos los usuarios. Los permisos se analizan con más detalle en páginas posteriores de este capítulo. El nombre del grupo indica que grupo tiene permiso para acceder al objeto de archivo. En el ejemplo del siguiente párrafo, el tipo de archivo `misdatos` es común. Sólo existe un vínculo, indicando que el archivo no tiene otros nombres ni vínculos. El nombre del usuario es `chris`, el mismo de inicio de sesión y del grupo `clima`. Es probable que también otros usuarios pertenezcan al grupo `clima`. El tamaño del archivo es de 207 bytes, modificado por última vez el 20 de febrero a las 11:55 AM. El nombre del archivo es `misdatos`.

Si quiere desplegar esta información detallada para todos los archivos en el directorio, sólo utilice el comando `ls -l` sin un argumento.

```
$ ls -l
-rw-r--r-- 1 chris clima 207 Feb 20 11:55 misdatos
-rw-r--r-- 1 chris clima 568 Feb 14 10:30 misdatos
-rw-r--r-- 1 chris clima 308 Feb 17 12:40 misdatos
```

Todos los archivos de Linux tienen un formato físico —un flujo de bytes. Un *flujo de bytes* no es más que una secuencia de bytes. Esto permite a Linux aplicar el concepto de archivo a cada componente de datos del sistema. Los directorios se clasifican como archivos, al igual que dispositivos. El hecho de tratar todo como un archivo, permite a Linux organizar y cambiar datos de manera más sencilla. Los datos en un archivo pueden enviarse directamente a un dispositivo como la pantalla, porque un dispositivo interactúa con el sistema usando el mismo formato de archivos de flujo de bytes que los archivos regulares.

Este mismo formato de archivo se usa para implementar otros componentes de sistema operativo. La interfaz con un dispositivo, como la pantalla o teclado, se designa con características de archivo. Otros componentes, como directorios, son por sí solos archivos de flujo de byte, pero tienen una organización interna. Un archivo de directorio contiene información acerca de un directorio, organizado en un formato de directorio especial. Debido a que estos componentes diferentes se tratan como archivos, se puede decir constituyen diferentes *tipos de archivo*. Un dispositivo de carácter es un tipo de archivo. Un directorio es otro tipo de archivo. El número de estos tipos de archivo puede variar dependiendo de su implementación de Linux. Sin embargo, existen cinco tipos comunes de archivo: archivos ordinarios, directorio, canalizaciones primero-en-entrar primero-en-salir, dispositivo de carácter y de bloqueo de dispositivo. A pesar de que tal vez sea muy raro hacer referencia a un tipo de archivo, resulta útil cuando busca directorios o dispositivos. En páginas posteriores de este capítulo, verá cómo utilizar el tipo de archivo en un criterio de búsqueda con el comando **find**, para encontrar específicamente directorios o nombres de dispositivos.

A pesar de que todos los archivos ordinarios tienen un formato de flujo de bytes, es probable se utilicen de diferentes maneras. La diferencia más significativa es entre archivos binarios y de texto. Los programas compilados son ejemplos de archivos binarios. Sin embargo, aun los archivos de texto pueden clasificarse de acuerdo con sus diferentes usuarios. Puede tener archivos conteniendo código fuente de programación C o comandos de shell, incluso un archivo vacío. El archivo puede ser un programa ejecutable o de directorio. El comando **file** de Linux le ayuda a determinar para qué se utiliza un archivo. Éste examina las primeras líneas de un archivo e intenta determinar una clasificación para éste. El comando **file** busca palabras clave o números especiales en las primeras líneas, pero no es siempre preciso. En el siguiente ejemplo, el archivo **file** examina el contenido de dos archivos y determina una clasificación para éstos:

```
$ file lunes reportes
lunes: text
reportes: directory
```

Si necesita examinar todo el archivo, byte por byte, puede hacerlo con el comando **od** (octal dump, volcado octal). El comando **od** realiza un volcado de un archivo. Como opción predeterminada, imprime cada byte en su representación octal. Sin embargo, también puede especificar una representación de carácter, decimal o hexadecimal. El comando **od** es útil cuando necesita detectar cualquier carácter especial en su archivo o quiere desplegar un archivo binario.

La estructura de archivos

Linux organiza archivos en un conjunto de directorios conectados jerárquicamente. Cada directorio puede contener archivos o directorios. En este sentido, los directorios realizan dos funciones importantes. Un *directorio* almacena archivos, de manera muy similar a los archivos guardados en un archivero y el directorio se conecta con otros directorios, de la misma forma en que una rama en árbol está conectada a otras ramas. Debido a semejanza con un árbol, a dichas estructuras suele conocerseles como *estructura de árbol*.

La estructura de archivos de Linux se ramifica en varios directorios, empezando por el raíz, /. En éste, varios directorios de sistema contienen archivos y programas característicos de Linux. El directorio raíz también contiene un directorio denominado **/home**, alojando el directorio de inicio y todos los usuarios del sistema. Cada directorio de inicio de usuario, a cambio, contiene los directorios creados por el usuario para uso propio. A su vez, cada uno de estos también contiene directorios. Tales directorios anidados se ramifican desde el directorio de inicio del usuario.

NOTA El directorio de inicio del usuario puede ser cualquiera, aunque suele ser el nombre de inicio de sesión del usuario. Este directorio se localiza en el directorio denominado **/home** de su sistema Linux. Por ejemplo, un usuario llamado **dylan** tendrá un directorio de inicio denominado **dylan**, en el directorio de sistema **/home**. El directorio de inicio del usuario es un subdirectorio del directorio denominado **/home** en su sistema.

Directorios de inicio

Cuando inicia sesión en su sistema, se le coloca dentro de su directorio de inicio. El nombre asignado por el sistema a ese directorio es el mismo que de inicio de sesión. Cualquier archivo creado cuando inicia sesión por primera vez se organiza dentro de su directorio de inicio. Sin embargo, dentro este puede crear más directorios. Después puede cambiar a esos directorios y almacenar archivos en éstos. Lo mismo es cierto para otros usuarios de su sistema. Cada usuario tiene su propio directorio de inicio, identificado por el nombre de inicio de sesión apropiado. Los usuarios, a su vez, pueden crear sus propios directorios.

Se accede a un directorio ya sea por nombre o convirtiéndolo en directorio de trabajo. A cada directorio se le asigna un nombre cuando se crea. Puede utilizar este nombre en las operaciones para acceder a archivos de ese directorio. También puede hacer que el directorio sea de trabajo. Si no utiliza nombre de directorio en una operación de archivo, se accederá al directorio de trabajo. Éste es el directorio en que está trabajando actualmente. Cuando inicia sesión, el directorio de trabajo es su directorio de inicio, que suele tener el mismo nombre del inicio de sesión. Puede cambiar el directorio de trabajo con el comando **cd** para designar otro directorio.

Nombres de ruta

El nombre dado a un directorio o archivo cuando lo crea, no es su nombre completo. El nombre completo de un directorio es su *nombre de ruta*. Las relaciones jerárquicas anidadas dentro de los directorios forman rutas, y estas pueden utilizarse para identificar y hacer referencia a cualquier directorio o archivo único o absoluto. Puede afirmarse que cada directorio de la estructura de archivos tiene ruta propia única. El nombre actual por el que el sistema identifica un directorio siempre comienza con el directorio root e incluye todos los directorios anidados bajo ese directorio.

En Linux, usted escribe un nombre de ruta al mostrar una lista de cada directorio en la ruta, separando cada directorio con una diagonal. Una diagonal antes del primer directorio en la ruta representa el directorio raíz. El nombre de ruta del directorio **robert** es **/home/robert**. El nombre de ruta del directorio **reportes** es **/home/chris/reportes**. Los nombres de ruta también aplican para archivos. Cuando crea un archivo dentro de un directorio, se le asigna un nombre al archivo. Sin embargo, el nombre real por el cual el sistema identifica al archivo, es el nombre del archivo combinado con la ruta de los directorios, desde el directorio raíz hasta el directorio del archivo. Como ejemplo, el nombre de ruta de **lunes** es **/home/chris/reportes/lunes** —el directorio raíz está representado por la primera diagonal. La ruta del archivo **lunes** incluye directorios raíz, **home**, **chris** y **reportes**, así como nombre de archivo **lunes**.

Los nombres de ruta pueden ser absolutos o relativos. Un *nombre de ruta absoluto* es el nombre de ruta completo de un archivo o directorio comenzando con el directorio raíz. Un *nombre de ruta relativo* empieza desde su directorio de trabajo; es la ruta de un archivo en relación con su directorio de trabajo. El de trabajo es el directorio en que está operando. En el ejemplo anterior, si **chris** es su directorio de trabajo, el nombre de ruta relativo del archivo **lunes** es **reportes/lunes**. El nombre de ruta absoluto de **lunes** es **/home/chris/reportes/lunes**.

El nombre de ruta absoluto, desde el directorio raíz hasta el de inicio, puede ser muy complejo y, algunas veces, incluso está sujeto a cambios por parte del administrador del sistema. Para que sea más fácil hacer referencia a él, puede utilizar un carácter especial, la tilde (~), que representa el nombre de ruta absoluto para su directorio de inicio. En el siguiente ejemplo, desde el directorio **gracias**, el usuario hace referencia al archivo **clima** en el directorio de inicio, colocando una tilde y una diagonal antes de **clima**:

```
$ pwd
/home/chris/cartas/gracias
$ cat ~/clima
lluvioso y cálido
$
```

Debe especificar el resto del archivo desde su directorio de inicio. En el siguiente ejemplo, el usuario hace referencia al archivo **lunes** en el directorio **reportes**. La tilde representa la ruta al directorio de inicio del usuario, **/home/chris**, y después se especifica el resto de la ruta al archivo **lunes**.

```
$ cat ~/reportes/lunes
```

Directorios de sistema

El directorio raíz que comienza la estructura de archivos de Linux contiene varios directorios de sistema. Estos contienen archivos y programas empleados para ejecutar y mantener el sistema. Muchos contienen subdirectorios con programas para ejecutar características específicas de Linux. Por ejemplo, el directorio **/usr/bin** contiene los diversos comandos de Linux ejecutados por los usuarios, como **lp1**. El directorio **/bin** almacena comandos al nivel del sistema. En la tabla 6-1 se muestra una lista de directorios básicos del sistema.

Listas, despliegue e impresión de archivos: ls, cat, more, less y lpr

Una de las funciones primarias de un sistema operativo es la administración de archivos. Tal vez necesite realizar ciertas operaciones de salida básicas en sus archivos, como desplegarlas en su pantalla o imprimirlas. El sistema Linux proporciona un conjunto de comandos que realizan operaciones básicas de administración de archivo, como mostrar listas, desplegar e imprimir archivos, además de copiar, cambiar nombre y borrar archivos. Los nombres de comando suelen formarse a partir de versiones abreviadas de palabras. Por ejemplo, el comando **ls** es una forma corta de “list” (lista) y muestra una lista de archivos en su directorio. El comando **lpr** es una abreviación de “line print” (imprimir línea) e imprimirá un archivo. Los comandos **cat**, **less** y **more** despliegan el contenido de un archivo en pantalla. En la tabla 6-2 se muestra una lista de estos comandos con sus diferentes opciones. Cuando inicia sesión en su sistema Linux, es posible quiera una lista de archivos en su directorio de inicio. El comando **ls**, que da salida a una lista de archivos y nombres de directorio, es útil para esto. **ls** tiene muchas opciones posibles para desplegar nombres de archivo, de acuerdo con sus características especiales.

Directorio	Función
/	Comienza la estructura de archivos del sistema, denominada raíz.
/home	Contiene los directorios de inicio de los usuarios.
/bin	Almacena todos los comandos estándar y programas de utilerías.
/usr	Almacena archivos y comandos utilizados por el sistema; este directorio se desglosa en varios subdirectorios.
/usr/bin	Almacena comandos orientados al usuario y programas de utilería.
/usr/sbin	Almacena comandos de administración del sistema.
/usr/lib	Almacena bibliotecas para lenguajes de programación.
/usr/share/doc	Almacena documentación de Linux.
/usr/share/man	Almacena archivos en línea Man.
/var/spool	Almacena archivos en cola, como los generados para trabajos de impresión y transferencias de red.
/sbin	Almacena comandos administrativos para arranque del sistema.
/var	Almacena archivos variables, como los de buzón de correo.
/dev	Almacena interfaces de archivo para dispositivos como las terminales e impresoras (generado dinámicamente por udev, no lo edite).
/etc	Almacena archivos de configuración del sistema y cualquier otro archivo de sistema.

TABLA 6-1 Directorios de sistema estándar en Linux

Despliegue de archivos: **cat**, **less** y **more**

Tal vez también necesite ver el contenido de un archivo. Los comandos **cat** y **more** despliegan el contenido de un archivo en pantalla. El nombre **cat** viene de *concatenate* (unir).

```
$ cat misdatos
computadoras
```

El comando **cat** da salida de todo el texto de un archivo a la pantalla. Esto presenta un problema cuando el archivo es grande, porque el texto pasa rápidamente por la pantalla. Los comandos **more** y **less** están diseñados para evitar esta limitación y desplegar una pantalla de texto a la vez. Después, puede recorrer el texto hacia adelante o atrás en el texto, a su gusto. Puede invocar los comandos **more** o **less** introduciendo el nombre de comando seguido por el nombre del archivo que quiere ver (**less** es una utilería de despliegue poderosa y configurable).

```
$ less misdatos
```

Cuando los comandos **more** o **less** invocan un archivo, se despliega la primera pantalla del texto. Para pasar a la siguiente pantalla, oprima la tecla **F** o BARRA ESPACIADORA. Para moverse hacia atrás en el texto, oprima la tecla **B**. Puede salir cuando quiera al oprimir la tecla **q**.

Comando	Ejecución
<code>ls</code>	Muestra una lista de los nombres de archivo y directorio.
<code>cat nombresdearchivo</code>	Despliega un archivo. Puede tomar nombres de archivo para sus argumentos. Presenta el contenido de esos archivos directamente en la salida estándar, la que, como opción predeterminada, se dirige a la pantalla.
<code>more nombresdearchivo</code>	Despliega un archivo, pantalla por pantalla. Oprima la tecla <code>BARRA ESPACIADORA</code> para pasar a la siguiente pantalla y <code>q</code> para salir.
<code>less nombresdearchivo</code>	Despliega un archivo, pantalla por pantalla. Oprima la tecla <code>BARRA ESPACIADORA</code> para pasar a la siguiente pantalla y <code>q</code> para salir.
<code>lpr nombresdearchivo</code>	Envía un archivo a la impresora en línea para su impresión; puede utilizarse una lista de archivos como argumento. Utilice la opción <code>-p</code> para especificar una impresora.
<code>lpq</code>	Muestra una lista de la cola de impresiones para los trabajos de impresión.
<code>lprm</code>	Quita un trabajo de impresión de la cola de impresión.

TABLA 6-2 Listas, despliegue e impresión de archivos

Archivos de impresión: lpr, lpq y lprm

Con comandos de impresora como `lpr` y `lprm`, puede realizar operaciones de impresión, como imprimir archivos y cancelar trabajos de impresión (véase la tabla 6-2). Cuando necesite imprimir archivos, utilice el comando `lpr` para enviar archivos a la impresora conectada a su sistema. En el siguiente ejemplo, el usuario imprime el archivo `misdatos`:

```
$ lpr misdatos
```

Si quiere imprimir varios archivos a la vez, puede especificar más de un archivo en la línea de comandos después del comando `lpr`. En el siguiente ejemplo, el usuario imprime los archivos `misdatos` y `prefacio`:

```
$ lpr misdatos prefacio
```

Los trabajos de impresión se colocan en una cola e imprimen de uno en uno, en segundo plano. Puede continuar con otro trabajo mientras sus archivos se imprimen. Verá la posición de un trabajo de impresión en particular en cualquier momento con el comando `lpq`, indicando el propietario del trabajo de impresión (el nombre de inicio de sesión de un usuario que envía el trabajo), el ID del trabajo de impresión, tamaño en bytes y archivo temporal en que está actualmente almacenado.

Si necesita cancelar un trabajo de impresión no deseado, puede hacerlo con el comando `lprm`, que toma su argumento del ID del trabajo de impresión o nombre del dueño. Después quita el trabajo de impresión de la cola de impresión. Para esta tarea, `lpq` es útil, porque proporciona ID y nombre del dueño del trabajo de impresión que necesita utilizar con `lprm`.

Administración de directorios: mkdir, rmdir, ls, cd y pwd

Puede crear o eliminar sus propios directorios, además de cambiar su directorio de trabajo, con los comandos `mkdir`, `rmdir` y `cd`. Cada uno de estos comandos toma como argumento el nombre de

ruta de un directorio. El comando `pwd` despliega un nombre de ruta absoluto para su directorio de trabajo. Además de estos comandos, se pueden usar los caracteres especiales representados por un punto, dos puntos seguidos y una tilde pueden usarse para referir el directorio de trabajo, el padre del directorio de trabajo y el directorio de inicio, respectivamente. Juntos, estos comandos permiten administrar sus directorios. Puede crear directorios anidados, mover un directorio a otro y utilizar nombres de ruta para hacer referencia a cualquiera de sus directorios. Los comandos más comunes utilizados para administrar directorios se muestran en la lista de la tabla 6-3.

Creación y eliminación de directorios

Para crear o eliminar directorios se usan los comandos `mkdir` y `rmdir`. En ambos casos, también puede usar los nombres de ruta de los directorios. En el siguiente ejemplo, el usuario crea el directorio **reportes**. Después, el directorio **cartas**, utilizando un nombre de ruta:

```
$ mkdir reportes
$ mkdir /home/chris/cartas
```

Comando	Ejecución
<code>mkdir directorio</code>	Crea un directorio.
<code>rmdir directorio</code>	Borra un directorio.
<code>ls -F</code>	Muestra una lista del nombre del directorio precedido con una diagonal.
<code>ls -R</code>	Muestra una lista del directorio de trabajo, además de todos los subdirectorios.
<code>cd nombrededirectorio</code>	Cambia al directorio específico, haciéndolo directorio de trabajo. <code>cd</code> sin nombre de directorio lo cambia de nuevo al directorio de inicio: <code>\$ cd reports</code>
<code>pwd</code>	Despliega el nombre de ruta del directorio de trabajo.
<code>nombrededirectorio/ nombredearchivo</code>	Se usa una diagonal en los nombres de ruta para separar cada nombre de directorio. En el caso de nombres de ruta de archivos, una diagonal separa los nombres del directorio anterior, desde el nombre de archivo.
<code>..</code>	Hace referencia al directorio padre. Puede utilizarlo como argumento o parte de un nombre de ruta: <code>\$ cd ..</code> <code>\$ mv ../larisa cartasanteriores</code>
<code>.</code>	Hace referencia al directorio de trabajo. Puede utilizarlo como un argumento o parte de un nombre de:
<code>~/directorio</code>	La tilde es un carácter especial representando el nombre de ruta para el directorio de inicio. Es útil cuando necesita emplear un nombre de ruta absoluto para un archivo o directorio: <code>\$ cp monday ~/today</code>

TABLA 6-3 Comandos de directorio

Puede eliminar un directorio con el comando **rmdir**, seguido por el nombre del directorio. En el siguiente ejemplo, el usuario elimina el directorio **reportes** con el comando **rmdir**:

```
$ rmdir reportes
```

Para eliminar un directorio y todos sus subdirectorios, necesita utilizar el comando **rm** con la opción **-r**. Se trata de un comando poderoso que puede utilizarse de manera sencilla para eliminar todos sus archivos. Si su comando **rm** tiene un alias de **rm -i** (modo interactivo), se le pedirá confirme la eliminación de cada archivo. Para quitar todos los archivos y subdirectorios sin confirmación, agregue la opción **-f**. En el siguiente ejemplo se elimina el directorio **reportes** y todos sus subdirectorios.

```
rm -rf reportes
```

Despliegue del contenido de directorios

Ya ha visto cómo utilizar el comando **ls** para mostrar una lista de los archivos y directorios de su directorio de trabajo. Sin embargo, para distinguir entre el nombre de un archivo y un directorio, necesita utilizar el comando **ls** con la opción **-F**. Luego se coloca una diagonal tras cada nombre de directorio en la lista.

```
$ ls
clima reportes cartas
$ ls -F
clima reportes/ cartas/
```

El comando **ls** también toma como argumento cualquier nombre de directorio o ruta de directorio. Esto permite mostrar una lista de archivos en cualquier directorio, sin cambiar a ese directorio. En el siguiente ejemplo, el comando **ls** toma como argumento el nombre de un directorio, **reportes**. Después el comando **ls** se ejecuta nuevamente, sólo que esta vez se usa el nombre de ruta absoluto de **reportes**.

```
$ ls reportes
lunes martes
$ ls /home/chris/reportes
lunes martes
$
```

Desplazamiento entre directorios

El comando **cd** toma como argumento el nombre de un directorio al que quiere cambiarse. El nombre del directorio puede ser un subdirectorio en su directorio de trabajo o la ruta completa de cualquier directorio del sistema. Si quiere regresar a su directorio de inicio, sólo necesita insertar el comando **cd** por sí solo, sin argumento de nombre de archivo.

```
$ cd props
$ pwd
/home/dylan/props
```

Referencia al directorio padre

Un directorio siempre tiene padre (excepto, por supuesto, el raíz). Por ejemplo, en la lista anterior, el padre de **props** es **dylan**. Cuando se crea un directorio, se incluyen dos entradas: una presentada con un punto (.) y otra con dos puntos seguidos (.). El punto representa el nombre de ruta del directorio y los dos puntos seguidos representan el nombre de ruta de su directorio padre. Los dos puntos seguidos, utilizados como argumento en un comando, hacen referencia al directorio padre. Un punto hace referencia al propio directorio.

Puede utilizar solo un punto para referirse a su directorio de trabajo, en vez de usar un nombre de ruta. Por ejemplo, para copiar un archivo al directorio de trabajo, con el mismo nombre, puede usar el punto en lugar del nombre de ruta del directorio de trabajo. En este sentido, el punto es otro nombre para el directorio de trabajo. En el siguiente ejemplo, el usuario copia el archivo **clima**, del directorio **chris**, a **reportes**. El directorio **reportes** es el de trabajo y puede representarse con un punto.

```
$ cd reportes
$ cp /home/chris/clima .
```

El símbolo **..** suele utilizarse para hacer referencia a archivos en un directorio padre. En el siguiente ejemplo, el comando **cat** despliega el archivo **clima** en el directorio padre. El nombre de ruta del archivo es el símbolo **..** seguido por una diagonal y el nombre de archivo.

```
$ cat ../clima
lluvioso y cálido
```

***SUGERENCIA** Puede utilizar el comando **cd** con el símbolo **..** para ir hacia atrás a través de directorios padre sucesivos en el árbol, desde un directorio inferior.*

Operaciones de archivo y directorio: find, cp, mv, rm e ln

A medida que crea más y más archivos, tal vez quiera respaldarlos, cambiar sus nombres, borrar algunos e incluso darle nombres agregados. Linux proporciona varios comandos de archivo que permiten buscar y copiar, cambiar de nombre y eliminar archivos (véase la tabla 6-5, en páginas posteriores de este capítulo). Si tiene gran número de archivos, también puede buscarlos para localizar uno específico. El comando nombra formas cortas de palabras completas, constando de dos caracteres. El comando **cp** viene de “copy” y copia un archivo, **mv** viene de “move”, cambia de nombre y mueve archivos, **rm** viene de “remove” (eliminar) y elimina un archivo, asimismo **ln** viene de “link” (vincular) y agrega otro nombre a un archivo, por lo que a menudo se usa como acceso rápido al original. Una excepción a la regla de dos caracteres es el comando **find**, para realizar búsquedas de nombres de archivo. Todas estas operaciones pueden llevarse a cabo con escritorios GUI como GNOME y KDE (consulte los capítulos 7 y 8).

Búsqueda de directorios: find

Apenas tenga gran cantidad de archivos en muchos directorios, tal vez necesite buscar entre ellos para localizar uno o varios archivos específicos, de cierto tipo. El comando **find** le permite realizar dicha búsqueda desde la línea de comandos. El comando **find** toma como argumento los nombres de directorio, seguidos por varias opciones posibles, especificando tipo de búsqueda y criterio de ésta; luego busca dentro de directorios y subdirectorios de la lista, aquellos archivos satisfaciendo

dichos criterios. El comando `find` puede buscar cualquier nombre, tipo, propietario e incluso hora de la última actualización.

```
$ find lista-directorios -opción criterios
```

SUGERENCIA Desde el escritorio GNOME puede utilizar la herramienta *Buscar*, en el menú *Lugares*, para rastrear archivos. Desde el escritorio KDE, la herramienta de búsqueda del administrador de archivos. Seleccione *Buscar* desde el menú de herramientas del administrador de archivos (Konqueror).

La opción **-name** tiene como criterio un patrón e instruye a **find** para buscar el nombre de archivo coincidente con ese patrón. Para buscar un archivo por nombre, puede emplear el comando **find** con el nombre de directorio seguido por la opción **-name** y el nombre del archivo.

```
$ find lista-directorios -name nombredelarchivo
```

El comando **find** también tiene opciones que simplemente realizan acciones, como dar salida a resultados de una búsqueda. Si quiere que **find** despliegue los nombres de archivo encontrados, sólo debe incluir la opción **-print**, en la línea de comandos junto con cualquier otra. La opción **-print** es una acción instruyendo a **find** para escribir en la salida estándar nombres de todos los archivos localizados (también puede utilizar la opción **-ls**, en lugar de mostrar una lista de archivos en formato largo). En el siguiente ejemplo, el usuario busca todos los archivos en el directorio **reportes** con el nombre **lunes**. Una vez localizado, se imprime el archivo, con su nombre de ruta relativo.

```
$ find reportes -name lunes -print
reportes/lunes
```

El comando **find** imprime nombres de archivo recurriendo al nombre de directorio especificado en la lista del directorio. Si indica un nombre de ruta absoluto, se enviará la salida a la ruta absoluta de directorios encontrados. En el ejemplo anterior, el usuario especificó un nombre de ruta relativo, **reportes**, en la lista de directorios. Los archivos localizados se enviaron a la salida, comenzando con su nombre de ruta relativo. En el siguiente ejemplo, el usuario especifica un nombre de ruta absoluto en la lista de directorios. Los nombres de archivo localizados después se envían a la salida, mediante su nombre de ruta absoluto.

```
$ find /home/chris -name lunes -print
/home/chris/reportes/lunes
```

SUGERENCIA Si necesita encontrar la ubicación de un programa o archivo de configuración específico, puede utilizar **find** para buscar el archivo desde el directorio raíz. Inicie sesión como usuario **root** y utilice **/** como directorio. Este comando busca la ubicación del comando **more** y los archivos en todo el sistema: **find / -name more -print**.

Búsqueda del directorio de trabajo

Si quiere buscar su directorio de trabajo, emplee el punto en el nombre de ruta del directorio, para representar su directorio de trabajo. Los dos puntos seguidos indican el directorio padre. En el siguiente ejemplo se buscan todos los archivos y subdirectorios del directorio de trabajo, utilizando el punto para representar el directorio de trabajo. Si se encuentra en el directorio de trabajo, es una

forma conveniente de buscar en todos sus directorios. Observe que los nombres de archivo localizados se envían a la salida con un punto al principio.

```
$ find . -name clima -print
./clima
```

Puede utilizar caracteres comodín de shell, como elemento de su criterio de patrón para búsqueda de archivos. Sin embargo, el carácter especial debe estar entre comillas, para evitar que la shell lo evalúe. En el siguiente ejemplo, se buscan todos los archivos con extensión **.c** en el directorio **programas** y después se despliegan en el formato largo mediante la acción **-ls**:

```
$ find programas -name '*.c' -ls
```

Localización de directorios

Puede utilizar el comando **find** para localizar otros directorios. En Linux, un directorio está clasificado oficialmente como un tipo especial de archivo. Aunque todos los archivos tienen el mismo formato de flujo de bytes, algunos, como los directorios, se utilizan de manera especial. En este sentido, se puede decir que un archivo tiene un tipo de archivo. El comando **find** tiene la opción denominada **-type**, para buscar un archivo de tipo determinado. La opción **-type** toma un modificador de un solo carácter representando el tipo de archivo. El modificador indicando un directorio es una **d**. En el siguiente ejemplo, el nombre de directorio y tipo de archivo de directorio, se manejan para buscar el directorio llamado **gracias**:

```
$ find /home/chris -name gracias -type d -print
/home/chris/cartas/gracias
$
```

Los tipos de archivo no difieren tanto como los formatos de archivo aplicados a otros componentes del sistema operativo, en este caso, los dispositivos. En este sentido, un dispositivo se trata como tipo de archivo, y puede utilizar **find** para buscar dispositivos y directorios, además de archivos comunes. En la tabla 6-4 se muestra una lista de diferentes tipos disponibles para la opción **-type**, del comando **find**.

También puede usar la operación de búsqueda para localizar archivos por criterio de propiedad o seguridad, como los pertenecientes a un usuario específico o que tienen cierto tipo de contexto de seguridad. La opción **user** permite localizar todos los archivos relativos a cierto usuario. En el siguiente ejemplo se muestra una lista de todos los archivos creados por el usuario **chris** o de los que es dueño en todo el sistema. Para mostrar una lista de éstos, sólo en el directorio de inicio del usuario, utilice **/home** como directorio de búsqueda inicial. Esto encuentra todos los archivos del directorio de inicio del usuario, además de cualquier archivo propiedad de ese usuario en otros directorios.

```
$ find / -user chris -print
```

Copia de archivos

Para hacer una copia de archivo, sólo necesita dar a **cp** dos nombres de archivo como argumento (véase la tabla 6-5). El primer nombre es del archivo a ser copiado (el que ya existe). A éste a menudo se conoce como *archivo fuente*. El segundo nombre de archivo es el que quiere para la copia. Esto será un nuevo archivo conteniendo una copia de todos los datos en el fuente. Al segundo argumento a menudo se le conoce como *archivo destino*. La sintaxis para el comando **cp** sigue:

```
$ cp archivo-fuente archivo-destino
```

Comando u opción	Ejecución
find	Busca archivos en los directorios, de acuerdo con los criterios de búsqueda. Este comando tiene varias opciones especificando criterios de búsqueda y acciones a ser tomadas.
-name <i>patrón</i>	Busca archivos con <i>patrón</i> en el nombre.
-lname <i>patrón</i>	Busca archivos de vinculación simbólica.
-grupo <i>nombre</i>	Busca archivo pertenecientes al <i>name</i> de grupo.
-gid <i>nombre</i>	Busca archivos pertenecientes al grupo, de acuerdo con el ID de grupo.
-user <i>nombre</i>	Busca archivos propiedad de un usuario.
-uid <i>nombre</i>	Busca archivos propiedad de un usuario, de acuerdo con el ID de usuario.
-size <i>numc</i>	Busca archivos con el <i>num</i> de tamaño en bloques. Si se agrega c tras <i>num</i> , se busca el tamaño en bytes (caracteres).
-mtime <i>num</i>	Busca archivos modificados hace un <i>num</i> de días.
-newer <i>patrón</i>	Busca archivos modificados tras una coincidencia con el <i>patrón</i> .
-context <i>contexto de seguridad</i>	Busca archivos de acuerdo con el contexto de seguridad (SE Linux).
-print	Envía la salida de los resultados de la búsqueda a la salida estándar. El resultado suele ser una lista de nombres de archivo, incluyendo sus nombres de ruta completos.
-type <i>tipodearchivo</i>	Busca archivos según un archivo específico. El tipo de archivo puede ser b para dispositivo de bloqueo, c para dispositivo de carácter, d para directorio, f para archivo o l para un vínculo simbólico.
-perm <i>permiso</i>	Busca archivos con cierta configuración de permisos. Utiliza formato octal o simbólico para permisos.
-ls	Presenta una lista detallada de cada archivo, con información de propietario, permiso, tamaño y fecha.
-exec <i>comando</i>	Ejecuta un comando cuando se encuentran los archivos.

TABLA 6-4 El comando **find**

En el siguiente ejemplo, el usuario copia un archivo llamado **propuesta** a un nuevo archivo llamado **propanterior**:

```
$ cp propuesta propanterior
```

Sin intención, podría destruir otro archivo con el comando **cp**. Este comando genera una copia al crear primero un archivo y después copiar los datos en él. Si otro archivo tiene el mismo nombre que el de destino, ese archivo se destruirá y creará uno nuevo con ese nombre. Como opción predeterminada, Red Hat configura su sistema para revisar una copia existente con el mismo nombre (**cp** tiene un alias junto con la opción **-i**, (consulte el capítulo 5). Para copiar un archivo desde su directorio de trabajo a otro directorio, sólo debe utilizar ese nombre de directorio como

Comando	Ejecución
<code>cp nombreadarchivo nombreadarchivo</code>	Copia un archivo. cp toma dos argumentos: archivo original y nombre de la nueva copia. Puede utilizar nombres de ruta para copiar entre directorios: \$ cp hoy reportes/Lunes
<code>cp -r nombrededirectorio nombrededirectorio</code>	Copia un subdirectorio de un directorio a otro. El directorio copiado incluye todos sus subdirectorios: \$ cp -r cartas/gracias cartasanteriores
<code>mv nombreadarchivo nombreadarchivo</code>	Mueve (cambia el nombre de) un archivo. El comando mv toma dos argumentos: el primero es el archivo al que se moverá. El segundo argumento puede ser el nombre del archivo nuevo o de ruta de un directorio. Si es el nombre de un directorio, entonces el archivo se mueve literalmente a ese directorio, al cambiar el nombre de ruta del archivo: \$ mv hoy /home/chris/reportes
<code>mv dirname dirname</code>	Mueve directorios. En este caso, primero y último argumento son los directorios: \$ mv cartas/gracias cartasanteriores
<code>ln nombreadarchivo nombreadarchivo</code>	Crea nombres agregados para archivos a los que se refiere como vínculos. Un vínculo puede crearse en un directorio que hace referencia a un archivo en otro directorio: \$ ln hoy reportes/Lunes
<code>rm nombreadarchivos</code>	Elimina (borra) un archivo. Puede tomar cualquier número de archivos como argumento. Quita los vínculos a un archivo. Si un archivo tiene más de un vínculo, necesita eliminar todos para eliminar un archivo: \$rm hoy clima findesemana

TABLA 6-5 Operaciones de archivo

segundo argumento en el comando **cp**. En el siguiente ejemplo, el archivo **propuesta** se sobrescribe con el archivo **nuevaprop**. El archivo **propuesta** ya existe.

```
$ cp nuevaprop propuesta
```

Puede utilizar cualquiera de los caracteres comodín para generar una lista de nombres de archivo, para ser usados con **cp** o **mv**. Por ejemplo, supongo que necesita copiar todos sus archivos de código fuente C a un directorio dado. En vez de hacer una lista de cada uno individualmente, en la línea de comandos, puede utilizar un carácter ***** con la extensión **.c**, para relacionar y generar una lista de los archivos de código fuente C (todos los archivos con una extensión **.c**). En el siguiente ejemplo, el usuario copia todos los archivos de código fuente del directorio actual al directorio **respfuente**:

```
$ cp *.c respfuente
```

Si quiere copiar todos los archivos de un directorio determinado a otro, puede utilizar ***** para generar una lista de esos archivos en un comando **cp**. En el siguiente ejemplo, el usuario copia todos los archivos del directorio **props** a **propanterior**. Observe el uso del nombre de ruta de **props** con el carácter especial *****.

```
$ cp props/* propanterior
```


Por supuesto, puede utilizar cualquiera de los caracteres especiales, como `.`, `?` o `[]`. En el siguiente ejemplo, el usuario copia los archivos de código fuente y código de objeto (`.c` y `.o`) al directorio **resproy**:

```
$ cp *. [oc] resproy
```

Cuando copia un archivo, tal vez quiera dar a la copia un nombre diferente al original. Para ello, coloque el nuevo nombre de archivo tras el nombre de directorio, separado por una diagonal.

```
$ cp nombredearchivo nombre-directorio/nuevo-nombredearchivo
```

Desplazamiento de archivos

Puede utilizar el comando **mv** para cambiar el nombre de un archivo o moverlo de un directorio a otro. Cuando utiliza **mv** para cambiar el nombre de un archivo, sólo debe usar el nuevo nombre de archivo como segundo argumento. El primero es el nombre actual del archivo. Si quiere cambiar el nombre a un archivo cuando lo mueve, especifique el nuevo nombre del archivo tras el de directorio. En el siguiente ejemplo, se cambia el nombre del archivo **propuesta** por **versión1**:

```
$ mv propuesta versión1
```

Al igual que con **cp**, es fácil borrar un archivo por accidente con **mv**. Cuando cambia el nombre de un archivo, puede elegir por accidente un nombre en uso de otro archivo. En este caso, ese otro se eliminará. El comando **mv** también tiene la opción **-i**, que revisa primero si ya existe un archivo con ese nombre.

También puede utilizar cualquier carácter especial descrito en el capítulo 3, para generar una lista de nombres de archivo que podrán utilizarse con **mv**. En el siguiente ejemplo, el usuario mueve todos los archivos de código fuente del actual al directorio **nuevoproy**:

```
$ mv *.c nuevoproy
```

Si quiere desplazar todos los archivos de un directorio determinado a otro, puede utilizar *****, para generar una lista de todos los archivos. En el siguiente ejemplo, el usuario mueve todos los archivos del directorio **reportes** a **resrep**:

```
$ mv reportes/* resrep
```

NOTA En GNOME o KDE, la manera más sencilla de copiar archivos a un disco CD-R/RW o DVD-R/RW consiste en utilizar la capacidad de grabadora incluida en el escritorio. Sólo inserte un disco en blanco, ábralo como carpeta, arrastre y coloque los archivos en éste. Se le preguntará automáticamente si quiere grabar los archivos. También puede utilizar cualquier cantidad de herramientas de grabación de CD/DVD, como K3B.

Copia y desplazamiento de directorios

También puede copiar o desplazar todo un directorio de una sola vez. Ambos comandos, **cp** y **mv**, pueden tomar como primer argumento el nombre del directorio, permitiéndole copiar y mover subdirectorios de un directorio a otro (véase la tabla 6-5). El primer argumento es el nombre del directorio que se desplazará o copiará, mientras el segundo argumento es el nombre del directorio en el cual se colocará. La misma estructura de nombre de ruta utilizada para archivos, aplica para el desplazamiento o copia de directorios.

De igual manera, puede copiar subdirectorios de un directorio a otro. Para copiar un directorio, el comando **cp** requiere utilice la opción **-r**. El nombre de esta opción viene de “recursivo”. Indica al comando **cp** copie un directorio, además de cualquier subdirectorio que puede contener. En otras palabras, todo el subárbol del directorio, desde ese directorio, se copiará. En el siguiente ejemplo, el directorio **gracias** se copia a **cartasanteriores**. Ahora existen dos subdirectorios **gracias**, uno en **cartas** y otro en **cartasanteriores**.

```
$ cp -r cartas/gracias cartasanteriores
$ ls -F cartas
/gracias
$ ls -F cartasanteriores
/gracias
```

Eliminación de archivos y directorios: el comando rm

Mientras emplea Linux, encontrará que el número de archivos en uso se incrementa rápidamente. Generar archivos en Linux es sencillo. Los editores y comandos como **cp** crean archivos de manera sencilla. Con el tiempo, muchos de estos archivos se volverán viejos y obsoletos. Entonces podrá eliminarlos con el comando **rm**. Este comando puede tomar cualquier número de argumentos, permitiéndole mostrar una lista de varios nombres de archivo para eliminar todos al mismo tiempo. En el siguiente ejemplo, el usuario elimina el archivo **propanterior**:

```
$ rm propanterior
```

Tenga cuidado cuando utilice el comando **rm**, porque es irreversible. Una vez elimine un archivo, no podrá restaurarlo (no existe deshacer). Con la opción **-i**, se le pregunta por separado si quiere eliminar el archivo. Si inserta **y**, el archivo se eliminará. Si inserta cualquier otra cosa, el archivo no se elimina. En el siguiente ejemplo, se instruye al comando **rm** para eliminar los archivos **propuesta** y **propanterior**. El comando **rm** después le pide confirmación para cada archivo. El usuario decide eliminar **propanterior**, pero no **propuesta**.

```
$ rm -i propuesta propanterior
Remove propuesta? n
Remove propanterior? y
$
```

Vínculos: el comando ln

Puede darle a un archivo más de un nombre, con el comando **ln**. Tal vez quiera hacer referencia a un archivo utilizando diferentes nombres de archivo, para acceder a éste desde otros directorios. A los nombres agregados suele conocerseles como *vínculos*. Linux soporta dos tipos diferentes de vínculos, duros y simbólico. Los vínculos *duros* son, literalmente, otros nombres para el mismo archivo, mientras los vínculos *simbólicos* funcionan más como accesos rápidos refiriendo a otro archivo. Los vínculos simbólicos son mucho más flexibles y funcionan en muchos sistemas de archivos diferentes, mientras los vínculos duros se limitan a su sistema de archivos local. Además, los vínculos duros introducen temas de seguridad, porque permiten acceso directo desde un vínculo para tener acceso público a un archivo original que tal vez quiera protegido. Debido a esto, los vínculos suelen implementarse como vínculos simbólicos.

Vínculos simbólicos

Para configurar un vínculo simbólico, utilice el comando **ln** con la opción **-s** y dos argumentos: nombre del archivo original y nuevo nombre de archivo agregado. La operación **ls** muestra una lista de nombres de archivo, pero sólo existirá un archivo físico.

```
$ ln -s nombre-archivo-original nombre-archivo-agregado
```

En el siguiente ejemplo, al archivo **hoy** se le da un nombre adicional **clima**. Sólo es otro nombre para el archivo **hoy**.

```
$ ls
hoy
$ ln -s hoy clima
$ ls
hoy clima
```

Puede asignar varios nombres al mismo archivo usando el comando **ln** en el mismo archivo varias veces. En el siguiente ejemplo, al archivo **hoy** se le dan los nombres **clima** y **findesemana**:

```
$ ln -s hoy clima
$ ln -s hoy findesemana
$ ls
hoy clima findesemana
```

Si muestra una lista de información completa acerca de un vínculo simbólico y su archivo, encontrará que la información desplegada es diferente. En el siguiente ejemplo, el usuario muestra una lista de información completa para **lunch** y **/home/george/listavegs** al utilizar el comando **ls**, con la opción **-l**. El primer carácter en la línea especifica el tipo de archivo. Los vínculos simbólicos tienen su propio tipo de archivo, representado por una **l**. El tipo de archivo para **lunch** es **l**, indicado en su vínculo simbólico, no un archivo ordinario. El número después del termino “group” es el tamaño del archivo. Observe que los tamaños difieren. El tamaño del archivo **lunch** es sólo de cuatro bytes. Esto es porque **lunch** sólo es un vínculo simbólico (un archivo almacenando el nombre de ruta de otro archivo) y un nombre de ruta sólo ocupa unos pocos bytes. No es un vínculo duro, directo al archivo **listavegs**.

```
$ ls -l lunch /home/george/listavegs
-rw-rw-r-- 1 george group 793 Feb 14 10:30 listavegs
lrw-rw-r-- 1 chris group 4 Feb 14 10:30 lunch
```

Para eliminar un archivo, sólo necesita eliminar su nombre original (y cualquier vínculo duro con éste). Si queda cualquier vínculo simbólico, no podrá acceder al archivo. En este caso, un vínculo simbólico almacenará el nombre de ruta de un archivo inexistente.

Vínculos duro

Puede dar al mismo archivo varios nombres, mediante el comando **ln** en el mismo archivo muchas veces. Para configurar un vínculo duro, use el comando **ln** con la opción **-s** y dos argumentos: nombre del archivo original y nuevo nombre de archivo agregado. La operación **ls** muestra una lista de ambos nombres de archivo, pero sólo existirá un archivo físico.

```
$ ln nombre-archivo-original nombre-archivo-agregado
```

En el siguiente ejemplo, al archivo **lunes** se da el nombre adicional **tormenta**. Es sólo otro nombre para el archivo **lunes**.

```
$ ls
hoy
$ ln lunes tormenta
$ ls
lunes tormenta
```

Para borrar un archivo con vínculos duros, necesita eliminar todos sus vínculos duros. El nombre de un archivo se considera realmente un vínculo a ese archivo (de allí que el comando **rm** elimine el vínculo a ese archivo). Si tiene varios vínculos a ese archivo y sólo elimina uno de ellos, los otros permanecerán en su lugar y puede hacer referencia al archivo a través de estos. También funcionará cualquier vínculo agregado. En el siguiente ejemplo, el archivo **hoy** se elimina con el comando **rm**. Sin embargo, existe un vínculo a ese mismo archivo, llamado **clima**. Luego, puede hacerse referencia al archivo bajo el nombre **clima**.

```
$ ln hoy clima
$ rm hoy
$ cat clima
La tormenta terminó hoy
Y luego salió el sol.
$
```

NOTA Cada archivo y directorio en Linux contiene un conjunto de permisos determinando quién accede a éstos y cómo. Puede configurar estos permisos para limitar el acceso por una de tres maneras: restringir el acceso sólo a usted mismo, permitir el acceso a usuarios de un grupo o permitir que cualquier usuario de su sistema tenga acceso. También puede controlar la manera en que se accede a un archivo o directorio dado. Un archivo o directorio puede tener permisos de lectura, escritura y ejecución. Cuando se crea un archivo, automáticamente se da permiso para lectura y escritura al propietario, permitiéndole desplegar y modificar el archivo. Puede cambiar los permisos a cualquier combinación deseada (consulte el capítulo 28 para conocer más detalles).

Las utilerías de mtools: msdos

Su sistema Linux proporciona un conjunto de utilerías, conocidas como mtools, para acceder de manera sencilla a discos flexibles o duros formateados para MS-DOS. Sólo trabajan con el antiguo sistema de archivos de MS-DOS o FAT32, no con Windows Vista, XP, NT o 2000, usando un sistema de archivos NTFS. El comando **mcopy** permite copiar archivos a un disco flexible de MS-DOS y desde éste, o en una partición FAT32 de Windows en su disco duro. No se necesitan operaciones especiales, como el montaje. Con mtools no necesita montar una partición de MS-DOS para acceder a ésta. En el caso de un disco flexible de MS-DOS, una vez haya colocado su disco, puede utilizar los comandos de mtool para acceder a esos archivos. Por ejemplo, para copiar un archivo desde un disco flexible de MS-DOS a su sistema Linux, utilice el comando **mcopy**. Especifique el disco MS-DOS con **a**: para la unidad A. A diferencia de los nombres de ruta normales de DOS, los nombres de ruta utilizados con los comandos de mtool requieren diagonales normales en lugar de diagonales invertidas. Se hará referencia al directorio **docs** en una unidad A por el nombre de ruta **a:/docs**, no **a:\docs**. A diferencia de MS-DOS, en el que, como opción predeterminada, el segundo argumento es el directorio actual, siempre necesita proporcionar el segundo argumento para **mcopy**.

En el siguiente ejemplo se copia el archivo **misdatos** al disco MS-DOS y después se copia el archivo **prefacio**, desde el disco al directorio actual de Linux.

```
$ mcopy misdatos a:
$ mcopy a:/prefacio
```

SUGERENCIA Puede utilizar **mtool** para copiar datos a un disco flexible formateado para Windows o una partición FAT32 de Windows, que se leen o escriben con Windows XP, también, pero no puede acceder a sistemas de archivo de discos duros (NTFS) de Windows Vista, XP, NT o 2000 con **mtools**. Las particiones NTFS requieren una herramienta diferente, el módulo NTFS del kernel.

Puede utilizar el comando **mdir** para una lista de archivos en su disco MS-DOS, asimismo el comando **mcd** para cambiar los directorios en éste. En el siguiente ejemplo se muestra una lista de archivos en el disco MS-DOS, en su unidad de disco flexible, que después se cambia al directorio **docs** en esa unidad:

```
$ mdir a:
$ mcd a:/docs
```

El acceso a MS-DOS o particiones Windows 95, 98 o Me con **mtools**, se configura mediante el archivo **/etc/mttools.conf**. El archivo muestra una lista de particiones predeterminadas de MS-DOS o Windows y discos duros. Cada unidad o partición se identifica con un nombre de dispositivo particular.

Archivado y compresión de archivos

Los archiveros se utilizan para respaldar archivos o combinarlos en un paquete, que después puede transferirse como archivo en Internet o puede publicarse en un sitio FTP para una descarga sencilla. La utilidad de archivero estándar usada en sistemas Linux y Unix es **tar**, para el que existen varias GUI. Se tienen varios programas de compresión para escoger, incluidos GNU zip (gzip), Zip, bzip y compress.

SUGERENCIA Puede utilizar la herramienta **unrar** para leer y extraer los populares archivos **rar**, pero no para crearlos. **unrar** está disponible desde **rpm.livna.org** y puede ser descargado e instalado con **yum**. **File Roller** está disponible para extraer archivos **RAR**, una vez la herramienta **unrar** está instalada. Otras fachadas gráficas, como **Xarchiver** y **Linrar**, están disponibles en **freshmeat**. Para crear archivos **rar**, debe comprar el archivador **Rarlab** en **rarlab.com**.

Archivado y compresión de archivos con File Roller

GNOME proporciona la herramienta **File Roller** (se accede a ella desde el menú Accesorios, etiquetado Gestor de archivadores), operable como GUI para archivar y comprimir archivos, permitiéndole realizar operaciones Zip, gzip, tar y bzip2 mediante una GUI. Puede examinar el contenido de los archiveros, extraer archivos deseados y crear nuevos archiveros comprimidos. Cuando crea un archivo, usted determina su método de compresión especificando extensión de nombre de archivo, como **.gz** para gzip o **.bz2** para bzip2. Puede seleccionar diferentes extensiones desde el menú Tipo de archivo o insertar la extensión usted mismo. Para archivar y comprimir archivos, puede seleccionar una extensión combinada como **.tar.bz2**, que archiva con tar y comprime con bzip2. Haga clic en **Agregar** para añadir archivos a su archivero. Para extraer

archivos y archivar, abra el archivo para desplegar la lista de nombres de archivo. Luego puede hacer clic en Extraer, para sacar archivos determinados o el archivero completo.

SUGERENCIA *File Roller también puede usarse para examinar el contenido de un archivero de manera sencilla. Desde el administrador de archivos, haga clic con el botón derecho en el archivador y seleccione Abrir con el administrador de archivos. Se desplegará la lista de archivos y directorios en ese archivero. En el caso de subdirectorios, haga doble clic en sus entradas. Este método también funciona para archivos de software RPM, permitiéndole explorar todos los archivos creados por el paquete de software.*

Archivos y dispositivos de archivero: tar

La utilidad tar crea archiveros para archivos y directorios. Con tar, puede integrar archivos específicos, actualizarlos en el archivero y agregar nuevos archivos al mismo. Incluso directorios completos con todos los archivos y subdirectorios, todo lo cual puede restaurarse desde el archivero. La utilidad tar fue diseñada originalmente para crear archiveros en cintas. (El término “tar” viene de tape archive [archivo de cinta]. Sin embargo, puede crear archiveros en cualquier dispositivo, como un disco flexible o crear un archivo de archivero para almacenar este último.) La utilidad tar es ideal para crear copias de seguridad de sus archivos o combinar varios archivos en uno, solo para transmisión a través de la red (File Roller es una GUI para tar).

NOTA *Como opción de tar, puede utilizar pax, diseñada para trabajar con diferentes tipos de formatos de archiveros Unix como cpio, bcpio y tar. Puede extraer, mostrar una lista y crear archiveros. La utilidad pax es útil si está administrando archiveros creados en sistemas Unix, manejando diferentes formatos de archivos.*

Despliegue del contenido de los archiveros

Ambos administradores de archivo en los escritorios GNOME y K Desktop tienen la capacidad para desplegar el contenido de un archivo de archivero tar automáticamente. El contenido se despliega como si fueran archivos en un directorio. Puede mostrar una lista de archivos como íconos o con detalles, ordenarlos por nombre, tipo u otros campos. Incluso puede desplegar el contenido de los archivos. Al hacer clic en un archivo de texto, se abre con un editor de texto y despliega una imagen con un visor de imagen. Si el administrador de archivos no puede determinar qué programa utilizar para desplegar el archivo, le pide seleccione una aplicación. Ambos administradores de archivos pueden realizar el mismo tipo de operaciones en archivos residentes en archiveros remotos, como archivos tar o sitios FTP. Puede obtener una lista del contenido e incluso leer sus archivos readme. El administrador de archivos Nautilus (GNOME) también puede extraer un archivo. Haga clic con el botón derecho en el icono Archivo y seleccione Extraer.

Creación de archiveros

En Linux, a menudo se usa tar para generar archiveros en dispositivos o archivos. Puede instruir a tar para almacenar los archivos en un dispositivo por determinado archivo usando la opción **f** con el nombre del dispositivo o archivo. En el siguiente ejemplo se muestra la sintaxis del comando **tar**, empleando la opción **f**. A menudo, se toma el nombre del dispositivo o archivo como nombre del archivero. Cuando crea un archivo para un archivero **tar**, suele añadirse a su nombre la extensión **.tar**. Esto sólo es una convención y no es obligatorio. Puede mostrar una lista de todos los nombres de archivo que quiera. Si se especifica un nombre de directorio, todos los subdirectorios se incluyen en el archivero.

```
$ tar opcioness nombre-archivero.tar nombres-directorio-y-archivo
```

Para crear un archivero, use la opción **c**. Combinada con **f**, **c** crea un archivero en un archivo o dispositivo. Puede ingresar esta opción antes y justo después de la opción **f**. Observe que no hay diagonales antes de la opción **tar**. En la tabla 6-6 se muestra una lista de diferentes opciones para **tar**. En el siguiente ejemplo, el directorio **midir** y todos sus subdirectorios se guardan en el archivo **miarch.tar**. En este ejemplo, el directorio **midir** almacena dos archivos, **misreuniones** y **fiesta**, además de un directorio llamado **reportes** conteniendo tres archivos: **clima**, **lunes** y **viernes**.

```
$ tar cvf miarch.tar midir
midir/
midir/reportes/
midir/reportes/clima
```

Comandos	Ejecución
tar <i>opciones</i> <i>archivos</i>	Crea una copia de seguridad de los archivos en cinta, en un dispositivo o archivo de archivero.
tar <i>opcionesf</i> <i>nombre_archivero</i> <i>lista-de-archivos</i>	Crea una copia de seguridad de los archivos en un archivo o dispositivo especificado por <i>nombre_archivero</i> . <i>lista-de-archivos</i> ; pueden ser los nombres de archivos o directorios.
Opciones	
c	Crea un nuevo archivero.
t	Muestra una lista de los nombres de los archivos en un archivero.
r	Adjunta archivos a un archivero.
U	Actualiza un archivero con elementos nuevos y modificados; sólo agrega archivos modificados desde que fueron archivados o archivos que no estaban presentes en el archivero.
--delete	Elimina un archivo del archivero.
w	Espera confirmación del usuario antes de archivar cada archivo; le permite actualizar un archivero selectivamente.
x	Extrae archivos de un archivero.
m	Cuando se está extrayendo un archivo de un archivero, no se asigna nueva marca de tiempo.
M	Crea un archivero de varios volúmenes para almacenarse en varios discos flexibles.
f <i>nombre-archivo</i>	Guarda el archivero en cinta con el nombre de archivero dado al archivo, en vez de hacerlo en el dispositivo predeterminado de cinta. Cuando le asigna un nombre al archivero, la opción f guarda el archivero tar en el archivo de ese mismo nombre.
f <i>nombre-dispositivo</i>	Guarda un archivo tar a un dispositivo como un disco flexible o una cinta. /dev/fd0 es el nombre del dispositivo para su disco flexible; the default device is held in /etc/default/tar-file .
v	Despliega cada nombre de archivo como si estuviera archivado.
z	Comprime o descomprime un archivero de archivos utilizando gzip .
j	Comprime o descomprime un archivero de archivos utilizando bzip2 .

TABLA 6-6 Archiveros de archivo: **tar**

```
midir/reportes/lunes
midir/reportes/viernes
midir/misreuniones
midir/fiesta
```

Extracción de archiveros

El usuario puede extraer después archivos y directorios del archivero con la opción **x**. La opción **xf** extrae archivos desde un archivo o dispositivo de archivero. La operación de extracción **tar** genera todos los subdirectorios. En el siguiente ejemplo, la opción **xf** instruye a **tar** para extraer todos los archivos y subdirectorios del archivo **tar** **miarch.tar**:

```
$ tar xvf miarch.tar
midir/
midir/reportes/
midir/reportes/clima
midir/reportes/lunes
midir/reportes/viernes
midir/misreuniones
midir/fiesta
```

Puede utilizar la opción **r** para agregar archivos a un archivero ya creado. La opción **r** adjunta archivos al archivero. En el siguiente ejemplo, el usuario adjunta los archivos del directorio **cartas** al archivero **miarch.tar**. Aquí, el directorio **misdocs** y sus archivos se agregan al archivo **miarch.tar**:

```
$ tar rvf miarch.tar misdocs
misdocs/
misdocs/doc1
```

Actualización de archiveros

Si cambia cualquiera de los archivos en directorios archivados previamente, puede usar la opción **u** para instruir a **tar** que actualice el archivero con archivos modificados. El comando **tar** compara la hora de la última actualización de cada archivo almacenado con los del directorio del usuario y copia en el archivero cualquier archivo modificado tras la última vez que se archivó. Cualquier archivo creado recientemente en estos directorios también se agrega al archivero. En el siguiente ejemplo, el usuario actualiza el archivo **miarch.tar** con cualquier archivo creado o modificado recientemente en el directorio **midir**. En este caso, el archivo **regalos** se agrega al directorio **midir**.

```
tar uvf miarch.tar midir
midir/
midir/regalos
```

Si necesita ver que archivos se almacenaron en un archivero, utilice el comando **tar** con la opción **t**. En el siguiente ejemplo se muestra una lista de todos los archivos almacenados en el archivero **miarch.tar**:

```
tar tvf miarch.tar
drwxr-xr-x root/root 0 2000-10-24 21:38:18 midir/
drwxr-xr-x root/root 0 2000-10-24 21:38:51 midir/reportes/
-rw-r--r-- root/root 22 2000-10-24 21:38:40 midir/reportes/clima
-rw-r--r-- root/root 22 2000-10-24 21:38:45 midir/reportes/lunes
-rw-r--r-- root/root 22 2000-10-24 21:38:51 midir/reportes/viernes
-rw-r--r-- root/root 22 2000-10-24 21:38:18 midir/ misreuniones
```



```
-rw-r--r-- root/root 22 2000-10-24 21:36:42 midir/ fiesta
drwxr-xr-x root/root 0 2000-10-24 21:48:45 misdocs/
-rw-r--r-- root/root 22 2000-10-24 21:48:45 misdocs/doc1
drwxr-xr-x root/root 0 2000-10-24 21:54:03 midir/
-rw-r--r-- root/root 22 2000-10-24 21:54:03 midir/regalos
```

Archivado en discos flexibles

Para crear copias de seguridad de archivos en un dispositivo específico, señale el dispositivo como si fuera el archivero. Asegúrese de emplear un disco flexible vacío. Cualquier dato colocado previamente en el disco se borrará con esta operación. En el siguiente ejemplo, el usuario crea un archivero en el disco flexible en el dispositivo **/dev/fd0** y copia en éste todos los archivos en el directorio **midir**:

```
$ tar cf /dev/fd0 midir
```

Para extraer archivos de copia de seguridad en el disco del dispositivo, utilice la opción **xf**:

```
$ tar xf /dev/fd0
```

Compresión de archiveros

La operación **tar** no realiza compresión de archivos archivados. Si quiere comprimirlos, puede instruir a **tar** para invocar la utilidad **gzip**. Con la opción **z** minúscula, **tar** primero utiliza **gzip** para comprimir archivos antes de archivarlos. La misma opción **z** invoca **gzip** para descomprimirlos cuando se extraen los archivos.

```
$ tar czf miarch.tar.gz midir
```

Si desea recurrir a **bzip** en vez de **gzip** para comprimir los archivos antes de archivarlos, use la opción **j**. La misma opción **j** invoca **bzip** para descomprimirlos cuando se extraigan los archivos.

```
$ tar cjf miarch.tar.bz2 midir
```

Recuerde que existe diferencia entre comprimir archivos individuales en un archivero y comprimir el archivero como un todo. A menudo, un archivero se crea para transferir varios archivos a la vez como archivo **tar**. Para acortar el tiempo de transmisión, el archivo debe lo más pequeño posible. Puede utilizar la utilidad de compresión **gzip** en el archivo **tar** para comprimirlo, reduciendo su tamaño y después enviar la versión comprimida. La persona que lo recibe puede descomprimirlo, restaurando el archivo **tar**. El uso de **gzip** en un archivo **tar** a menudo produce un archivo con la extensión **.tar.gz**. La extensión **.gz** se agrega a un archivo **zip** comprimido. En el siguiente ejemplo se crea una versión comprimida de **miarch.tar** empleando el mismo nombre con la extensión **.gz**:

```
$ gzip miarch.tar
$ ls
$ miarch.tar.gz
```

En vez de escribir otra vez el comando **tar** para diferentes archivos, puede colocar el comando en una secuencia de comandos y pasar los archivos a ésta. Asegúrese de hacer ejecutable la secuencia de comandos. En el siguiente ejemplo, se crea una secuencia de comandos simple **miprograrch**, que archivará nombres de archivo mostrados en la lista como argumentos.

```
miprogarch
tar cvf miarch.tar $*
```

Aquí se muestra la ejecución de una secuencia de comandos con varios argumentos:

```
$ miprogarch misdatos prefacio
misdatos
prefacio
```

Archivado en cinta

Si tiene un dispositivo predeterminado señalado, como una cinta, y quiere crear un archivero en él, sólo debe usar **tar** sin la opción **f** y un dispositivo o nombre de archivo. Esto es útil para crear copias de seguridad de sus archivos. El nombre de un dispositivo predeterminado se almacena en un archivo llamado **/etc/default/tar**. En el siguiente ejemplo se muestra la sintaxis del comando **tar** manejada por el dispositivo de cinta predeterminado. Si se especifica un nombre de directorio, todos los subdirectorios se incluyen en el archivero.

```
$ tar opcion nombres-de-directorio-y-archivo
```

En el siguiente ejemplo **midir** y todos sus subdirectorios se guardan en una cinta del dispositivo de cinta predeterminado:

```
$ tar c midir
```

En este ejemplo, el directorio **midir**, todos sus archivos y subdirectorios se extraen del dispositivo de cinta predeterminado, para colocarse en el directorio de trabajo del usuario:

```
$ tar x midir
```

***NOTA** Puede utilizar otros programas para archivar, como **cpio**, **pax** y **shar**. Sin embargo, **tar** es el más común para cualquier aplicación de software de archivero.*

Compresión de archivos: **gzip**, **bzip2** y **zip**

Existen varias razones para reducir tamaño de un archivo. Las dos más comunes son liberar espacio o, si quiere transferir el archivo en una red, para ahorrar tiempo de transmisión. Reducirá de manera efectiva el tamaño de un archivo creando una copia comprimida de éste. En cualquier momento que necesite de nuevo el archivo, lo puede descomprimir. La compresión se usa junto con el archivado para permitirle comprimir directorios completos y sus archivos de una sola vez. La descompresión genera una copia del archivo del archivero, que después puede ser ejecutado, generando una copia de archivos y directorios. File Roller proporciona una GUI para estas tareas.

Compresión con **gzip**

Varias utilerías de compresión están disponibles en sistemas Linux y Unix. Casi todo el software de sistemas Linux recurre a las utilerías GNU **gzip** y **gunzip**. La utilería **gzip** comprime archivos y **gunzip** los descomprime. Para comprimir un archivo, inserte el comando **gzip** y el nombre del archivo. Esto reemplaza el archivo con una versión comprimida, con la extensión **.gz**.

```
$ gzip misdatos
$ ls
misdatos.gz
```

Para descomprimir un archivo **gzip**, utilice ya sea **gzip** con la opción **-d** o el comando **gunzip**. Estos comandos descomprimen un archivo comprimido con la extensión **.gz** y lo reemplazan por una versión descomprimida con el mismo nombre raíz, pero sin la extensión **.gz**. Ni siquiera necesita escribir la extensión **.gz**; **gunzip** y **gzip -d** la asumen.

En la tabla 6-7 se muestra una lista con diferentes opciones de **gzip**.

```
$ gunzip misdatos.gz
$ ls
misdatos
```

SUGERENCIA En su escritorio, puede extraer el contenido de un archivero al localizarlo con el administrador de archivos y hacer doble clic en éste. También puede hacer clic con el botón derecho y elegir Abrir con el Administrador de archivos. Esto iniciará la aplicación File Roller, que abrirá el archivero, mostrando una lista de contenidos. Entonces puede elegir que se extraiga el archivo. File Roller utilizará las herramientas apropiadas para descomprimir el archivo (**bzip2**, **zip** o **gzip**), si está comprimido y después extraerá el archivero (**tar**).

También puede comprimir archivos **tar** archivados. Esto da como resultado archivos con extensión **.tar.gz**. Los archivos de archivero comprimidos a menudo se usan para transmitir archivos demasiado grandes en redes.

```
$ gzip miarch.tar
$ ls
miarch.tar.gz
```

Opción	Ejecución
-c	Envía una versión comprimida del archivo a la salida estándar; cada archivo de la lista se comprime por separado: <code>gzip -c misdatos prefacio > misarchivos.gz</code>
-d	Descomprime un archivo comprimido; o puede utilizar gunzip : <code>gzip -d misarchivos.gz</code> <code>gunzip misarchivos.gz</code>
-h	Despliega listas de ayuda.
-l listaarchivo	Despliega el tamaño de cada archivo comprimido y descomprimido de la lista: <code>gzip -l misarchivos.gz</code>
-r nombre-directorio	Busca repetitivamente directorios especificados y comprime todos los archivos en éstos; la búsqueda comienza en el directorio de trabajo actual. Cuando se utiliza con gunzip , se descomprimen archivos comprimidos de un directorio especificado.
-v lista-archivo	Para cada archivo comprimido o descomprimido, se despliega nombre y porcentaje de su reducción de tamaño.
-num	Determina velocidad y tamaño de la compresión; el rango es de -1 a -9. Números más bajos dan mayor velocidad pero menor compresión, lo que da como resultado un archivo más grande que se comprime y descomprime rápidamente. Por tanto, -1 da la compresión más rápida pero con el tamaño más grande; -9 produce un archivo muy pequeño pero toma más tiempo para comprimir y descomprimir. La opción predeterminada es -6.

TABLA 6-7 Las opciones de **gzip**

Puede comprimir los miembros del archivo tar individualmente con la opción **tar z** que invoca a gzip. Con la opción **z**, tar invoca a gzip para comprimir un archivo antes de colocarlo en un archivero. Sin embargo, no puede actualizarse un archivero con miembros comprimidos con la opción **z**, ni es posible agregar algo a ellos. Todos los miembros deben comprimirse y agregarse al mismo tiempo.

Los comandos **compress** y **uncompress**

También puede utilizar los comandos **compress** y **uncompress** para crear archivos comprimidos. Estos generan un archivo con extensión **.Z** y emplean un formato de compresión diferente desde gzip. Los comandos **compress** y **uncompress** no se utilizan de manera amplia, pero en ocasiones puede encontrarse con archivos **.Z**. Puede utilizar el comando **uncompress** para descomprimir un archivo **.Z**. gzip es la utilidad de compresión GNU estándar y debe utilizarse en vez de **compress**.

Compresión con **bzip2**

Otra utilidad de compresión popular es **bzip2**. Ésta comprime archivos utilizando un algoritmo de compresión de texto de ordenamiento por bloques Burrows-Wheeler y codificación Huffman. Las opciones de línea de comandos son parecidas a gzip por diseño, pero no son exactamente iguales. (Consulte la página Man bzip2 para conocer una lista completa.) Usted comprime archivos utilizando el comando **bzip2** y descomprime con **bunzip2**. El comando **bzip2** crea archivos con la extensión **.bz2**. Puede utilizar **bzcat** para dirigir la salida de datos comprimidos a la salida estándar. El comando **bzip2** comprime archivos en bloques y le permite especificar el tamaño (bloques más largos dan mayor compresión). Cuando utiliza gzip, puede recurrir a bzip2 para comprimir archivos de archivero tar. En el siguiente ejemplo se comprime el archivo **misdatos** en un archivo comprimido bzip con la extensión **.bz2**:

```
$ bzip2 misdatos
$ ls
misdatos.bz2
```

Para descomprimir, maneje el comando **bunzip2** en un archivo bzip:

```
$ bunzip2 misdatos.bz2
```

Uso de **zip**

Zip es una utilidad de compresión y archivado modelada en PKZIP, usada originalmente en sistemas DOS. Zip es una utilidad de varias plataformas, utilizada en sistemas Windows, Mac, MS-DOS, OS/2, Unix y Linux. Los comandos Zip funcionan con archivos creados con PKZIP y pueden utilizar archivos Zip. Usted comprime un archivo con el comando **zip**. Esto genera un archivo Zip con extensión **.zip**. Si no hay archivos en la lista, **zip** dirige la salida de los datos comprimidos a la salida estándar. También puede utilizar el argumento **-r** para que **zip** lea desde la entrada estándar. Para comprimir un directorio, necesita incluir la opción **-r**. En el primer ejemplo se archiva y comprime un archivo:

```
$ zip misdatos
$ ls
misdatos.zip
```

En el siguiente ejemplo se archiva y comprime el directorio **reportes**:

```
$ zip -r reportes
```

Se da soporte a un conjunto completo de operaciones. Con la opción **-f**, puede actualizar un archivo particular en el archivero Zip con una nueva versión. La opción **-u** reemplaza y agrega archivos, mientras la opción **-d** elimina archivos del archivero Zip. También hay opciones para archivos codificados, haciendo traducciones de final de línea de DOS a Unix e incluyendo archivos ocultos.

Para descomprimir y extraer archivos Zip, necesita utilizar el comando **unzip**.

```
$ unzip misdatos.zip
```

Escritorio

Capítulo 7

X Windows System, Xorg y administradores de pantalla

Capítulo 8

GNOME

Capítulo 9

KDE

X Windows System, Xorg y administradores de pantalla

Los sistemas Unix y Linux utilizan la misma utilería gráfica estándar conocida como X Windows System, X ó X11. Esto significa que, en casi todos los casos, un programa basado en X puede ejecutarse en cualquiera de los administradores de ventana y escritorios. El software basado en X a menudo se encuentra en sitios FTP de Linux o Unix, en directorios etiquetados **X11**. Puede descargar estos paquetes y ejecutarlos en cualquier administrador de ventana ejecutado en su sistema Linux. Tal vez algunos estén en forma de binarios Linux que puede descargar, instalar y ejecutar directamente. Netscape es un ejemplo. Otros están en forma de código fuente para ser configurados, compilados e instalados de manera sencilla en su sistema con unos cuantos comandos simples. Es posible que algunas aplicaciones, como Motif, necesiten bibliotecas especiales.

X Windows System está diseñado para dar flexibilidad (puede configurarlo de varias formas). Es posible la ejecución de X Windows System con casi todas las tarjetas de video disponibles, sin estar atado a una interfaz de escritorio específica. Proporciona un conjunto básico de operaciones gráficas que las aplicaciones de interfaz de usuario como los administradores de ventanas, de archivos e incluso escritorios pueden utilizar. Un administrador de sistema utiliza estas operaciones para construir widgets que permitan manipular ventanas, como barras de desplazamiento, controladores de cambio de tamaño y cuadros de cierre. Diferentes administradores de ventana los construyen para tener un aspecto distinto, proporcionando interfaces con diferentes apariencias. Todos los administradores de ventana funcionan con X Windows System. Puede elegir entre varios administradores de ventanas diferentes y cada usuario de un sistema puede ejecutar un administrador de ventanas diferente, utilizando las operaciones gráficas básicas de X Windows System. Incluso es posible ejecutar programas X sin administrador de ventanas o archivos.

Para ejecutar X Windows System, necesita instalar un servidor X Windows System. Las versiones gratuitas del servidor X Windows System se ofrecen por el proyecto original XFree86 (xfree86.org) y el más reciente X.org Foundation (www.x.org). El proyecto XFree86, aunque es gratis y de fuente abierta, utiliza licencia propia. Por esta razón, el proyecto X.org tomó otra dirección, para desarrollar una versión de la licencia pública GNU's Not Unix (GNU) de X Windows System. Actualmente, la versión X.org se utiliza en casi todas las distribuciones de Linux. La configuración para ambas implementaciones permanece igual, con cambio de nombre del archivo

de configuración de **Xfree86.conf** a **xorg.conf**, para la versión X.org Foundation. Los dos grupos también utilizan conversiones de nombre diferentes para sus versiones. XFree86 emplea su propia numeración, actualmente 4.6, mientras X.org se ajusta a las versiones de X Windows System, actualmente X11R7.2. Este capítulo se concentra en la versión X.org, puesto que es el más utilizado. Tenga en cuenta que configuración y organización son casi las mismas para XFree86.

Una vez instala el servidor Xorg, debe proporcionar información de configuración sobre su monitor, ratón y teclado. Esta información se utiliza en un archivo de configuración denominado **/etc/X11/xorg.conf**, incluyendo información técnica mejor generada por un programa de configuración de X Windows System, como Xorgconfig, xlixd o XF86Setup. Cuando configura X Windows System al instalarlo en su sistema, el archivo se genera automáticamente.

También puede configurar su propia interfaz X empleando archivos de configuración **.xinitrc** y **/etc/X11/xinit/xinitrc**, donde pueden seleccionarse o iniciarse administradores de ventana, de archivo y aplicaciones X iniciales. Asimismo, usar un conjunto de comando X especializados para configurar su ventana raíz, cargar fuentes o configurar recursos de X Windows System, como el color de los bordes de las ventanas. También puede descargar utilerías X de recursos en línea sirviendo como sitios espejo de Linux, generalmente en su directorio **/pub/Linux/X11**. Si debe compilar una aplicación X, tal vez tenga que usar procedimientos especiales, además de instalar paquetes de soporte. Un recurso oficial de noticias, herramientas y administradores de sistema de X Windows System es **www.x.org**. Allí puede encontrar información detallada acerca de las características de X Windows System, junto con escritorios y administradores de ventana compatibles.

X Windows System fue desarrollado y es mantenido por The Open Group (TOG), un consorcio de más de cien compañías, incluidas Sun, HP, IBM, Motorola e Intel (**opengroup.org**). El desarrollo es administrado por el grupo X.org en representación de TOG. X.org es una organización no lucrativa que mantiene el código existente de X Windows System. Periódicamente, X.org proporciona versiones actualizadas, oficiales y gratuitas de X Windows System al público general. Controla el desarrollo de especificaciones X11R6, trabajando con grupos apropiados para revisar y lanzar actualizaciones al estándar. Xorg es una versión de los servidores X Windows System distribuida de manera gratuita y utilizada en casi todos los sistemas Linux. Conocerá más acerca de Xorg en **www.x.org**.

El protocolo X

El protocolo X se desarrolló para sistemas Unix a mediados de la década de 1980, con el fin de proporcionar una interfaz gráfica de usuario (GUI) de red transparente. El protocolo X organiza y despliega operaciones en una relación cliente-servidor, en la que un cliente envía peticiones de despliegue a un servidor. Al cliente se le conoce como cliente X y al servidor como servidor X. El cliente, en este caso, es una aplicación, mientras el servidor es una pantalla. Esta relación separa aplicaciones del servidor. La aplicación actúa como cliente enviando peticiones a un servidor, que realmente hace el trabajo de la operación de despliegue solicitada. Esto tiene por ventaja permitir que el servidor interactúe con el sistema operativo y los dispositivos, mientras la aplicación no necesita saber de estos detalles. Una aplicación operando como cliente X despliega cualquier sistema usando un servidor X. De hecho, un cliente remoto X puede enviar peticiones para que un servidor X en una máquina local realice ciertas operaciones de despliegue. En efecto, la relación servidor/cliente X se invierte, considerando la forma en que normalmente se piensa en los servidores. Generalmente, varios sistemas de cliente acceden a un solo servidor. El modelo de servidor X tiene a cada sistema operando como un servidor X al que se accede desde un solo sistema almacenando programas de cliente X.

Xorg

La X.org Foundation (www.x.org) es una organización no lucrativa proporcionando de manera gratuita servidores y materiales de soporte a X Windows System para varios sistemas operativos en PC y otras minicomputadoras. X.org Foundation proporciona servidor, programas de cliente y documentación de X, suele conocerseles además como Xorg. El servidor Xorg está disponible de manera gratuita e incluye código fuente. El proyecto es financiado completamente por donaciones.

Xorg utiliza un servidor, denominado Xorg X server, con paquetes de controladores adicionales para su tarjeta de video específica. Sólo necesita instalar el paquete del servidor Xorg X, junto con paquetes de soporte básico, como los relacionados con fuentes, además de controladores para su tarjeta de video: xf86-video-ati-X11R6 para tarjetas Ati o xf86-video-nv-X11R6 para tarjetas Nvidia. El servidor Xorg X tendrá soporte para tarjetas de video dadas y monitores implementados como bibliotecas estáticas o módulos que pueden cargarse cuando se necesiten. En la actualidad, los servidores Xorg X soportan plataformas Intel, Alpha, PowerPC y Sparc. El servidor Xorg soporta gran variedad de tarjetas de video y monitores, incluidas tarjetas de video monocromáticas, VGA, Super VGA y las tarjetas de video con aceleración de hardware.

Por lo general, su distribución de Linux notificará de cualquier actualización para Xorg mediante herramientas de actualización. Después pueden descargarse las actualizaciones e instalarse automáticamente. Siempre es preferible descargar desde los sitios de distribución de Linux, porque es probable que esos paquetes estén modificados para trabajar mejor con su sistema. La versión completa del software de Xorg incluye el servidor Xorg X y sus módulos, incluidos varios paquetes de soporte como los relacionados con fuentes y archivos de configuración. En la tabla 7-1 se muestra una lista de paquetes Xorg actuales. Como opción, puede descargar el código fuente de nuevas actualizaciones en el sitio Web X.org. En el caso de las versiones de código fuente, se recomienda usar el instalador Xinstall.sh. Éste pedirá información de instalación y después descargará e instalará los componentes de Xorg necesarios.

Además del servidor, Xorg incluye programas de soporte y bibliotecas de desarrollo. Las aplicaciones y servidores Xorg se instalan en el directorio **/usr/bin**. Bibliotecas de soporte, como el módulo específico de la tarjeta de video, se instalan en el directorio **/usr/X11R6/lib**. Documentación de los diferentes paquetes se encuentra en **/usr/share/doc**; los directorios de paquetes empiezan con el prefijo **xorg**. Una copia de documentación detallada de todos los componentes X.org se encontrará en **/usr/share/X11/doc**. La página Man del servidor X.org es **Xorg**, y la aplicación de servidor es **/usr/bin/Xorg**. Los archivos de configuración se colocan en el directorio **/etc/X11**. Las aplicaciones escritas para soporte de X suelen instalarse en el directorio **/usr/bin**.

Directorio	Descripción
/usr/X11R6/lib	Bibliotecas de soporte
/usr/bin	Programas (clientes y servidores X Windows System)
/usr/include/X11	Archivos de encabezado de desarrollo
/usr/share/man/X11	Páginas Man
/usr/share/X11/doc	Documentación
/usr/share/X11	Archivos de configuración y soporte de System X11
/etc/X11	Archivos de configuración

TABLA 7-1 Directorios de Xorg

Herramienta	Descripción
xorgcfg	Herramienta de configuración de X Windows System basada en pantalla de Xorg
Xorg -configure	Herramienta de configuración X Windows System de Xorg, construida en el servidor Xorg X
xorgconfig	Antigua herramienta de configuración de Xorg
Sax2	Herramienta de configuración SUSE X Windows System
/etc/X11/xorg.conf	Archivo de configuración de X Windows System; editado por las herramientas de configuración

TABLA 7-2 Herramientas de configuración de X Window System

También encontrará servidores **Xorg** y programas de soporte. En la tabla 7-2 se muestra una lista de directorios de configuración de Xorg.

NOTA *Xorg ahora incluye Direct Rendering Interface (DRI) y soporte a OPenGL (GLX), para tarjetas de video 3-D como ATI y Nvidia.*

Puede usar servidores X para ejecutar aplicaciones de X Windows System en un sistema remoto. Cuando accede a uno de estos sistemas, puede hacer que el servidor X de ese sistema genere un nuevo despliegue, para ejecutar la aplicación X remota. Cada servidor X tiene un nombre de despliegue constando de un nombre de host, número de despliegue y número de pantalla. Las aplicaciones los utilizan para determinar cómo conectarlas al servidor y la pantalla que debe usarse.

nombrehost:númerodespliegue.númerodepantalla

El nombre host es donde está localizado físicamente el servidor X. El número de despliegue administra el servidor X. En una estación de trabajo local, sólo suele existir un despliegue. Sin embargo, en un sistema de varios usuarios, donde existen varias terminales (cada una con su propio ratón y teclado) conectadas a un solo sistema, cada terminal tiene pantalla propia con número de despliegue. De esta forma, varios usuarios ejecutan aplicaciones X simultáneamente en el mismo servidor X. Si su sistema tiene dos o más monitores compartiendo mismo teclado y ratón, un número de pantalla diferente se aplicará a cada monitor, a pesar de que tendrán el mismo número de despliegue.

El despliegue utilizado por un usuario se muestra en la lista como variable de entorno DISPLAY. En un sistema de un solo usuario, encontrará que la entrada del despliegue comienza con dos puntos seguidos por un 0, como se muestra aquí. Esto indica que el servidor X está en el sistema local (no un host remoto) y tiene el número de despliegue 0.

```
$ echo $DISPLAY
:0
```

Para utilizar una aplicación remota X, debe cambiar el nombre de despliegue por la variable DISPLAY. Puede hacer esto de forma manual, asignando nuevo nombre de host y de despliegue a la variable o usar la secuencia de comandos **xon**:

```
$ DISPLAY=conejo.mipista.com:0
$ export DISPLAY
```

Utilice la opción **-display** al invocar una aplicación X para especificar el servidor remoto X que utilizará:

```
$ xterm -display conejo.mipista.com:0
```

Configuración de Xorg: /etc/X11/xorg.conf

Los servidores Xorg proporcionan gran rango de soporte a hardware, pero puede ser todo un reto configurarlos. Consulte los documentos HOWTO de X Window en tldp.org o el directorio **/usr/share/doc/** para la mayoría de distribuciones. También existen páginas Man para Xorg y **xorg.conf**, se dispone también de documentación y FAQs en www.x.org. El archivo de configuración empleado para su servidor Xorg se denomina **xorg.conf**, localizado en el directorio **/etc/X11**. **xorg.conf** contiene todas las especificaciones de su tarjeta gráfica, monitor, teclado y ratón. Para configurar el archivo **xorg.conf**, necesita especificar la información de su hardware. Para su monitor, debe conocer los rangos de frecuencia de sincronización tanto horizontal como vertical y ancho de banda. En el caso de tarjetas gráficas, necesita saber los parámetros de la tarjeta de video y quizás también deba conocer los relojes. En el caso de su ratón, si es compatible con Microsoft o cualquier otra marca, como Logitech, también el puerto al que está conectado.

A pesar de que puede crear y editar el archivo directamente, es preferible usar su herramienta de configuración de pantalla de la distribución. Xorg detectará automáticamente su configuración y generará un archivo **xorg.conf** apropiado. Incluso puede iniciar sin un archivo de configuración **xorg.conf**. En la tabla 7-2 se muestra una lista de varias herramientas y archivos de configuración.

Como opción, puede emplear la utilería de configuración Xorg, integrada en el servidor Xorg. Puede utilizar el comando **xorg** con la opción **-configure**. Esto detectará y generará automáticamente un archivo de configuración **xorg.conf**. El archivo tendrá el nombre **xorg.conf.new** y se colocará en su directorio raíz. Para utilizar este comando primero debe salir del servidor X. Esto requiere cambiar niveles de ejecución, pasando de una interfaz gráfica a la línea de comandos. En muchas distribuciones, la interfaz de línea de comandos se ejecuta en un nivel de ejecución 3 (las excepciones son Debian y Ubuntu). Puede utilizar el comando **telinit** para cambiar niveles de ejecución. Si la interfaz de línea de comando trabaja en el nivel de ejecución 3, por ejemplo, puede utilizar el comando **telinit 3** para cambiarlo. Esto termina el escritorio y pide que inicie sesión al utilizar la línea de comandos.

```
telinit 3
```

Inicie sesión como root y luego ejecute el comando **Xorg -configure**.

```
Xorg -configure
```

Puede probar después el nuevo archivo de configuración **xorg** con X para ver si funciona. Use la opción **-config**. Una vez en funcionamiento, puede cambiar el nombre del original y después el del nuevo como **/etc/X11/xorg.conf**.

```
X -config /root/xorg.conf.new
```

Como opción, puede utilizar **xorgcfg** o el más antiguo **xorgconfig**. Con estas herramientas, sólo responde preguntas acerca de su hardware o selecciona opciones y el programa genera el archivo apropiado **/etc/X11/xorg.conf**. Para una configuración difícil, necesitará editar el archivo **xorg.conf** directamente. Generalmente, sólo se necesitan pequeñas ediciones en el archivo generado automáticamente.

El archivo `/etc/X11/xorg.conf` está organizado en varias secciones. Puede encontrar un análisis detallado de todas estas secciones y sus entradas en la página [Man xorg.conf](#). Todos son configurados por el programa `XF86Setup`. Por ejemplo, la pantalla Monitor genera la sección Monitor en el archivo `xorg.conf`, la pantalla Mouse genera la sección Input Device para el ratón, etc. Una sección en el archivo comienza con la palabra clave **Section**, seguida por el nombre de sección entre comillas. La sección termina con el término **EndSection**. Los comentarios tienen un signo `#` al principio de la línea. A continuación se muestran diferentes tipos de secciones.

Sección	Descripción
Archivos	Directorios para fuente y archivos rgb
Módulo	Módulo de carga dinámica
ServerFlags	Varias opciones
Input Device	Configuración de ratón y teclado
Monitor	Configuración del monitor (configure las frecuencias horizontal y vertical)
Device	Configuración de tarjeta de video
Screen	Configuración de despliegue, configuración de pantalla virtual, colores de despliegue, tamaño de pantalla y otras características
ServerLayout	Especificación del diseño de pantallas y dispositivos de entrada

Las entradas de cada sección comienzan con datos de especificación, seguidos por una lista de valores. Con la versión 4.0, muchas especificaciones anteriores de datos se implementan mediante la entrada **Option**. Puede insertar la palabra clave **Option**, seguida por la especificación de datos y sus valores. Por ejemplo, ahora la especificación de diseño de teclado, `XkbLayout`, se implementa utilizando la entrada **Option** como se muestra aquí:

```
Option "XkbLayout" "us"
```

A pesar de que puede editar directamente el archivo al utilizar el editor de texto estándar, siempre es mejor depender de los programas de configuración como `xorgcfg` para hacer cambios. No necesita tocar casi ninguna de las secciones, pero en algunos casos, querrá hacer cambios a la sección Screen, ubicada al final del archivo. Para ello, necesita editar el archivo y agregarle o cambiar entradas en la sección Screen. En la sección Screen, puede configurar su despliegue de pantalla virtual y configurar el número de colores soportado. Debido a que es probable que sea la sección Screen la que más quiera cambiar, se analiza primero, aunque venga al final del archivo.

Screen

La sección Screen comienza con una entrada Identifier para dar nombre a la Pantalla. Después de la entrada Identifier, las entradas Device y Monitor especifican monitor y tarjeta de video en uso. El nombre dado en la entrada Identifier en estas secciones se usa para hacer referencia a esos componentes.

```
Section "Screen"
    Identifier "Screen0"
    Device "Videocard0"
    Monitor "Monitor0"
    DefaultDepth 24
    Subsection "Display"
```

```

Viewport    0 0
Depth       32
Modes       "1024x768" "1920x1200"
EndSubSection
EndSection

```

La sección Screen tiene subsecciones Display, una para cada resolución soportada. Aunque en la sección anterior configuró el hardware, en la subsección Display puede configurar las características de despliegue, como número de colores desplegado y tamaño virtual de la pantalla. Existen dos entradas principales: Depth y Modes. La entrada Depth es la resolución de pantalla: 8, 16 y 24. Puede agregar la entrada Default Depth para configurar la profundidad de color predeterminada, de acuerdo con la que soporte su servidor X: 8 para 256 K, 16 para 32 K y 24 para 16 M. Modes son los modos permitidos, de acuerdo con la resolución. También puede agregar a la entrada Virtual, para especificar el tamaño virtual de la pantalla. La pantalla virtual puede ser más grande que su área despliegue. Cuando mueve su ratón al borde de la pantalla de despliegue, se desplaza a un área escondida de la pantalla. De esta forma, puede tener una pantalla de trabajo más grande que el tamaño físico de su monitor. El tamaño de pantalla físico para un monitor de 17 pulgadas suele ser 1024 x 768. Puede establecerla en 1152 x 864, es decir, un tamaño de monitor de 21 pulgadas, con una entrada Virtual.

Cualquiera de estas características de esta sección puede ser cambiada de manera segura. En realidad, para cambiar el tamaño de pantalla virtual, debe modificar esta sección. No deben modificarse otras secciones del archivo **xorg.conf**, a menos que esté seguro de lo que hace.

En general, estas entradas se detectan automáticamente. Sin embargo, para algunas combinaciones de monitor y tarjetas de video, es probable que la resolución de pantalla se detecte mal, dejándole sólo resoluciones más bajas. Para arreglar esto, tal vez deba colocar una entrada Modes en la sección Screen Display, mostrando sus resoluciones posibles. Las versiones generadas automáticamente de **xorg.conf** no tienen entrada Modes. Como opción, puede instalar una versión proporcionada por un vendedor de su controlador X11, si está disponible, como las de Nvidia o ATI.

```
Modes "1024x768" "1980x1200"
```

Files, Modules y Serverflags

Usualmente, Files, Modules y Serverflags no se necesitan para una configuración automática sencilla. Configuraciones más complejas tal vez sí los necesiten. **Xorg -configure** generará entradas de sistema para ellos.

La sección de configuración muestra una lista de directorios diferentes para recursos que Xorg necesita. Por ejemplo, para especificar la ubicación de la lista donde se encuentran los datos de color RGB, una línea empieza con la especificación de datos **RgbPath**, seguida por el nombre de ruta del archivo de datos de color **rgb**. Las fuentes para X Window System se manejan con el servidor XFS, cuyos archivos de configuración se localizan en el directorio **X11/fs**. Como opción, pueden mostrarse fuentes específicas en una lista en la sección Files, utilizando la opción **FontPath**, la entrada **ModulePath** especifica el nombre de ruta de los módulos de directorio. Este directorio almacenará módulos para controladores específicos de video. Aquí se muestra un ejemplo de estas entradas:

```

RgbPath "/usr/share/X11/rgb"
ModulePath "/usr/lib/xorg/modules"

```

Una fuente de X Window System se designa con una entrada FontPath. Aquí se muestra un ejemplo de dicha entrada, utilizando una fuente localizada en el directorio /usr/share/fonts/X11 (Ubuntu).

```
FontPath "/usr/share/fonts/X11/75dpi"
```

Si no se especifican `FontPaths` ni se usa el servidor XFS, el servidor X vuelve a las rutas de fuente predeterminadas compiladas en el servidor X (consulte la página `Man xorg.conf` para conocer más detalles). La sección `Module` especifica módulos que habrán de cargarse en forma dinámica mientras la entrada `Load` carga un módulo, usado para guardar extensiones de servidor y módulos de fuente. Se trata de una característica introducida en versiones 4.0 que permiten a los componentes del servidor X extender la funcionalidad del servidor X para cargarse como módulos. Esta característica proporciona una actualización sencilla, permitiéndole actualizar módulos sin reemplazar el servidor X entero. Por ejemplo, el módulo `extmod` contiene extensiones para permitir funciones utilizadas comúnmente en el servidor X. En el siguiente ejemplo, se carga el módulo `extmod` conteniendo un conjunto de extensiones necesarias. Son de especial interés los módulos `dri`, `glx` y `GLcore`. Estos proporcionan soporte acelerado a tarjetas 3-D. Consulte la página `Man xorg.conf` para conocer más detalles.

```
Load "extmod"
Load "dri"
Load "glx"
Load "GLcore"
```

Se pueden configurar varias marcas para el servidor Xorg. Éstas ahora se implementan como opciones. (Encontrará una lista completa en la página `Man xorg.conf`.) Por ejemplo, el valor **`BlankTime`** especifica el tiempo de espera para el protector de pantallas. **`DontZap`** deshabilita el uso de CTRL-ALT-RETROCESO para apagar el servidor. **`DontZoom`** deshabilita el cambio entre modos gráficos. Puede crear una entrada `Option` con la marca como opción. En el siguiente ejemplo se configura la marca de servidor para el tiempo de espera del protector de pantalla:

```
Option "BlankTime" "30"
```

Input Device

Con la versión 4.0, la sección `Input Device` reemplazó secciones previas: `Keyboard`, `Pointer` y `XInput`. Para proporcionar soporte a un dispositivo de entrada, como un teclado, puede crear una sección `Input Device` para éste e insertar las entradas `Identifier` y `Driver` para el dispositivo. Por ejemplo, la siguiente entrada crea una sección `Input Device` para el teclado:

```
Section "Input Device"
    Identifier "Keyboard 0"
    Driver "kbd"
```

Cualquier característica se agrega como opción, sea diseño o modelo del teclado. Existe gran número de opciones para esta sección. Consulte las páginas `Man xorg.conf` para ver una lista completa. En el siguiente ejemplo se muestra una entrada de teclado completa con opciones de autorrepeticición, modelo del teclado (`XkbModel`) y diseño del teclado (**`XkbLayout`**) insertadas:

```
Section "Input Device"
    Identifier "Keyboard 0"
    Driver "kbd"
    Option "AutoRepeat" "500 5"
    Option "XkbModel" "pc105"
    Option "XkbLayout" "us"
EndSection
```


Puede crear una sección Input Device para su ratón y cualquier otro dispositivo de puntero. Sin embargo, los sistemas que utilizan escritorios como GNOME y KDE, la configuración del ratón se maneja directamente en el escritorio. No existe configuración Xorg.

La sección Mouse sólo tiene unas cuantas entradas, con algunas adecuadas para tipos específicos de ratón. Las características se definen utilizando entradas Option. La opción Protocol especifica el protocolo para los usos del ratón, como PS/2, Microsoft o Logitech. La opción Device es el nombre de ruta para el dispositivo de ratón. En el siguiente ejemplo se muestra una sección Pointer estándar, para un ratón de tres botones PS/2. El archivo del dispositivo es `/dev/mouse`.

```
Section "Input Device"
  Identifier "Mouse 1"
  Driver "mouse"
  Option "Protocol" "PS/2"
  Option "Device" "/dev/mouse"
  Option "Emulate3Buttons" "off"
EndSection
```

Monitor

Debe existir una sección Monitor para cada usado por su sistema. Las frecuencias horizontal y vertical deben ser acertadas o puede dañar su monitor. Una sección Monitor comienza con entradas identificando el monitor, como el nombre del vendedor y modelo. Las entradas HorizSync y VerRefresh están donde las frecuencias horizontal y vertical se especifican. Casi todos los monitores soportan varias resoluciones. Dichas resoluciones se especifican en la sección Monitor con las entradas ModeLine. Existe una entrada ModeLine para cada resolución. La entrada ModeLine tiene cinco valores: nombre de la resolución, valor de reloj de punto, después dos conjuntos de cuatro valores, uno para tiempo horizontal y otro para vertical, que terminan con marcas. Las marcas especifican características diferentes del modo, como Interlace, para indicar modo entrelazado y +hsync y +vsync para seleccionar la polaridad de la señal.

ModeLine "nombre" relojpuntofrec-horizontal frec-vertical marcas

Aquí se muestra un ejemplo de la entrada ModeLine. Lo mejor es dejar toda la sección Monitor sin modificación; dependa, en cambio, de entradas generadas por XF86Setup.

ModeLine "800x600" 50.00 800 856 976 1040 600 637 643 666 +hsync +vsync

Aquí se muestra una lista de entradas comunes para la sección Monitor:

Opción	Descripción
Identifier	Un nombre para identificar el monitor
VendorName	Fabricante
ModelName	Marca y modelo
HorizSync	Frecuencia de actualización horizontal; puede ser un rango o serie de valores
VerRefresh	Frecuencia de actualización vertical; puede ser un rango o serie de valores
Gamma	Corrección gamma
ModeLine	Especifica una resolución con reloj de punto, el tiempo horizontal y vertical para esa resolución

Aquí se muestra un ejemplo de la sección Monitor:

```
Section "Monitor"
    Identifier "Monitor0"
    VendorName "Dell 2405FPW (Analog)"
    ModelName "Unknown"
    HorizSync 30 - 83.0
    VertRefresh 56 - 76.0
    Option "dpms"
EndSection
```

Device

La sección Device especifica su tarjeta de video. Comienza con una entrada Identifier y otra entrada para el controlador de tarjeta de video. En el siguiente ejemplo se crea un Identifier para una tarjeta Nvidia, denominado "Videocard0" y después se especifica que el controlador nv (Nvidia) se utilizará para éste:

```
Identifier " Videocard0"
Driver "nv"
```

Las entradas siguientes identifican la tarjeta, como VendorName, BoardName y Chipset. La cantidad de RAM de video se indica en la entrada VideoRam. La entrada Clocks muestra una lista con valores de reloj. Muchas entradas diferentes pueden hacerse en esta sección, como Ramdac para un chip Ramdac, si la tarjeta tiene uno, y MemBase para la dirección base de un búfer de marco, de ser accesible. Consulte las páginas Man de **xorg.conf** para ver una lista y descripciones más detalladas.

Pese a que puede cambiar la entrada VideoRam de manera segura (por ejemplo, agregando más memoria a su tarjeta), no es seguro cambiar la entrada Clocks. Si coloca mal los valores del reloj, puede destruir fácilmente su monitor. Dependa de los valores de reloj generados por xorgcfg u otros programas de configuración de Xorg. Si faltan los valores de reloj, esto significa que el servidor los determinará automáticamente.

```
Section "Device"
    Identifier "Videocard0"
    Driver "nv"
EndSection
```

Dependiendo de su nivel de detección, puede generarse información más detallada:

```
Identifier "Card0"
Driver "nouveau"
VendorName "nVidia Corporation"
BoardName "NV43 [GeForce 6600]"
BusID "PCI:3:0:0"
```

ServerLayout

La sección ServerLayout permite especificar el diseño de sus pantallas y la selección de dispositivos de entrada. Las secciones ServerLayout incluyen opciones que suelen encontrarse en la sección ServerFlags. Puede configurar varias secciones ServerLayout y seleccionarlas desde la línea de comandos. En el siguiente ejemplo se muestra una sección ServerLayout simple para una configuración básica:

```
Section "ServerLayout"
    Identifier "single head configuration"
    Screen 0 "Screen0" 0 0
    InputDevice "Keyboard0" "CoreKeyboard"
EndSection
```

Varios monitores

Si tiene más de una tarjeta de video con un monitor conectado a cada una, entonces su servidor X detectará e implementará las tarjetas, cada una con entrada propia de dispositivo y monitor. Los monitores conectados a la misma tarjeta de video requieren una configuración más compleja. En efecto, tiene dos monitores utilizando el mismo dispositivo, la misma tarjeta de video. Con los controladores propiedad de Nvidia y ATI, puede usar sus herramientas de configuración para establecer secciones Monitors separadas. Estos controladores también soportan escritorios extendidos, donde un monitor puede desplegar una extensión de otro (TwinView en Nvidia).

Los servidores X estándar para casi todas las tarjetas de video soportan varios despliegues. La mayoría de distribuciones proporcionan herramientas de configuración de despliegue para configurar de manera sencilla despliegues separados. También puede implementar un escritorio extendido usando el servicio Xinerama del servidor X.

Para configurar dos monitores separados en una tarjeta Nvidia, las secciones Device, Screen y Monitor correspondientes se configuran para cada monitor, con la sección Screen conectando las secciones Monitor y Device. La sección ServerLayout muestra ambas pantallas. En el caso de un escritorio extendido, la opción TwinView se establece en la sección Device, con especificaciones para cada monitor. Revise el archivo readme de Nvidia para conocer más detalles de la instalación en Linux.

Las tarjetas ATI usan un formato muy parecido. La opción MergedFB implementa la opción de escritorio extendido del controlador ATI X y la opción DesktopSetup se utiliza para el controlador propiedad de ATI.

Argumentos de línea de comandos de X Window System

Cualquier aplicación X Window System puede iniciarse en una secuencia de comandos **.xinitrc**, **.xsession** o la línea de comandos en una ventana Xterm. Algunas distribuciones, incluidas Mandrake y Red Hat, permiten a los usuarios colocar aplicaciones de inicio de X Window System en un archivo **.Xclients** que se lee con la secuencia de comandos **.xinitrc**. Casi todas las aplicaciones de X Window System toman un conjunto de argumentos estándar usados para configurar la ventana y desplegar los usos de aplicación. Puede configurar el color de las barras de ventana, darle un título específico a la ventana, especificar el color y fuente para el texto, además de colocar la ventana en una ubicación específica de la pantalla. En la tabla 7-3 se muestra una lista de argumentos de X Window System. Se analizan con más detalle en las páginas X Man, **man X**.

Un argumento de uso común es **-geometry**. Toma un argumento adicional especificando la ubicación donde quiere que una ventana de aplicación se despliegue en la pantalla. En el siguiente ejemplo, la aplicación **xclock** de X Window System se llama con un argumento **-geometry**. Un conjunto de hasta cuatro números especifican la posición. El valor **+0+0** hace referencia a System. El valor **-0-0** indica la esquina superior derecha.

```
& xclock -geometry +0+0 &
```

Argumentos de configuración de aplicación X Window	Descripción
-bw <i>num</i>	Ancho del borde de los pixeles en un marco
-bd <i>color</i>	Color de borde
-fg <i>color</i>	Color de primer plano (para texto o gráficos)
-bg <i>color</i>	Color de segundo plano
-display <i>nombre-despliegue</i>	Despliega el cliente donde se ejecutará; despliega el nombre que incluye nombre host, número de despliegue y número de pantalla (consulte las páginas Man de X)
-fn <i>fuentes</i>	Fuente para utilizar para el despliegue de texto
-geometry <i>desplazamientos</i>	Ubicación en pantalla donde se colocará la ventana de aplicación de X Window System; los desplazamientos se miden en relación con el despliegue de la pantalla
-iconic	Inicia la aplicación con iconos, no con una ventana abierta
-rv	Cambia el color del segundo y primer planos
-title <i>cadena</i>	Título de la barra de título de la ventana
-name <i>cadena</i>	Nombre de la aplicación
-xrm <i>cadena-recursos</i>	Especifica valores de recursos

TABLE 7-3 Opciones de configuración para aplicaciones basadas en X Window System

Con la opción **-title**, puede configurar el título desplegado en la ventana de aplicación. Observe el uso de comillas para títulos con más de una palabra. Puede configurar la fuente con el argumento **-fn**, el color de texto y gráficos con el argumento **-fg**. **-bg** configura el color del segundo plano. En el siguiente ejemplo se inicia la ventana Xterm con el título “Mi nueva Ventana” en la barra de título. El color del texto y gráficos es verde, cuando color de segundo plano es gris. La fuente es Helvetica.

```
$ xterm -title "My New Window" -fg green -bg gray -fn /usr/fonts/helvetica &
```

Comandos y archivos de configuración de X Window System

X Windows System utiliza varios archivos de configuración, además de comandos X para configurar X Windows System. Algunos archivos de configuración pertenecen al sistema y no deben modificarse. Sin embargo, cada usuario tiene su propio conjunto de archivos de configuración, por ejemplo **.xinitrc**, **.xsession** y **.Xresources**, que pueden utilizarse para configurar una interfaz de X Windows System personalizada. El directorio **fs** almacena el archivo de configuración para las fuentes de X Windows System. Un archivo **.Xclients** puede almacenar aplicaciones de inicios de X Windows System. Estos archivos de configuración se leen y ejecutan automáticamente cuando X Windows System inicia con el comando **startx** o un administrador de despliegue X, como XDM o GDM. Con estos archivos de configuración, se ejecutan comando X utilizados para definir su sistema. Con comandos como **xset** y **xsetroot**, agrega fuentes o controla el despliegue de su ventana raíz. Más adelante en este capítulo, en la tabla 7-4 se presenta una lista de comandos y archivos de configuración de X Windows System.

Comando de X Window System	Explicaciones
xterm	Abre una nueva ventana de terminal
xset	Configura las opciones de X Windows System; consulte las páginas Man para conocer una lista completa -b Configura el timbre -c Configura los clics de teclas +fp listadefuentes Agrega fuentes -fp listadefuentes Quita fuentes led Enciende o apaga los LED del teclado m Configura el ratón p Configura los valores de color de pixel s Configura el protector de pantalla q Muestra una lista de la configuración actual
xsetroot	Configura la ventana raíz -cursor <i>archivo de cursor</i> arachivomáscara Asigna al puntero imágenes de mapa de bits cuando está fuera de cualquier ventana -bitmap filename Configura el patrón de la ventana raíz a un mapa de bits. -gray Configura el color del segundo plano en gris -fg color Configura el color del mapa de bits del primer plano -bg color Configura el color del mapa de bits del segundo plano -solid color Configura el color del segundo plano -name string Asigna a la ventana raíz el nombre de una <i>cadena</i>
xmodmap	Configura la entrada de dispositivos; lee el archivo .Xmodmap -pk Despliega el mapa de teclas actual -e expresión Configura la combinación de teclas keycode NUMBER = KEYSYMNAME Asigna a la tecla un símbolo de tecla específico keysym KEYSYMNAME = KEYSYMNAME Configura la tecla para operar de la misma forma que la tecla especificada pointer = NUMBER Configura los códigos de botón del ratón
xrdb	Configura los recursos de X Windows System; lee el archivo .Xresources
xdm	X Window Administrador de despliegue X Windows System; ejecuta el servidor Xorg para su sistema, usualmente llamado por xinitrc
startx	Inicia X Windows System al ejecutar xinit y lo instruye para leer el archivo xinitrc
xfs <i>archivo-config</i>	El servidor de fuentes de X Windows System
mkfontdir <i>font-directory</i>	Indiza nuevas fuentes, haciéndolas accesibles para el servidor de fuentes
xlsfonts	Muestra una lista de fuentes en su sistema
xfontsel	Despliega las fuentes instaladas
xdpyinfo	Muestra una lista detallada de información acerca de su configuración de X Windows System
xinit	Inicia X Windows System, pero antes lee el archivo de sistema xinitrc ; cuando se invoca desde startx , también lee el archivo de usuario .Xclients ; xinit no se llama directamente, sino a través de startx
xmkmf	Crea un archivo para una aplicación de X Windows System, al utilizar Imakefile ; de la aplicación; invoca imake para generar un archivo de creación (nunca invoque imake directamente)
xauth	Lee el archivo .Xauthority para configurar el control de acceso a una cuenta de usuario a través de XDM para sistemas remotos

TABLA 7-4 Comandos de X Window System

Obtendrá una descripción completa de su configuración X actual usando el comando **xdpyinfo**. Las páginas Man de X proporcionan una introducción detallada a comandos y archivos de la configuración de X.

Fuentes XFS

Casi todas las fuentes se manejan ahora directamente en escritorios al utilizar **fontconfig** (GNOME o KDE, **fonts:/**), muy fáciles de instalar. Estas fuentes se almacenan en el directorio **.fonts** del usuario o en **/usr/share/fonts**. Además, tiene soporte de fuentes separado para su X Windows System, que toma fuentes en **/usr/share/X11/fonts**. El servidor de fuentes XFS administra las de X Windows System, definido según el archivo de configuración **/etc/X11/fs/config**. Este archivo muestra listas de fuentes en la entrada **catalogue**. Las páginas X Man proporcionan un análisis detallado acerca de las fuentes. Las fuentes pueden cargarse de manera manual con el comando **xfx**. Antes de acceder a fuentes instaladas recientemente, primero tiene que indizarlas con el comando **mkfontdir**. Para que las fuentes se carguen automáticamente, agregue el directorio con el nombre de ruta completo a la entrada **catalogue** en el archivo de configuración XFS.

NOTA Algunas distribuciones recientes están abandonando el servidor de fuentes XFS para dejar sólo unas cuantas fuentes específicas instaladas en un directorio designado X11.

Recursos de X

Varios comandos de X, como **xrdb** y **xmodmap**, configuran su interfaz de X Windows System. Las configuraciones gráficas de X Windows System se muestran en una lista de un archivo de recursos denominado **.Xresources**. Cada usuario puede tener un archivo **.Xresources** personalizado en su directorio home, configurando X Windows System en especificaciones particulares. El archivo **.Xresources** contiene entradas para configurar programas específicos, como el color de ciertos widgets. También existe una versión para todo el sistema denominada **/etc/X11/Xresources**. El archivo **.Xdefaults** es un archivo de configuración predeterminado, cargado por todos los programas, conteniendo el mismo tipo de entradas para la configuración de recursos que **.Xresources**. Los programas de su sistema pueden acceder un archivo **.Xdefaults**, pero no los que están fuera de su sistema. El directorio **/usr/share/X11/app-defaults** almacena archivos conteniendo configuraciones de recursos predeterminados para aplicaciones particulares de X, como **Xterm**, **Xclock** y **Xmixer**. El archivo **Xterm** almacena entradas de recursos que especifican el despliegue de una ventana Xterm. Es posible sobrescribir cualquiera de estas opciones predeterminadas con entradas alternas en un archivo **.Xresources** de su directorio de inicio. También, crear un archivo **.Xresources** propio en su directorio de inicio y agregar entradas de recursos a éste. Por igual, copiar el archivo **/etc/X11/Xresources** y editar las entradas ahí o agregar entradas propias nuevas.

La configuración se lleva a cabo con el comando **xrdb**, que lee el archivo **.Xresources** del sistema y cualquier archivo **.Xresources** o **.Xdefaults** en su directorio de inicio. El comando **xrdb** se ejecuta actualmente en la secuencia de comandos **/etc/X11/xinit/xinitrc** y **/etc/X11/xdm/Xsession**. Si crea su propia secuencia de comandos **.xinitrc** en su directorio de inicio, asegúrese de ejecutar el comando **xrdb** con al menos su archivo **.Xresources** o **/etc/X11/Xresources** (es preferible ejecutar ambos). Puede asegurar esto usando sólo una copia de su secuencia de comandos de sistema **xinitrc**, como su archivo **.xinitrc** y después modificarla como quiera. Consulte las páginas Man en **xrdb** para conocer más detalles acerca de los recursos. También puede encontrar una discusión detallada de Xresources, además de otros comandos X en las páginas Man para X.

Una entrada en el archivo **.Xresources** consta de un valor asignado a recursos, clase o grupo de recursos para una aplicación. Usualmente, los recursos se emplean para widgets o clases de widgets en una aplicación. La designación de recursos suele constar de tres elementos separados por puntos: aplicación, objeto en la aplicación y recurso. La designación completa está determinada por dos puntos, después el valor. Por ejemplo, suponga que quiere cambiar el color de la manecilla del reloj a azul en la aplicación `oclock`. La aplicación es `oclock`, el objeto es `clock` y el recurso `hour`: `oclock.clock.hour`. Esta entrada se ve así:

```
oclock.clock.hour: blue
```

El elemento de objeto es en realidad una lista de objetos denotando la jerarquía que guía a un objeto determinado. En el ejemplo de `oclock`, sólo existe un objeto, pero en muchas aplicaciones, la jerarquía del objeto puede ser compleja. Esto requiere un extenso conjunto de objetos en una lista para especificar el que quiere. Para evitar esta complejidad, puede utilizar la notación asterisco para referirse directamente al objeto que quiere utilizando un asterisco en lugar de un punto. Sólo necesita saber el nombre del recurso que pretende cambiar. En el siguiente ejemplo se configuran las manecillas del reloj de horas y minutos en verde:

```
oclock*hour: green
oclock*minute: green
```

También puede utilizar el asterisco para aplicar un valor a todas las clases de objeto. Muchos recursos individuales se agrupan en clases. Puede hacer referencia a todos los recursos de una clase por su nombre de clase, mismo que comienza con un carácter en mayúscula. En la aplicación `Xterm`, por ejemplo, los recursos de color del segundo plano y cursor son parte de la clase `Background`. La referencia **`XTerm*Background`** cambia todos estos recursos en una ventana `Xterm`. Sin embargo, las referencias específicas siempre sobrepasan las más generales.

También puede utilizar el asterisco para cambiar valores de un recurso en objetos de todas sus aplicaciones. En este caso, puede colocar un asterisco antes del recurso. Por ejemplo, para cambiar el color del primer plano a rojo en todos los objetos en cada aplicación, ingrese:

```
*foreground: red
```

Si quiere cambiar el color del primer plano de las barras de desplazamiento en todas sus aplicaciones, utilice:

```
*scrollbar*foreground: blue
```

El comando **`showrgb`** muestra una lista de colores diferentes disponibles en su sistema. Puede utilizar el nombre descriptivo o una forma hexadecimal. Los valores también pueden ser fuentes, mapas de bits y mapas de píxeles. Puede cambiar la fuente desplegada por ciertos objetos en aplicaciones gráficas, además de cambiar las imágenes del segundo plano o borde. Los recursos varían con cada aplicación. Las aplicaciones toleran diferentes tipos de objetos y recursos. Consulte las páginas y documentación Man para saber qué recursos soporta la aplicación y valores aceptados por ésta. Algunos recursos toman valores booleanos para activar o desactivar características, mientras otros especifican opciones. Algunas aplicaciones tienen un conjunto de valores de recursos predeterminado colocados automáticamente en sus archivos **`.Xresources`** o **`.Xdefaults`** del sistema.

El archivo **`.Xmodmap`** almacena configuraciones para sus dispositivos de entrada, como ratón y teclado (por ejemplo, podría unir teclas como `RETROCESO` o invertir operaciones de clic entre

botones derecho e izquierdo del ratón). El archivo **.Xmodmap** usado por su administrador de despliegue está en el directorio de configuración del administrador de despliegue, como **/etc/X11/xdm**, aunque el empleado por **startx** se ubica en **/etc/X11/xinit**. Cada usuario crea su archivo **.Xmodmap** personalizado en el directorio de inicio para configurar dispositivos de entrada del sistema. Esto es útil si los usuarios conectan sus propias terminales a su sistema Linux. El comando **xmodmap** lee el archivo **.Xmodmap** y realiza la configuración. El comando busca primero este archivo en el directorio de inicio del usuario y lo utiliza. Si no existe **.Xmodmap** en el directorio de inicio, usa el de su administrador de despliegue o el comando **startx**. Puede ver entradas para el comando **xmodmap** en el archivo **/etc/X11/xinit/xinitrc** y el archivo **Xsession** del administrador de despliegue. Si tiene su propia secuencia de comandos **.xinitrc** o **.xsession** en su directorio de inicio, éste ejecutará el comando **xmodmap**, ya sea con su propio archivo **.Xmodmap** o el archivo **Xmodmap** del sistema. Consulte las páginas Man en **xmodmap** para conocer más información.

Comandos de X

Generalmente, una secuencia de comandos **.xinitrc** o **.xsession** tiene comandos propios para X Window System, como **xset** y **xsetroot**, usados para configurar diferentes características en su sesión de X Window System. El comando **xset** configura diferentes opciones, como activar el protector de pantalla o configurar el volumen para diferentes sonidos. Utilice **xset** para cargar fuentes. Consulte las páginas Man de **xset** para conocer detalles más específicos. Con la opción **b** y el argumento **on** u **off**, **xset** activa o desactiva la bocina. En el siguiente ejemplo se desactiva ésta:

```
xset b on
```

Puede utilizar **xset** con la opción **-s** para configurar el protector de pantalla. Con los argumentos **on** y **off**, puede activar o desactivar el protector de pantalla. Dos números insertados como argumentos especifican tiempo de espera y periodo en segundos. El tiempo de espera son los segundos que esperará el protector de pantalla antes de activarse y el periodo es el lapso comprendido antes de regenerar el patrón.

El comando **xsetroot** permite configurar características de su ventana raíz (configurando color o desplegando un patrón de mapa de bits, puede incluso usar un cursor de su propio diseño). En la tabla 7-5 se muestra una lista de opciones diferentes de **xsetroot**. Consulte las páginas Man de **xsetroot** para conocer opciones y detalles. El siguiente comando **xsetroot** utiliza la opción **-solid**, para configurar el color de segundo plano de la ventana raíz en azul:

```
xsetroot -solid blue
```

En la tabla 7-4 se muestra una lista de comandos comunes de X Windows System, y en la tabla 7-5 una lista de archivos de configuración y directorios asociados con X Windows System.

Administradores de despliegue: XDM, GDM y KDM

Un administrador de despliegue inicia automáticamente X Windows System cuando enciende su computadora, presentando una ventana de inicio de sesión y un menú para seleccionar administrador de ventanas o escritorio que quiere utilizar. Las opciones de apagado de su sistema también están aquí. Actualmente, puede utilizar tres administradores de despliegue. K Display Manager (KDM) administrador proporcionado por KDE. GNOME Display Manager (GDM) integrado en GNOME. XDM el administrador original y rara vez se utiliza directamente en sistemas Linux.

Archivos de configuración	Explicación
.Xmodmap	Archivo de configuración de dispositivos de entrada de X Windows System del usuario
.Xresources	Archivo de configuración de recursos de X Windows System del usuario
.Xdefaults	Archivo de configuración de recursos de X Windows System del usuario
.xinitrc	El archivo de configuración de X Windows System del usuario se lee automáticamente (por xinit , si existe)
.Xclients o .Xsessions	Archivo de configuración de X System del usuario
.Xauthority	Controles de acceso del usuario a través de la interfaz de inicio de sesión GUI XDM
/etc/X11/	Directorio almacenando el archivo de configuración y subdirectorios de X Window System release 6
/etc/X11/fs	Sistema de directorio de configuración de fuentes de X Window System.
/etc/X11/xinit/xinitrc	Sistema de archivo de inicialización de X Window System; xinit lo lee automáticamente
/etc/X11/xinit/Xclients	Sistema de archivo de configuración de X Window System
/etc/X11/Xresources	Sistema de archivo de recursos de X Window System
/etc/X11/Xmodmap	Sistema de archivo de dispositivos de entrada de X Window System
/usr/share/X11/rgb.txt	Colores de X Window System. Cada entrada tiene cuatro campos: los primeros tres tienen números para red, green y blue: el último campo es el nombre dado al color
/usr/share/X11	Directorio administrado por el sistema X Window System para almacenamiento de fuentes y configuración de aplicación

TABLA 7-5 Archivos de configuración y directorios de X Window System

Cuando se inicia un sistema configurado para ejecutar un administrador de sistema, X Window System inicia de inmediato y despliega un cuadro de diálogo de inicio de sesión, pidiendo al usuario ingrese su nombre y contraseña de inicio de sesión. Una vez introducidos, inicia la interfaz elegida de X Window System (digamos, con GNOME, KDE u otro escritorio o administrador de ventanas). Cuando el usuario sale del administrador de ventanas o escritorio, el sistema regresa al cuadro de diálogo de inicio de sesión y permanece allí hasta que otro usuario inicia sesión. Puede cambiar a una interfaz de línea de comandos, pulsando las teclas CTRL-ALT-F1 y regresar al cuadro de diálogo de inicio de sesión de su administrador de despliegue con CTRL-ALT-F7. Para detener el servidor X completamente, detenga el administrador de despliegue, **/etc/init.d/gdm stop**.

También utilice el administrador de despliegue para acceder al control de diferentes host y usuarios en su red. El archivo **.Xauthority** de cada directorio de inicio del usuario, contiene información de autenticación para ese usuario. Un administrador de despliegue como XDM, soporta el X Display Manager Control Protocol (XDMCP). Fueron diseñados originalmente para sistemas tales como estaciones de trabajo en operación continua, pero también se usan para iniciar X Windows System automáticamente en cada sistema de un solo usuario cuando el sistema inicia.

NOTA Casi todas las distribuciones instalarán KDM o GDM. Las distribuciones que favorecen a KDE instalarán KDM, mientras aquellas que incluyen GNOME y KDE, instalarán GDM.

Un administrador de sistema se ejecuta automáticamente cuando su sistema inicia a nivel gráfico. En muchas distribuciones este nivel de ejecución es 5. Su sistema puede ejecutarse en diferentes niveles de ejecución; por ejemplo, el nivel multiusuario estándar, de usuario sin red y administración de sistema. El nivel de ejecución gráfico es el mismo que de multiusuario estándar, excepto porque inicia automáticamente X Window System en máquinas conectadas y activa la pantalla del administrador de inicio de sesión.

En casi todas las instalaciones de distribución, su sistema se configura para iniciar automáticamente en un nivel de ejecución gráfico (el número 5, en la mayoría de las distribuciones), activando el administrador de despliegue. Si en vez de eso, inicia con la petición de inicio de sesión en modo de línea (multiusuario estándar), puede cambiar manualmente el administrador de despliegue modificando su número de ejecución al número de nivel de ejecución gráfico. Para hacer esto temporalmente, puede especificar su nivel de ejecución con la utilería de administración **telinit**. El siguiente comando cambia del nivel de ejecución multiusuario estándar (3 en muchas distribuciones), a la línea de comandos:

```
telinit 3
```

Este comando cambiará el nivel de ejecución gráfico (inicio de sesión gráfico), al inicio de sesión gráfico:

```
telinit 5
```

Para que un nivel de ejecución sea el predeterminado, debe editar el archivo **/etc/inittab**.

Xsession

Un administrador de despliegue se refiere a un inicio de sesión del usuario e inicio de un administrador de ventanas, además del escritorio como una sesión. Cuando el usuario sale del escritorio y cierra su sesión, esta termina. Cuando otro usuario inicia sesión, una nueva comienza. X Window System nunca se apaga; sólo el escritorio o programas del administrador de ventanas se apagan. Los menús de sesión son una ventana del administrador de inicio de sesión mostrando diferentes tipos de sesiones que puede iniciar —en otras palabras, diferentes tipos de administradores de ventana o escritorios. Para cada sesión, la secuencia de comandos **Xsession** es la utilizada para configurar un despliegue de X Window System del usuario y ejecutar el escritorio o administrador de ventanas seleccionado.

Xsession es una secuencia de comandos de inicio del administrador de despliegue de sesión usado por GDM, además de KDM y XDM. Contiene muchos de los comandos X, también utilizados en la secuencia de comandos de inicio **xinitrc** utilizada por **startx**. Los comandos de ejecución común para todos los administradores de despliegue y escritorios se almacenan en la secuencia de comandos **xinitrc-common**, que **.Xmodmap** ejecuta primero. La secuencia de comandos **xinitrc-common** ejecuta los comandos **xmodmap** y **xrdb** mediante los archivos **.Xmodmap** y **.Xresources** en el directorio **/etc/X11/xinit**. **Xsession** guarda cualquier error en el archivo de usuario **.xsession-errors** en el directorio de inicio. **Xsession** también leerá cualquier secuencia de comandos de shell ubicada en el directorio **/etc/X11/xinit/xinitrc.d**. En la actualidad, esto almacena una secuencia de comandos de entrada para detectar el tipo de lenguaje del teclado, además de secuencias de comandos para configuraciones de teclado adicionales, como el servicio **xdg-user-dir** implementado por algunas distribuciones.

```
Xsession gnome
```

Estos entornos se muestran en una lista en la secuencia de comandos **Xsession** con la instrucción `case`. Aquí encontrará entradas para GNOME, KDE y el administrador de ventanas `twm simple`. GNOME se invoca directamente con el comando `gnome-session` y KDE con el comando `startkde`. Si **Xsession** no se invoca con un entorno específico, se busca en el directorio de inicio del usuario una secuencia de comandos `.Xsession` y `.Xclients`. Si esas secuencias faltan, se utiliza la del sistema `Xclients, /etc/X11/xinit/Xclients`. `Xclients` revisará si GNOME o KDE está instalado e iniciará el que se encuentre instalado. Si ninguno está, utiliza el viejo administrador de ventanas `twm`.

Si los usuarios quieren configurar sus propios archivos de inicio, pueden copiar el archivo **Xsession** a su directorio de inicio, asignarle el nombre `.xsession` y después editarlo. En el siguiente ejemplo se muestra una secuencia de comandos **Xsession** para ejecutar la secuencia del usuario `.xsession`, si existe. Se espera que esta última inicie un administrador de ventanas o escritorio. El siguiente ejemplo está tomado del código de la secuencia de comandos `Xclients`, que inicia un administrador de ventanas `simple twm`, para abrir una ventana de terminal.

```
#
# Xsession

startup=$HOME/.xsession
resources=$HOME/.Xresources

if [ -f "$startup" ]; then
    exec "$startup"
else
    if [ -f "resources" ]; then
        xrdp -load "$resources"
    fi

    if [ -x /usr/bin/xterm ] ; then
        /usr/bin/xterm -geometry 80x50-50+150 &
    fi
    if [ -x /usr/bin/twm ] ;then
        exec /usr/bin/twm
    fi
fi
```

NOTA Como una mejora a **startx** o administrador de despliegue, utilice el *X session manager (xsm)*. Úselo para ejecutar X Window System con diferentes sesiones. Una sesión es un grupo especificado de aplicaciones para X. Iniciar con una sesión puede activar GNOME y Mozilla, mientras iniciar con otro puede comenzar KDE y KOffice. Puede guardar su sesión mientras la utiliza o cuando apaga su sistema. Las aplicaciones en ejecución se vuelven parte de una sesión guardada. Cuando inicia, *xsm* despliega un menú de sesión donde seleccionar de una lista con sesiones previas guardadas.

X Display Manager (XDM)

XDM administra una serie de despliegues de X, ya sea en su sistema local o servidores remotos. El diseño de XDM se basa en el estándar X Consortium XDMCP. El programa XDM modera inicios de sesión de usuario, proveyendo autenticación e inicios de sesión. Para inicios de sesión basados en caracteres, una sesión es el periodo de actividad de un usuario desde su ingreso al sistema a través de la shell por la línea de comandos.

En el caso de XDM y otros administradores de despliegue, la sesión está determinada por el administrador, que suele ser la duración de un administrador de ventanas o escritorio. Cuando el escritorio o administrador de ventanas termina, también la sesión.

El programa XDM despliega una ventana de inicio de sesión con cuadros para nombre y contraseña. El usuario inicia sesión y un administrador de ventanas o escritorio inicia. Cuando el usuario sale del administrador de ventanas, X Window System reinicia automáticamente, desplegando la ventana de inicio de sesión nuevamente. Las autenticaciones para controlar el acceso para usuarios específicos se almacenan en el archivo **.Xauthority**.

Los archivos de configuración de XDM se ubican en el directorio **/etc/X11/xdm/**. El archivo de configuración principal de XDM es **xdm-config**. Los archivos como **Xresources** configuran la manera en que se despliega el cuadro de diálogo y **Xsetup** especifica una imagen de ventana raíz u otras ventanas para despliegue. Cuando el usuario inicia una sesión, la secuencia de comandos **Xsession** se ejecuta para configurar X Window System del usuario y ejecutar el administrador de ventanas o escritorio del usuario. La secuencia de comandos suele llamar a la secuencia de comandos **.xsession** en el directorio de inicio del usuario, si existe uno. Almacena cualquier comando X del usuario.

Si quiere iniciar XDM desde la interfaz de línea de comandos, puede insertar el comando **xdm** con la opción **-nodaemon**. CTRL-C después apaga XDM:

```
xdm -nodaemon
```

En la tabla 7-6 se muestra una lista de archivos de configuración y directorios asociados con XDM. **xdm-errors** contendrá mensajes de error de XDM y las secuencias de comandos que ejecuta, como **Xsession** y **Xstartup**. Revise este archivo si está teniendo problemas con XDM.

GNOME Display Manager

GDM administra el inicio de sesión del usuario y sesiones GUI. GDM da servicio a varios despliegues y genera un proceso para cada uno. El proceso principal de GDM escucha peticiones de XDMCP desde un despliegue remoto y monitorea las sesiones de despliegue local. GDM muestra una ventana de inicio de sesión con cuadros para insertar nombre y contraseña, también despliega

Nombres de archivo	Descripción
/etc/X11/xdm	Directorio de configuración XDM
xdm-config	Archivo de configuración XDM
Xsession	Secuencia de comandos de inicio para sesión de usuario
Xresource	Características de recursos para la ventana de inicio de sesión XDM
Xsetup	Configura la ventana de inicio de sesión y la pantalla de inicio de sesión XDM
Xstartup	Secuencia de comandos de inicio de la sesión
xdm-errors	Errores de las sesiones XDM
.xsession	Secuencia de comandos de sesión del usuario en el directorio de inicio; usualmente ejecutado por Xsession
Xreset	Restablece X Window System después de terminar la sesión
.Xauthority	Archivo de autorización de usuario donde XDM almacena claves de clientes para lectura

TABLA 7-6 Archivos de configuración y directorios de XDM

entradas para submenús de sesiones y apagado. El menú de sesiones presenta diferentes administradores de ventana y escritorios para iniciar, como GNOME o KDE.

Cuando GDM inicia, muestra la ventana de inicio de sesión con un recuadro para tal efecto. Están disponibles varios temas de GDM, que puede seleccionar utilizando la herramienta de configuración de GDM. Tres menús desplegables se localizan al centro de la pantalla, etiquetados Idioma, Opciones y Apagar. Para iniciar sesión, escriba su nombre de usuario en el cuadro de entrada etiquetado Nombre de usuario y oprima ENTER. Como opción predeterminada, el escritorio GNOME inicia inmediatamente después.

Al cerrar su sesión desde el escritorio, vuelve a la ventana de inicio de sesión GDM. Para apagar su sistema Linux, haga clic en el botón Apagar. Para reiniciar, seleccione Reiniciar desde el menú Opciones. De manera alterna, también puede apagar desde GNOME. En el menú Sistema, seleccione la entrada Apagar. GNOME desplegará un cuadro de diálogo con los botones Suspend, Apagar o Reiniciar. Apagar es la opción predeterminada y ocurrirá automáticamente luego de unos segundos. Si selecciona Reiniciar, apagará y reiniciará su sistema.

Desde el menú Opciones, puede seleccionar escritorio o administrador de ventanas que quiera iniciar. Por ejemplo, aquí puede seleccionar KDE para iniciar K Desktop, en vez de GNOME. En Fedora, KDE y GNOME utilizarán temas similares, mostrándose con apariencia muy similar. El menú Idioma muestra una lista de idiomas diferentes que Linux soporta. Seleccione uno para cambiar la interfaz de lenguaje.

Configuración de GDM: **gdmsetup**

Si quiere cambiar la pantalla de inicio de sesión de GDM, **gdmsetup**. A menudo puede accederse desde el menú Sistema de GNOME y etiquetado como Ventana de entrada. Con **gdmsetup** puede configurar imagen de fondo, iconos desplegados, tema a usarse, usuarios en lista e incluso mensaje de bienvenida. Las ventanas de inicio de sesión pueden configurarse para usuarios locales o remotos. Puede seleccionar entre pantalla simple, pantalla simple con un explorador de rostro o pantalla con un tema. El panel local le permite seleccionar qué pantalla usar para inicios de sesión locales, además de explorar los temas disponibles. Desde el panel remoto puede seleccionar simple, simple con buscador o usar la misma configuración que su inicio de sesión local.

En el panel Usuarios, seleccione qué usuarios quiere desplegar cuando use un explorador de rostro. En el panel local, puede seleccionar entre varios temas. También puede hacer que se seleccione un tema de manera aleatoria.

En el panel de seguridad, puede configurar un inicio de sesión automático, omitiendo la pantalla de inicio de sesión al arranque. Incluso configurar un inicio de sesión por tiempo: iniciar sesión automáticamente en un usuario específico, tras haber pasado un límite de tiempo dado luego de que se desplegó la pantalla de inicio de sesión. En el segmento Seguridad del panel, configure las opciones de seguridad: permisos para inicio de sesión root o acceso TCP (Internet), además de configurar el número de inicios de sesión permitidos. Haga clic en el botón Configurar Servidor X en este panel para abrir una ventana que configure el acceso al servidor X. Consulte el manual de referencia de GNOME Display Manager, accesible desde el botón Ayuda, para conocer más detalles.

Archivos de configuración de GDM

Los archivos de configuración de GDM se localizan en los directorios **/etc/gdm** y **/usr/share/gdm**. Utiliza dos archivos de configuración donde se determinan varias opciones, como la imagen del logotipo y el texto de bienvenida que habrá de desplegarse. No se debe editar **defaults.conf** en **/usr/share/gdm**. Se sobrescribirá en una actualización GDM y se ignora si configura el archivo **custom.conf**. El archivo **custom** está vacío, aunque contiene comentarios detallados.

El directorio **/etc/gdm** contiene cinco subdirectorios: **init**, **modules**, **Postlogin**, **PostSession** y **PreSession**. La configuración de GDM es sencilla colocando o editando archivos en estos

directorios. El directorio **Init** contiene secuencias de comandos a ejecutarse cuando GDM inicia. Este directorio aloja una secuencia de comandos **Default** que almacena comandos de X, como configuración de segundo plano. Éstos aplican para la pantalla, mostrando la ventana de inicio de sesión GDM. El directorio **modules** almacena configuraciones de teclado y ratón para un acceso alterno y mejorado, como el ampliador del escritorio.

El directorio **PreSession** almacena cualquier comando de presesión para ser ejecutado, mientras el directorio **PostSession** almacena secuencias de comandos para comandos que quiere se ejecuten cuando su sesión termine. Ambos tienen la secuencia de comandos **Default**. Ninguna de las secuencias de comandos **Init**, **PreSession** o **PostSession** es necesaria. El directorio **PostLogin** almacena secuencias de comandos que se ejecutarán después del inicio de sesión, aunque antes de iniciar la sesión de X Window System. Se ofrece un ejemplo de secuencia de comandos.

En el caso de GDM, un programa llamado *el receptor* genera la ventana de inicio de sesión. Al principio, éste busca iconos para cada usuario en el sistema, localizados en el archivo **.gnome/photo**, en los directorios de inicio de los usuarios. Al hacer clic en el icono se despliega automáticamente el nombre del usuario, en el cuadro de inicio de sesión. Luego el usuario puede insertar la contraseña y hacer clic en el botón Login para iniciar sesión.

En la tabla 7-7 se muestran los archivos de configuración y directorios asociados con GDM.

K Display Manager (KDM)

K Display Manager (KDM) también administra inicios de sesión del usuario e inicia sesiones de X Window System. KDM se deriva de XDM, recurriendo a los mismos archivos de configuración. La ventana de inicio de sesión de KDM despliega una lista de iconos para usuarios en el sistema. Un usuario puede hacer clic en su propio icono y ese nombre del usuario aparece después en el cuadro de inicio de sesión. Inserte la contraseña y haga clic en Ir a para iniciar sesión. Sesión es un menú desplegable mostrando posibles sesiones. Haga clic en el botón Apagar para cerrar el sistema.

Configure KDM al utilizar el Administrador de configuración de KDM alojado en el escritorio de usuario root de KDE. Existen paneles para la configuración del fondo, logotipo y mensajes de bienvenida, también para añadir iconos para usuarios en el sistema. Puede agregar una nueva entrada de sesión en el menú Sesión, inserte el nombre de la entrada en el cuadro Nuevo tipo del panel Sesiones y haga clic en Agregar.

KDM usa los mismos archivos de configuración que se ubican en **/etc/kde/kdm**. Muchas de las secuencias de comandos son vínculos a archivos en el directorio de configuración XDM, **/etc/X11/xdm**. Estas incluyen vínculos a **Xsession**, **Xresources** y **Xsetup** de XDM, entre otros. KDM maneja

Directorio o nombre de archivo	Descripción
/etc/gdm	Directorio de configuración de GDM
/usr/share/gdm	Directorio de configuración de GDM para opciones predeterminadas y temas
defaults.conf	Archivo de configuración predeterminado de GDM /usr/share/gdm
custom.conf	Archivo de configuración personalizado de GDM, /etc/gdm
Init	Secuencias de comandos de inicio para configurar el despliegue GDM
PreSession	Secuencias de comandos ejecutables al iniciar sesión
PostSession	Secuencias de comandos que se ejecutan cuando termina la sesión
PostLogin	Secuencias de comandos ejecutables tras iniciar sesión

TABLA 7-7 Archivos de configuración y directorios de GDM

su propio **Xstartup** y los recursos utilizados para controlar la manera en que se despliega la ventana de inicio de sesión KDM, se configuran en el archivo `/etc/X11/xdm/kdmrc` (`/etc/kde/kdm/kdmrc` vincula a éste). Este es el archivo configurado por el Administrador de configuración de KDM.

Arranque de línea de comandos de X Window System: **startx**, **xinit** y **xinitrc**

Si inicia Linux con la interfaz de línea de comandos, entonces una vez haya iniciado sesión, puede utilizar el comando **startx** para iniciar X Window System, el administrador de ventanas y su escritorio. El comando **startx** utiliza el comando **xinit** para iniciar X Window System; su secuencia de comandos de inicio es `/etc/X11/xinit/xinitrc`.

X Window System puede iniciarse desde la interfaz de línea de comandos con el comando **xinit**. No debe invocar el comando **xinit** directamente, sino a través de **startx**, que siempre debe utilizar para iniciar X Window System. El comando **startx** es una secuencia de comandos de shell que ejecuta el comando **xinit**. El comando **xinit**, a su vez, primero busca una secuencia de comandos de inicialización de X Window System denominada **.xinitrc**, en el directorio de inicio del usuario. Si no existe allí, **xinit** usará `/etc/X11/xinit/xinitrc` como secuencia de comandos de inicialización. Ambos **.xinitrc** y `/etc/X11/xinit/xinitrc` tienen comandos para configurar el servidor X Window System y ejecutar cualquier comando inicial de X, como iniciar el administrador del sistema. Puede pensar en `/etc/X11/xinit/xinitrc` como una secuencia de comandos predeterminada. Además, muchos sistemas utilizan un nombre de archivo aparte, llamado **Xclients**, donde pueden especificarse aplicaciones de X particulares, escritorios o administradores de ventana. Estas entradas pueden mostrarse en una lista, directamente en un archivo **xinitrc**, pero se crea un archivo separado para un formato más organizado. El archivo **xinitrc** ejecuta los archivos **Xclients** como secuencias de comandos shell. Existe una versión de usuario y otra de sistema: **.Xclients** y `/etc/X11/Xclients`. Se busca el archivo **.Xclients** en el directorio de inicio del usuario; si falta, se utiliza el archivo `/etc/X11/xinit/Xclients`.

Casi ninguna distribución configura inicialmente las secuencias de comandos **.xinitrc** o **.Xclients** en cualquiera de los directorios de inicio. El usuario que quiera cada uno deberá crearlas. Cada usuario puede cerrar una secuencia de comandos personalizada **.xinitrc** en su directorio de inicio, que configura e inicia X Window System como se desea. Hasta que un usuario configura una secuencia de comandos **.xinitrc**, se utiliza `/etc/X11/xinit/xinitrc` y puede examinar esta secuencia de comandos para ver como inicia X Window System. Ciertas operaciones de configuración requieren que X Window System esté en el archivo **.xinitrc**. Para que un usuario cree su propia secuencia de comandos **.xinitrc**, es mejor copiar primero `/etc/X11/xinit/xinitrc` al directorio de inicio y nombrarlo **.xinitrc**. Entonces cada usuario podrá modificar el archivo **.xinitrc** particular como se requiera. (Observe que el archivo **xinitrc** no tiene punto antes de su nombre, mientras el archivo **.xinitrc**, del directorio de inicio configurado por un usuario sí lo tiene.) En el siguiente ejemplo se muestra una versión simplificada del archivo **xinitrc** del sistema iniciando el administrador de sistema **twm** y una ventana **Xterm**. Los archivos **.Xresources** y **.Xmodmap** del sistema y usuario se ejecutan primero para configurar X Window System.

```
.xinitrc
#!/bin/sh
userresources=$HOME/.Xresources
usermodmap=$HOME/.Xmodmap
sysresources=/etc/X11/.Xresources
sysmodmap=/etc/X11/.Xmodmap

#merge in defaults and keymaps
```

```
if [ -f $sysresources ]; then
    xrdp -merge $sysresources
fi
if [ -f $sysmodmap ]; then
    xmodmap $sysmodmap
fi
if [ -f $userresources ]; then
    xrdp -merge $userresources
fi
if [ -f $usermodmap ]; then
    xmodmap $usermodmap
fi
# start some nice programs
xterm &
exec twm &
```


8

CAPÍTULO

GNOME

El GNU Network Object Model Environment (Entorno para Modelado de Objetos en Red GNU), conocido como *GNOME*, es un entorno poderoso y fácil de usar constando principalmente de un panel, escritorio y un conjunto de herramientas GUI que constituyen la interfaz del programa. GNOME está diseñado para proporcionar una plataforma para el desarrollo de aplicaciones poderosas. Actualmente, varias distribuciones soportan GNOME y es la principal interfaz de Red Hat y Fedora. GNOME es gratuito y se publica bajo la licencia GNU. Puede descargar el código fuente, además de documentación y software adicional de GNOME, directamente del sitio Web de GNOME en **gnome.org**. Varias compañías se han unido para integrar la fundación GNOME, organización dedicada a coordinar el desarrollo de GNOME y sus aplicaciones de software. Incluye empresas como Sun, IBM y Hewlett-Packard, además de distribuidores de Linux como Fedora, SUSE y TurboLinux. Modelada de acuerdo con la Apache Software Foundation, responsable de desarrollar el servidor Web Apache, la fundación GNOME ofrece orientación al desarrollo de GNOME, junto con soporte empresarial, financiero y legal.

Los componentes centrales del escritorio GNOME incluyen un panel para iniciar programas y funcionalidad de escritorio. Aplicaciones compatibles con GNOME proporcionan otros componentes encontrados normalmente en un escritorio, como administración de archivos, explorador Web y administrador de ventanas. GNOME brinda bibliotecas de herramientas para la GUI de GNOME que puede ser usada por desarrolladores para crear aplicaciones GNOME. Podría decirse que los programas usando botones, menús y ventanas, adheridos a los estándares de GNOME, son compatibles con éste. El administrador de archivos oficial para el escritorio de GNOME es Nautilus. El escritorio de GNOME no tiene administrador de ventanas propio, como KDE. En cambio, emplea cualquier administrador de ventanas compatible con GNOME. Metacity es el incluido en la distribución GNOME.

El soporte a interfaces de modelo de componentes está integrado en GNOME, permitiendo que los componentes se interconecten sin importar idioma del sistema en que se implementan o tipo de máquina en que se ejecutan. El estándar utilizado en GNOME para dichas interfaces es Common Object Request Broker Architecture (CORBA, arquitectura común de corredor de solicitudes de objeto), desarrollado por Object Model Group, para utilizarse en sistemas Unix. GNOME maneja la implementación ORBit de CORBA. Con dicha estructura, aplicaciones y clientes de GNOME pueden comunicarse directamente entre sí, permitiendo el uso componentes de una aplicación en otra. Con la versión 2.0, GNOME adoptó oficialmente GConf y sus bibliotecas como método fundamental para configurar GNOME y sus aplicaciones. GConf puede configurar programas coordinados independientemente, como los que integran el administrador de archivos Nautilus.

Sitio Web	Descripción
gnome.org	Sitio Web oficial de GNOME
developer.gnome.org	Sitio Web del desarrollador que trabaja con GNOME
art.gnome.org	Temas de escritorio y arte de fondo
gnomefiles.org	Aplicaciones, applets y herramientas de software de GNOME
gnome.org/gnome-office	Aplicaciones de oficina de GNOME

TABLA 8-1 Recursos de GNOME

Encontrará más información acerca de GNOME en su sitio Web, **gnome.org**. El sitio proporciona documentación en línea, como la guía de usuario GNOME y preguntas frecuentes; también mantiene extensas listas de correo electrónico para proyectos de GNOME a las que se puede suscribir. El sitio **gnomefiles.org** ofrece una lista detallada de software de las aplicaciones y proyectos GNOME. Si quiere desarrollar programas de GNOME, revise el sitio Web de los desarrolladores de GNOME en **developer.gnome.org**. El sitio cuenta con tutoriales, guías de programación y herramientas de desarrollo. Allí encontrará el manual completo de referencia de la API en línea, además de gran cantidad de herramientas de apoyo como tutoriales y entornos de desarrollo integrados (IDE, Integrated Development Environments). El sitio también incluye documentación en línea detallada para la biblioteca GTK+, widgets de GNOME y el escritorio de GNOME. En la tabla 8-1 se ofrece una lista muy útil de sitios relacionados con GNOME.

Características de GNOME 2.x

Revise **gnome.org** para conocer una descripción más detallada de características y mejoras de GNOME, con pantallas y referencias. GNOME lanza nuevas ediciones de manera frecuente. Desde el lanzamiento de la versión 2.0, se han agregado muchas capacidades nuevas con cada versión. Diversas aplicaciones y applets, como Deskbar y GConf, no se instalan como opción predeterminada.

Entre las características de GNOME se incluyen cambios de interfaz a Evolution, GNOME meeting y Eye of GNOME, además del tiempo de carga y uso de memoria más eficientes, lo que ofrece un tiempo de respuesta más rápido. Se ha corregido Gedit para adherirlo a especificaciones de la interfaz de documentación múltiple. Se ha puesto énfasis en nuevas herramientas, como administradores de imagen y cámara F-Spot, así como la herramienta de búsqueda Beagle (ambos paquetes tienen soporte a .NET Mono). El nuevo editor de menús, Alacarte, permite personalizar sus menús fácilmente. El analizador de uso de disco, Baobab, permite ver rápidamente cuánto espacio en disco es usado. El reproductor de video de GNOME, Totem, permite acceso Web, con soporte a características de Windows Media.

Las imágenes de escritorio se basan en Cairo, con iconos intuitivos y amistosos para el usuario. Botones y ventanas son más fáciles de utilizar y atractivos. El tema de imágenes Cairo es compatible con directrices de estilo TANGO, un estándar de fuente abierta para imágenes de escritorio, proporcionando el mismo estilo de imagen en todos los escritorios de fuente abierta. Consulte **tango.freedesktop.org** para adquirir más información al respecto. Además, GNOME se adhiere a especificaciones estándar de asignación de nombres **freedesktop.org**. En realidad, KDE, GNOME y XFCE se apegan a especificaciones de asignación de nombre, usando los mismos nombres estándar en iconos de sus escritorios.

Para cifrado, firma y descifrado de archivos y texto GPG, GNOME proporciona Seahorse Encryption Key Manager, que puede accederse desde el menú Sistema, como la entrada

Preferencias de encriptación. Con Seahorse puede administrar sus claves de cifrado almacenadas en el anillo de GNOME, además de claves y frases de contraseña de OpenPGP SSH. Puede importar claves existentes, buscar claves remotas y crear propias. Los servidores de clave predeterminados se muestran en la lista del panel Servidores de claves, donde puede agregar nuevas. El editor gedit ofrece plug-ins para cifrar archivos de texto, el explorador Web Epiphany para frases de texto y Nautilus para cifrado desde el menú contextual. Un applet del panel le permite cifrar, firmar y descifrar el contenido del portapapeles.

El Centro de control de GNOME brinda organización intuitiva y acceso a la configuración de su escritorio. Esto se integra al escritorio como submenús del menú Sistema | Preferencias. Las preferencias se organizan en las categorías Personal, Visualización y comportamiento, Internet y red, Hardware y Sistema. El Centro de control de GNOME también se implementa como GUI desplegando un cuadro de dialogo con iconos a la izquierda para diferentes categorías como Personal y Hardware, además de una lista continua de preferencias a la derecha. Al seleccionar una categoría se mueve y resaltan las preferencias apropiadas. Puede invocar Centro de control de GUI al insertar `gnome-control-center` en la ventana de Terminal.

GTK+

GTK+ es el conjunto de widgets usado para aplicaciones de GNOME. Su apariencia derivó originalmente de Motif. El conjunto de widgets está diseñado desde la base para ofrecer gran capacidad y flexibilidad. Por ejemplo, los botones pueden tener etiquetas, imágenes o cualquier combinación de éstos. Es posible consultar y modificar objetos dinámicamente en tiempo de ejecución. GTK+ también incluye un motor de temas permitiendo a los usuarios cambiar la apariencia de las aplicaciones mediante estos widgets. Al mismo tiempo, el conjunto de widgets de GTK+ permanece pequeño y eficiente.

El conjunto de widgets de GTK+ es completamente gratuito bajo la licencia mínima pública general (LGPL, Lesser General Public License). LGPL permite a los desarrolladores usar el conjunto de widgets con software de propietario, además de software gratuito (GPL lo restringiría sólo a software gratuito). La colección de widgets también presenta extenso conjunto de uniones con lenguajes de programación, incluidos C++, Perl, Python, Pascal, Objective C, Guile y Ada. La internacionalización tiene soporte total, permitiendo aplicaciones basadas en GTK+ se usen con otros conjuntos de caracteres, como los de idiomas asiáticos. Las funciones de arrastre y colocación soportan operaciones con otros conjuntos de widgets aceptando estos protocolos, como Qt.

La interfaz GNOME

La interfaz GNOME consta de panel y escritorio, como se presenta en la figura 8-1. El panel aparece como barra larga a través de la parte inferior de la pantalla. Almacena menús, programas y applets. (Una *applet* es un pequeño programa diseñado para ejecutarse en el panel.) En el panel superior hay un menú etiquetado Aplicaciones. El menú opera como menú Inicio y muestra una lista de entradas de aplicaciones para ejecutar en su escritorio. Los paneles se despliegan de manera horizontal o vertical y pueden ocultarse automáticamente para mostrar la pantalla completa. El menú Aplicaciones se reserva para aplicaciones. Otras tareas, como abrir una ventana del directorio home o salir de la sesión, se localizan en el menú Lugares. El menú Sistema almacena Preferencias, para configurar su interfaz GNOME, además de Administración, para acceder a herramientas administrativas de la distribución.

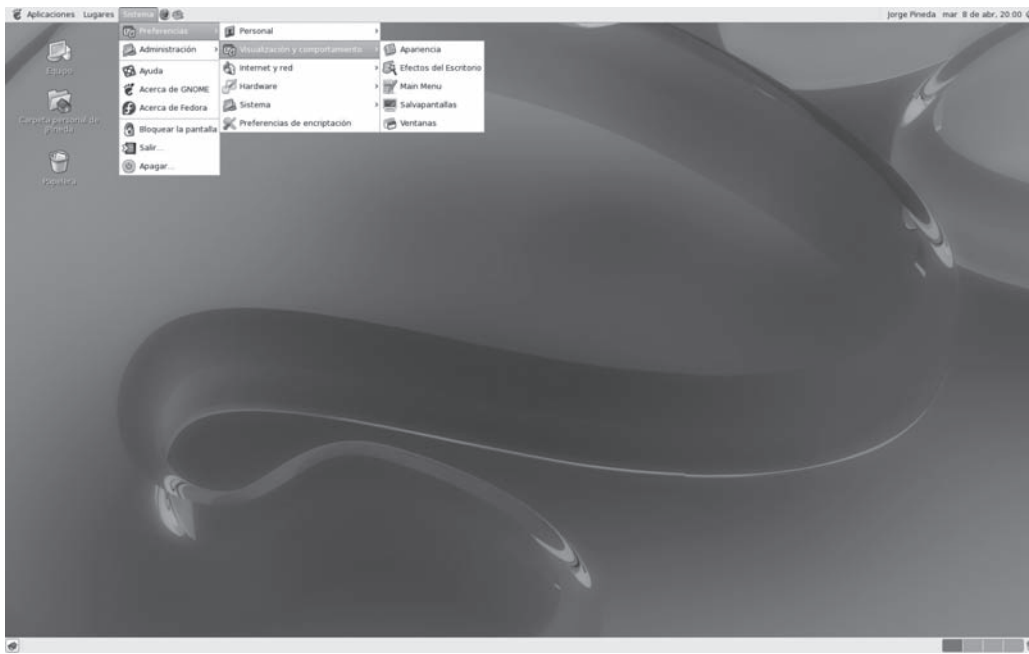


FIGURA 8-1 GNOME con el menú Preferencias

NOTA La interfaz GNOME usa dos paneles, uno en la parte superior para menús y tareas de notificación, como reloj, y otro en la parte inferior, para características interactivas con el espacio de trabajo y aplicaciones ancladas. Ahora se usan tres menús principales, en lugar de uno: Aplicaciones, Lugares y Sistema. El menú Sistema se utiliza para salir de su sesión.

El resto de la pantalla es el escritorio. Aquí, puede colocar directorios, archivos o programas. Puede crearlos directamente en el escritorio o arrastrarlos desde una ventana de administrador de archivos. Una operación clic y colocar moverá un archivo de una ventana a otra o al escritorio. Una operación de clic y colocar con la tecla CTRL oprimida copiará un archivo. La misma operación, pero con el botón central del ratón (los dos botones al mismo tiempo, en un ratón de dos botones) permitirá crear vínculos en el escritorio con programas instalados. Al principio, el escritorio sólo contiene un icono para su directorio home. Al hacer clic en éste, se abre una ventana del administrador de archivos para ese directorio. Un clic con el botón derecho en cualquier parte del escritorio despliega un menú de escritorio con el que puede abrir nuevas ventanas y crear carpetas.

SUGERENCIA Puede desplegar su escritorio de GNOME usando diferentes temas que cambien la apariencia de los objetos del escritorio como ventanas, botones y barras de desplazamiento. La funcionalidad de GNOME no se ve afectada por estos cambios. Puede elegir entre diversos temas. Muchos de éstos publicados en Internet en art.gnome.org. Se conocen técnicamente como temas GTK y permiten el cambio de aspecto del conjunto de widgets de GTK. Para seleccionar un tema, seleccione el menú Tema en Preferencias | Visualización y comportamiento | Apariencia. El tema GNOME predeterminado es Clearlooks.

Componentes de GNOME

Desde el punto de vista del usuario, puede considerar que la interfaz de GNOME tiene cuatro componentes: escritorio, paneles, menús principales y administrador de archivos.

En su configuración estándar predeterminada, el escritorio de GNOME despliega un icono Carpeta, para su directorio home, en la esquina superior izquierda, junto con un bote de basura donde puede eliminar elementos. Además, el escritorio también despliega una ventana Equipo, para acceder a su sistema de archivos, unidades de CD/DVD y recursos compartidos de red. Al hacer doble clic en el icono del directorio home, abrirá el administrador de archivos, desplegando archivos en su directorio home. Tiene dos paneles desplegados, uno para menús, iconos de aplicación y applets en ejecución, en la parte superior de la pantalla y uno en la parte inferior usado principalmente para administrar espacios de ventana y escritorio.

La barra superior tiene diversos menús e iconos de aplicación: Aplicaciones, Lugares, Sistema, explorador Web Mozilla Firefox (un globo con un zorro) y la herramienta de correo electrónico Evolution (un sobre). A la derecha se encuentran los iconos de hora y fecha. Un botón de actualización aparecerá si existen actualizaciones disponibles. Puede usar el icono de actualización para actualizar automáticamente su sistema. La barra inferior almacena iconos para ventanas minimizadas, además de sus applets en ejecución. Cuando abre una ventana, un botón correspondiente se desplegará en el panel inferior, que puede utilizar para minimizar y restaurar la ventana.

Para iniciar un programa, puede seleccionar su entrada en el menú Aplicaciones. También puede hacer clic en su icono de aplicación en el panel (si existe uno) o arrastrar un archivo de datos a este icono.

Salida de GNOME

Para salir de GNOME, seleccione las entradas Salir o Apagar en el menú Sistema. La entrada Salir lo saca de GNOME, devolviéndolo a la ventana de inicio de sesión (o la shell de línea de comandos, con su sesión aún dentro de la cuenta de Linux, si inició GNOME con **startx**). La entrada Apagar despliega un cuadro de diálogo que le permite hibernar, apagar, cancelar o reiniciar su sistema. Una entrada Reiniciar apaga y reinicia su sistema. Debe salir de forma separada de un administrador de ventanas que no sea compatible con GNOME, tras salir de su sesión de GNOME.

Ayuda de GNOME

El explorador Ayuda de GNOME (Yelp) cuenta con una interfaz similar a un explorador para desplegar el manual de usuario de GNOME, páginas Man y documentos de información. Puede seleccionarlo desde el menú Sistema. Presenta una barra de herramientas para recorrer la lista de documentos vistos previamente. Incluso, puede marcar elementos específicos. Una interfaz de explorador le permite utilizar vínculos para conectarse a diferentes documentos. En la página principal, se despliegan a la izquierda vínculos expansibles para varios temas de escritorio de GNOME, con entradas para Guía del usuario y Guía de administración de GNOME en el extremo derecho. En la parte inferior izquierda se muestra una lista de vínculos para páginas Man e Info. Puede usar estos vínculos para desplegar páginas Man e Info de manera sencilla. Use el cuadro Buscar para localizar documentos de ayuda de manera rápida. Protocolos especiales parecidos a URL se usan para diferentes tipos de documentos: **ghelp**, para ayuda de GNOME; **man**, para páginas Man; e **info**, para documentos de información, como **man:fstab** para desplegar la página Man del archivo **fstab**.

El explorador Ayuda de GNOME contiene un manual detallado de cada aspecto de su interfaz GNOME. Los vínculos del lado izquierdo despliegan categorías GNOME para diferentes aplicaciones como herramientas de Sistema y applets de GNOME. La entrada Miniaplicaciones del

panel proporciona descripciones detalladas de todas las applets de GNOME disponibles. Las categorías de aplicaciones como Internet, Programación, Herramientas del sistema y Sonido y video proveen documentos de ayuda para aplicaciones desarrolladas integralmente con el proyecto GNOME, como el cliente de correo electrónico Evolution; Totem, reproductor de películas; Analizador de uso de disco y Monitor del sistema GNOME. Haga clic en la entrada Escritorio, en la sección superior izquierda de la lista, para desplegar vínculos para manuales Guía de usuario y Guía de administración GNOME.

El escritorio de GNOME

El escritorio de GNOME cuenta con todas las opciones de sistemas operativos basados en GUI (consulte la figura 8-1). Puede arrastrar archivos, aplicaciones y directorios al escritorio, luego regresar a aplicaciones compatibles con GNOME. Si el escritorio deja de funcionar, puede reiniciarlo desde el administrador de archivos de GNOME (Nautilus). El escritorio realmente es un proceso de segundo plano en el administrador de archivos de GNOME, pero no necesita tener el administrador de archivos abierto para utilizar el escritorio.

***NOTA** Como opción al uso del escritorio, puede arrastrar cualquier programa, archivo o directorio al panel y utilizar éste, en cambio.*

Arrastre y colocación de archivos en el escritorio

Cualquier icono de un elemento arrastrado desde una ventana del administrador de archivos al escritorio, también aparecerá en éste. Sin embargo, la operación de arrastre y colocación predeterminada es **mover**. Si selecciona un archivo en la ventana del administrador de archivos y lo arrastra al escritorio, realmente está moviendo el archivo de su directorio actual al del escritorio de GNOME, ubicado en su directorio home, almacenando todos los elementos del escritorio. Para GNOME, el directorio del escritorio es **DESKTOP**. En caso de arrastrar carpetas de directorio al escritorio, todo el directorio y subdirectorios se moverán al escritorio de GNOME. Para quitar un icono del escritorio, muévelo a la papelera.

También puede copiar un archivo a su escritorio oprimiendo la tecla CTRL y después haciendo clic y arrastrándolo desde una ventana del administrador de archivos a su escritorio. Verá que la pequeña flecha en la esquina superior derecha del icono copiado toma la forma del símbolo +, indicando que ha creado una copia, en vez de mover el original.

***PRECAUCIÓN** Tenga cuidado cuando elimine iconos del escritorio. Si ha movido el archivo al escritorio, entonces el original reside en la carpeta DESKTOP, y cuando lo elimina, estará eliminando el archivo original. Si ha hecho una copia o vínculo del original, entonces sólo estará eliminando el vínculo o copia. Cuando arrastra aplicaciones de un menú o panel al escritorio, sólo estará creando una copia del botón de lanzamiento de la aplicación en el directorio DESKTOP. Éstos puede eliminarlos de manera segura.*

También tiene posibilidad de crear en el escritorio un vínculo a cualquier archivo. Esto es útil si quiere mantener una sola versión de un directorio especificado y tener la capacidad de acceder a él desde el escritorio. También puede usar vínculos a programas personalizados que tal vez no quiera en un menú o panel. Existen dos formas de crear un vínculo. Mientras oprime las teclas CTRL y MAYÚS (CTRL-MAYÚS), arrastre el archivo adonde quiera generar el vínculo. Entonces aparece una copia del icono con una pequeña flecha en la esquina derecha, indicando

que se trata de un vínculo. Puede hacer clic en este vínculo para iniciar el programa, abrir el archivo o directorio, dependiendo del tipo de vínculo creado. Como opción, primero haga clic y arrastre el archivo fuera de la ventana y, después de mover el archivo, pero antes de dejar de oprimir el botón del ratón, oprima la tecla ALT. Esto desplegará un menú contextual con selecciones para Cortar, Copiar y Enlazar. Seleccione la opción Enlazar para crear un vínculo.

La operación de archivo arrastrar y colocar de GNOME funciona en escritorios virtuales proporcionados por el Selector de áreas de trabajo GNOME. Éste, en el panel inferior, crea iconos para cada escritorio virtual del panel, junto con botones de tarea para cualquier aplicación abierta en ellos.

NOTA Pese a que el escritorio de GNOME acepta operaciones de arrastre y colocación, generalmente éstas sólo funcionan con aplicaciones compatibles con GNOME. Puede arrastrar cualquier elemento desde una aplicación compatible con GNOME a su escritorio y viceversa.

Aplicaciones en el escritorio

Generalmente, sólo querrá crear en el escritorio otra forma de acceder a un archivo sin moverlo de su directorio original. Puede hacer esto mediante el botón lanzador de aplicaciones de GNOME o creando un vínculo con el programa original. Los botones del lanzador de aplicaciones son componentes de GNOME, usados en menús y paneles para desplegar y acceder a aplicaciones. Los botones de OpenOffice, en la parte superior del panel, son botones de lanzador de aplicaciones. Para colocar en su escritorio un icono para la aplicación, basta arrastrar el botón de la aplicación desde el panel o menú. Por ejemplo, para colocar en el escritorio un icono para el explorador Web Firefox, sólo arrastre el icono del explorador Web del panel superior a cualquier espacio de su escritorio.

En el caso de aplicaciones que no aparecen en un panel o menú, puede crear un botón lanzador de aplicaciones o vínculo directo, como se describió en la sección anterior. Para crear un lanzador de aplicaciones, primero haga clic con el botón derecho en el fondo del escritorio para desplegar el menú. Después seleccione la entrada Crear un lanzador.

Menú del escritorio de GNOME

También puede hacer clic con el botón derecho en cualquier lugar vacío del escritorio para desplegar el menú del escritorio de GNOME. Éste mostrará una lista de entradas para tareas comunes, como crear un lanzador de aplicaciones, crear una nueva carpeta u organizar el despliegue de iconos. Tenga en mente que la entrada Crear una carpeta, genera un directorio nuevo en su escritorio, específicamente, en el directorio de su escritorio GNOME (**DESKTOP**), dentro del directorio home. Las entradas de este menú se muestran en una lista en la tabla 8-2.

Administrador de ventanas

GNOME trabaja con cualquier administrador de ventanas. Sin embargo, la funcionalidad del escritorio, como las capacidades para arrastrar y colocar, además del Selector de áreas de trabajo de GNOME (analizadas más adelante), sólo funcionan con administradores de ventanas compatibles con GNOME. La versión más reciente de GNOME emplea el administrador de ventanas Metacity. Es totalmente compatible con GNOME y diseñado para integrarse con el escritorio sin duplicar funcionalidad. También se usan otros administradores de ventanas como Enlightenment, IceWM y Window Maker. Revise la documentación del administrador de ventanas para ver si es compatible con GNOME.

Para soporte a 3-D, puede utilizar administradores de ventana compuestos, como Compiz o Beryl. Las ventanas se despliegan usando decoradores, permitiendo que las ventanas se balanceen,

Elemento de menú	Descripción
Crear un lanzador	Crea en el escritorio un icono nuevo para una aplicación.
Crear una carpeta	Crea en su escritorio un nuevo directorio dentro de su directorio DESKTOP .
Crear un documento	Crea un archivo usando las plantillas instaladas.
Ordenar por nombre	Organiza los iconos de su escritorio.
Mantener alineados	Alinea los iconos de su escritorio.
Cortar, Copiar, Pegar	Copia, corta o pega archivos, permitiéndole mover o copiar archivos entre carpetas.
Cambiar el fondo del escritorio	Abre el cuadro de diálogo Fondo, para seleccionar un nuevo fondo para su escritorio.

TABLA 8-2 El menú de escritorio de GNOME

curven y muevan en formas inusuales. Emplean características similares a los escritorios de Mac y Vista actuales. Un administrador de ventanas compuesto se basa en una tarjeta gráfica con soporte de aceleración Open GL 3-D. Asegúrese de que cuenta con soporte para su tarjeta gráfica. Compiz puede instalarse en su distribución como soporte 3-D predeterminado. Visite compiz.org para adquirir más información. Beryl fue desarrollado por Compiz y presenta decoradores de ventana propios. Encontrará más información acerca de Beryl en beryl-project.org

Metacity emplea casi las mismas operaciones de ventana de otros administradores de ventanas. Cambie el tamaño de una ventana haciendo clic en cualquiera de sus lados o esquinas y arrástrelas. Puede mover la ventana con un clic en su barra de título y desplazarla. También hacer clic con el botón derecho y arrastrar cualquier borde para mover la ventana, además de oprimir ALT y hacer clic en cualquier lugar de la ventana. En la esquina superior derecha se muestran los botones Maximizar, Minimizar y Cerrar. Minimizar crea un botón para la ventana en el panel; haga clic en él para restaurarla. Haga clic con el botón derecho en la barra de título de una ventana para desplegar un menú con entradas para operaciones de ventana. Éstas incluyen entradas del espacio de trabajo para mover la ventana a otro espacio de trabajo (escritorio virtual) o todos los espacios de trabajo, que despliega la ventana sin importar a qué espacio de trabajo se mueva.

Administrador de volúmenes de GNOME

Administrar DVD/CD-ROM, lectores de tarjetas, discos flexibles, cámaras digitales y otro medio extraíble es la tarea del Administrador de volúmenes de GNOME. Se trata de una utilería de bajo nivel que permanece transparente para el usuario, aunque puede configurarse la forma en que trata los medios extraíbles con la herramienta de preferencias Unidades y soportes extraíbles. El Administrador de volúmenes de GNOME no sólo permite acceder medios extraíbles, sino todos sus sistemas de archivos montados, remotos y locales, incluidos directorios compartidos de Windows accesibles desde Samba. Tiene la opción de explorar todos sus sistemas de archivos directamente desde GNOME, implementando esta capacidad con el sistema de archivo virtual de gnome (gnome-vfs) al asignar sus unidades, dispositivos de almacenamiento y medios extraíbles. El Administrador de volúmenes de GNOME emplea HAL y udev para acceder a medios extraíbles directamente y Samba para proporcionar soporte de red de Windows. Los medios son montados

por **gnomemount**, una envoltura usada para acceder a HAL y udev, responsables del montaje (ya no se utilizan **/etc/fstab**).

Puede acceder a sus sistemas de archivos y medios extraíbles mediante el icono Equipo, en el escritorio. Se trata de una ventana de alto nivel mostrando iconos para todos los medios extraíbles (medios montados como CD-ROM, discos flexibles, etcétera), su sistema de archivos local y recursos compartidos de red (véase la figura 8-2). Haga doble clic en cualquier icono para abrir una ventana del administrador de archivos desplegar su contenido. El icono del sistema de archivos abrirá una ventana mostrando el directorio raíz, el directorio más alto de su sistema de archivos. El acceso a directorios del sistema estará restringido, a menos que inicie sesión como usuario root. El icono de red abrirá una ventana mostrando los host de red conectados. Al abrir éstos se desplegarán recursos compartidos, como directorios compartidos, permitiéndole copiar archivos y carpetas desde un directorio compartido, de otro host a otro directorio de su sistema. Para explorar sistemas Windows en GNOME con Samba, primero debe configurar su firewall para aceptar conexiones de Samba.

Los medios extraíbles también aparecerán automáticamente como iconos, directamente en su escritorio. Un DVD o CD-ROM se monta automáticamente al insertarlo en su unidad DVD/CD-ROM, desplegando un icono para éste con su respectiva etiqueta. El mismo tipo de acceso se proporciona para lectores de tarjeta, cámaras digitales y unidades USB. Asegúrese desmontar las unidades USB antes de extraerlas, para que los datos se escriban.

Puede acceder después al disco en la unidad DVD/CD-ROM haciendo doble clic en éste o clic con el botón derecho y seleccionando la entrada Abrir. Se desplegará una ventana de administrador de archivos para mostrar el contenido del disco CD-ROM. Para expulsar un CD-ROM, haga clic con el botón derecho en el icono y seleccione Expulsar, del menú emergente. El mismo procedimiento sirve para discos flexibles, utilizando el icono Disco flexibles. Asegúrese de no quitar un disco flexible montado hasta haberlo desmontado, seleccionando la entrada Expulsar del menú emergente.

Grabar datos en un DVD/CD es sólo cuestión de colocar un DVD en blanco en su unidad. Nautilus reconoce automáticamente el disco en blanco y permite escribir en él. Todos los discos de lectura y escritura, aunque no estén en blanco, también se reconocen como discos para grabar y se abren en una ventana de grabador DVD/CD. Para grabar un disco, sólo arrastre los archivos que quiera copiar a la ventana del disco en blanco y después haga clic en Grabar al disco. Se abrirá un cuadro de diálogo con botones para configurar opciones como velocidad de escritura y nombre de disco. Después de grabado, aparecerá un cuadro de diálogo con una lista de botones para expulsar, volver a escribir y cerrar. Tenga en mente que el nuevo disco escrito no se monta. Puede expulsarlo cuando quiera.

Nautilus también graba imágenes ISO en DVD y CD. Sólo inserte un DVD o CD en blanco y arrastre el archivo con la imagen de disco ISO a un icono de CD/DVD en blanco en su escritorio. Se abrirá un cuadro de diálogo preguntándole si quiere grabar la imagen de CD o DVD. Nautilus trabaja con imágenes ISO (es decir, archivos que terminan con el sufijo **.iso**). Para grabar otros archivos de imágenes como IMG, necesita cambiar el sufijo a **.iso** y Nautilus reconocerá y grabará de manera normal el archivo de imagen.

FIGURA 8-2
Ventana Equipo de
GNOME (Administrador
de volúmenes de
GNOME)



GNOME desplegará iconos para cualquier medio extraíble y realizará operaciones predeterminadas en éstos. Por ejemplo, los CD de audio se reproducirán automáticamente en el reproductor de CD. Las películas en DVD se iniciarán en un reproductor de DVD. Para configurar la manera en que habrá de tratarse un medio extraíble, utilice las herramientas de preferencias Unidades y soportes extraíbles, que puede accederse desde la entrada Unidades y soportes extraíbles, en el menú Sistema | Preferencias | Hardware. Ciertas configuraciones ya existen.

NOTA GNOME ahora administra todos los medios extraíbles directamente con HAL, en vez de las entradas *fstab*.

El administrador de archivos de GNOME: Nautilus

Nautilus es el administrador de archivos de GNOME, que acepta características estándar para copiar, eliminar y borrar elementos, además de configurar permisos y elementos de despliegue. También brinda mejoras como capacidades de zoom, niveles de usuario y soporte a temas. Puede agrandar o reducir el tamaño de sus iconos de archivo; seleccionar los niveles de uso: novato, intermedio o experto; y personalizar la apariencia de Nautilus con diferentes temas. Nautilus hace posible configurar vistas personalizadas de listas de archivos, permitiéndole desplegar imágenes para iconos de directorio y ejecutar aplicaciones de componente en una ventana de administrador de archivos. Nautilus implementa un método espacial para la exploración de archivos. Una nueva ventana se abre para cada nueva carpeta.

Ventana de Nautilus

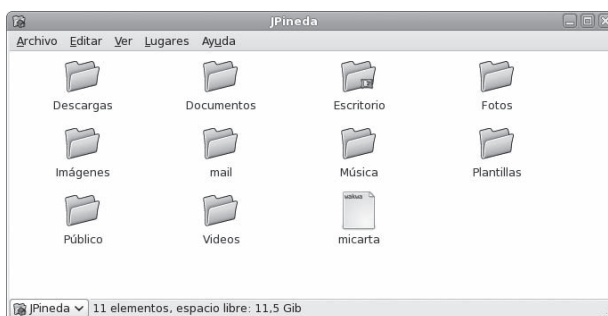
Nautilus fue diseñado como shell de escritorio cuyos distintos componentes pueden emplearse para agregar funcionalidad. Por ejemplo, en Nautilus puede ejecutarse un explorador Web para proporcionar capacidades de explorador Web en una ventana del administrador de archivos Nautilus. Un visor de imágenes puede desplegarlas. El reproductor de medios de GNOME puede ejecutar archivos de video y sonido. La herramienta GNOME File Roller puede comprimir archivos, además de extraerlos de sus respectivos archiveros. Con la implementación de GStreamer, las herramientas multimedia como la grabadora de audio de GNOME están ahora integradas de manera más sencilla en Nautilus.

SUGERENCIA Varias distribuciones como Red Hat y Fedora utilizan Common User Directory Structure (*xdg-user-dirs* en freedesktop.org) para administrar subdirectorios como **Music** y **Video** en el directorio home. Numerosas aplicaciones de escritorio recurren a estos directorios de usuario localizados como opción predeterminada. Los usuarios pueden cambiar el nombre de sus directorios o ubicar uno dentro de otro utilizando el explorador de archivos de GNOME. Por ejemplo, Music puede moverse a **Documents, Documents/Music**. Las configuraciones locales se almacenan en *.config/user-dirs.dirs* file. Las opciones predeterminadas de todo el sistema se configuran en el archivo */etc/xdg/user-dirs.defaults*.

Como opción predeterminada, las ventanas de Nautilus se despliegan con la vista Spatial. Esto ofrece un despliegue simplificado sin barras de herramientas o barras laterales (véase la figura 8-3). Mucha de esta funcionalidad se ha desplazado a menús y ventanas emergentes, dejando más espacio para desplegar archivos y carpetas. Sin embargo, puede abrir una ventana de Nautilus en la vista Navegador, que muestra la barra de menús y barras de herramientas de ubicación tradicionales. Para abrir una ventana en la vista Navegador, haga clic con el botón derecho en el icono de carpeta y seleccione Navegar por la carpeta, del menú desplegable.

FIGURA 8-3

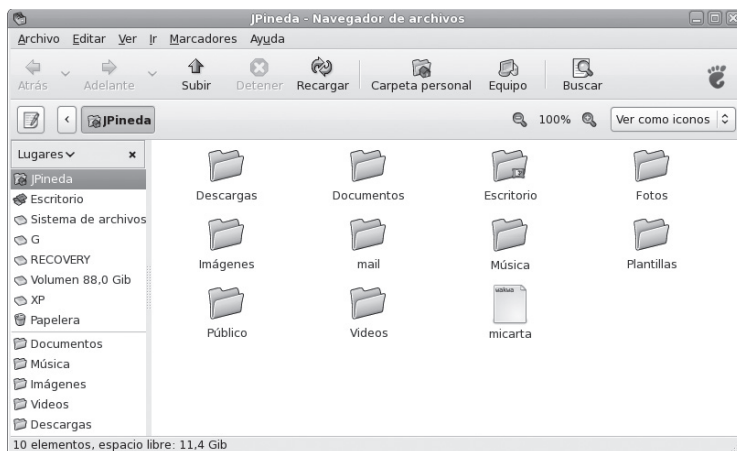
Vista espacial
de la ventana de
Nautilus



La vista Spatial de la ventana de Nautilus, presenta una barra de menús en la parte superior, con menús para administrar sus archivos. Una barra de información en la parte inferior da información acerca del directorio o los archivos seleccionados. En la parte inferior izquierda se encuentra una ventana emergente indicando directorios principales para su directorio de trabajo actual. Puede seleccionar cualquier entrada para abrir una ventana para ese directorio.

Con la vista Navegador, una ventana de Nautilus despliega barras de herramientas, incluidas barra de menús de los comandos del administrador de archivo y una barra de herramientas Lugar, en la parte superior, que puede cambiar entre un cuadro de ubicación o botones de vistas (véase la figura 8-4), junto con una barra lateral para información de archivos y directorios. El resto de la ventana se divide en dos paneles. El panel de la izquierda es utilizado para desplegar información acerca de su directorio de trabajo actual. El panel de la derecha es el panel principal, mostrando la lista de archivos y subdirectorios en el directorio de trabajo actual. Una barra de estado en la sección inferior de la ventana presenta información en torno a un archivo o directorio seleccionado. Puede activar o desactivar cualquiera de estos elementos seleccionando sus entradas en el menú Ver.

Junto a la barra Lugar (cuadro o botón), se encuentra un elemento para acercar o alejar la vista de los archivos. Haga clic en el botón + para acercarse y – para alejarse. Junto al elemento zoom hay menú desplegable para seleccionar vistas diferentes para sus archivos, como iconos, iconos pequeños o detalles.

**FIGURA 8-4** Vista Navegador, ventana del administrador de archivos de Nautilus

NOTA Nautilus viene con soporte integrado para grabar DVD y CD con el paquete *nautilus-cd-burner*, para archivos e imágenes ISO.

Barra lateral de Nautilus: árbol, historial y notas

La barra lateral tiene varias vistas diferentes, elegibles desde un menú emergente, para desplegar información adicional acerca de archivos y directorios: Lugares, Información, Árbol, Histórico y Notas. La vista Lugares muestra los sitios de su sistema de archivos a que accedería normalmente, empezando con su directorio home. Sistema de archivos lo ubica al inicio del sistema de archivos, permitiéndole moverse a cualquier región accesible del mismo. Información exhibe información detallada de su directorio actual o archivo seleccionado. Por ejemplo, si hace doble clic en un archivo de imagen, el panel Información desplegará datos detallados acerca de la imagen, mientras el panel Ventana mostrará la imagen completa. La vista Árbol despliega una vista jerárquica basada en un árbol de archivos y directorios de su sistema, resaltando el seleccionado. Puede utilizar ese árbol para moverse a otros directorios o archivos. El árbol crea un mapa de todos los directorios de su sistema, empezando por el directorio raíz. Puede expandir o encoger cualquier directorio haciendo clic en el símbolo + o –, antes del nombre. Para seleccionar un directorio haga clic en el nombre de directorio. El contenido de ese directorio se despliega entonces en el panel principal. La vista Histórico muestra archivos o directorios accedidos; es útil para moverse hacia delante o atrás a través de los directorios.

La vista Notas despliega notas ingresadas de un elemento o directorio. La vista Notas abre una ventana de texto para editar en el panel lateral. Sólo seleccione la vista Notas y escriba sus notas. Para agregar una nota en un elemento determinado, imagen o archivo de sonido, sólo haga doble clic en él para desplegarlo o ejecutarlo y después seleccione la vista Nota para escribir una. También puede hacer clic con el botón derecho en el elemento, para desplegar un menú emergente del elemento y seleccionar preferencias, desde donde puede hacer clic en un panel Notas. Luego de haber agregado una nota, verá una imagen de nota agregada al icono del elemento en la ventana de Nautilus.

Despliegue de archivos y carpetas

El contenido de un directorio se presenta en forma de iconos o lista detallada. En la vista Spatial, es posible seleccionar opciones diferentes desde el menú Ver. En la vista Navegador, puede usar el menú emergente ubicado en el extremo derecho de la barra Lugar. Lista presenta nombre, permisos, tamaño, fecha, propietario y grupo. En la presentación Ver como lista, se despliegan botones para cada campo a través de sección superior del panel principal. Puede utilizar estos botones para ordenar la lista de acuerdo con ese campo. Por ejemplo, en caso de ordenar los archivos por fecha, haga clic en el botón Fecha; para ordenar por tamaño, haga clic en Tamaño.

En la vista Icono, puede ordenar iconos y revisar una vista previa del contenido sin abrirlos. Para ordenar elementos en la vista Icono, seleccione la entrada Ordenar los elementos, en el menú Vista (en la vista Spatial o Navegador) y después seleccione una opción de diseño. Ciertos tipos de iconos de archivo desplegarán vistas previas del contenido (por ejemplo, iconos para archivos de imágenes desplegarán una versión pequeña de la imagen). Un archivo de texto desplegará en su icono las primeras palabras del texto. La entrada Ampliar hace más grande la vista de su ventana, aumentando el tamaño de los iconos y Reducir reduce su vista, haciéndolos más pequeños. Tamaño normal los restaura al tamaño estándar. También utilice los botones + o – en la barra Lugar para cambiar tamaños.

En las vistas Spatial y Navegador, puede cambiar el tamaño de iconos individuales. Seleccione el icono y después elija la entrada Estirar del menú Editar. Aparecerán controladores en la imagen de icono. Haga clic y arrastre los controladores para cambiar su tamaño. Para restaurar el icono, seleccione Restablecer el tamaño original del icono, en el menú Editar.

Para añadir un emblema a cualquier icono de archivo o directorio, seleccione la entrada Fondos y emblemas, del menú Editar, para abrir la ventana correspondiente. Aquí verá tres iconos para desplegar paneles de color y patrón de fondo, además de emblemas de archivo y directorio. Haga clic en uno de los emblemas para mostrar la selección de emblemas. Para agregar un emblema a un icono de archivo o directorio, haga clic y arrastre el emblema del panel Emblema al icono de archivo o directorio. El emblema aparecerá en ese icono. Si quiere agregar su propio emblema, haga clic en el botón Añadir un emblema nuevo, para buscar un archivo de imagen de emblema por nombre, o para buscar en su sistema de archivos la imagen que quiere utilizar (haga clic en el icono Imagen).

Menú de Nautilus

Haga clic en cualquier región del panel principal para desplegar un menú emergente con entradas para administrar y ordenar sus iconos del administrador de archivos (véase la tabla 8-3). El menú es el mismo para las vistas Spatial y Navegador. Para crear una nueva carpeta, seleccione Crear una carpeta. La entrada Organizar los elementos desplegará un submenú con entradas para ordenar sus iconos por nombre, tamaño, tipo, fecha o incluso emblema. La entrada Manualmente le permite mover iconos a cualquier lugar del panel principal. También puede cortar, copiar y pegar archivos para moverlos o copiarlos entre carpetas de manera más sencilla.

SUGERENCIA Para cambiar el fondo utilizado en la ventana del Administrador de archivos, seleccione *Fondos y emblemas*, del menú *Editar*, arrastrando el fondo que desee a la ventana del administrador de archivos. Puede seleccionar colores o patrones.

Directorios de navegación

Las vistas Spatial y Navegador emplean diferentes herramientas para navegar directorios. La vista Spatial se basa más en operaciones directas de ventanas, mientras la vista Navegador trabaja más como explorador. Recuerde que para abrir un directorio con la vista Navegador, debe hacer clic con el botón derecho en el icono del directorio y seleccionar Navegar la carpeta.

Elemento de menú	Descripción
Crear una carpeta	Crea un nuevo subdirectorio en el directorio.
Crear un documento	Crea un nuevo documento utilizando las plantillas instaladas.
Organizar elementos	Despliega un submenú para ordenar archivos por nombre, tamaño, tipo, fecha o emblema.
Cortar, Copiar, Pegar	Corta, copia o pega archivos, permitiéndole mover o copiar archivos entre carpetas.
Ampliar	Proporciona una vista más cercana de los iconos, haciéndolos parecer más grandes.
Reducir	Proporciona una vista más distante de los iconos, haciéndolos parecer más pequeños.
Tamaño normal	Restaura la vista de los iconos al tamaño estándar.
Propiedades	Abre los paneles Propiedades para el directorio abierto en la ventana.
Ordenar por nombre	Organiza los iconos por nombre.

TABLA 8-3 Menú del administrador de archivos de Nautilus

Navegación en la vista Spatial

En la vista Spatial, Nautilus abrirá una nueva ventana para cada directorio seleccionado. Para abrir un directorio, haga doble clic en éste o clic con el botón derecho y seleccione la entrada Abrir. El menú emergente del directorio principal, en la parte inferior izquierda, abre una ventana para cualquier directorio principal; en efecto, moviéndose a un directorio anterior. Para brincar a un directorio específico, seleccione la entrada Abrir lugar, del menú Archivo. Esto, por supuesto, abrirá una ventana nueva para ese directorio. La entrada Abrir contenedora, en el menú Archivo le permite acceder de manera rápida una nueva ventana para su directorio principal. Se dará cuenta rápidamente que moverse a diferentes directorios abrirá muchas ventanas nuevas.

Navegación en la vista Navegador

La vista Navegador, del administrador de archivos de Nautilus opera de manera similar a un explorador Web, usando la misma ventana para desplegar directorios abiertos. Mantiene una lista de directorios vistos previamente y puede avanzar o retroceder por esa lista, usando los botones de la barra de herramientas. El botón FLECHA IZQUIERDA lo lleva al directorio desplegado antes, y el botón FLECHA DERECHA al siguiente directorio desplegado. El botón FLECHA ARRIBA lo mueve al directorio principal y el botón HOME lo mueve a su directorio home. Para utilizar un nombre de ruta para ir directamente a un directorio determinado, escriba el nombre de ruta en el cuadro Lugar y oprima ENTER. Utilice el icono interruptor, a la izquierda de la barra de ubicación, para cambiar entre las vistas de ubicación de cuadro y botón.

Para abrir un subdirectorio, puede hacer doble clic en su icono o solo clic en el icono y seleccionar Abrir, del menú Archivo. Si quiere abrir una ventana, aparte de la vista Navegador de Nautilus para ese directorio, haga clic con el botón derecho en el icono del directorio y seleccione Abrir una nueva ventana.

Administración de archivos

Como es un administrador de archivos compatible con GNOME, Nautilus soporta operaciones GUI para arrastrar, colocar, copiar y mover archivos. Para mover un archivo o directorio, haga clic y arrástrelo de un directorio a otro, como haría en las interfaces de Windows o Mac. La operación mover es la predeterminada para arrastrar y colocar en GNOME. Para copiar un archivo, haga clic y arrastre normalmente, mientras oprime la tecla CTRL.

NOTA Si mueve un archivo a un directorio en otra partición (sistema de archivos), se copiará en vez de moverse.

El menú Archivo

También puede realizar operaciones de eliminación, cambio de nombre y creación de vínculos en un archivo, haciendo clic con el botón derecho en su icono y seleccionando la acción deseada en el menú emergente (véase la tabla 8-4). Por ejemplo, para eliminar un elemento, haga clic en éste y seleccione la entrada Mover a la papelera, del menú emergente. Esto coloca el archivo en el directorio **Trash**, donde puede eliminarlo más tarde al seleccionar Vaciar la papelera, del menú Archivo de Nautilus. Para crear un vínculo, haga clic con el botón derecho en el archivo y seleccione Crear un enlace, del menú emergente. Esto crea un nuevo archivo de vínculo que comienza con el término “enlace”.

Cambio de nombre de archivos

Para cambiar el nombre de un archivo, haga clic con el botón derecho en el icono del archivo y seleccione la entrada Renombrar, del menú emergente (u oprima la tecla R). El nombre del icono se resaltarán en un fondo negro, encerrado en un cuadro de texto pequeño. Entonces haga clic en el nombre y elimine la etiqueta anterior escribiendo un nuevo nombre. También puede cambiar el

Elemento de menú	Descripción
Abrir	Abre un archivo con su aplicación asociada. Los directorios se abren en el administrador de archivos. Se mostrará una lista con aplicaciones asociadas.
Abrir una nueva ventana	Abre un archivo o directorio en una ventana separada. Sólo vista Navegador.
Abrir con otra aplicación	Selecciona una aplicación con la que se abrirá el archivo. Se despliega un submenú de posibles aplicaciones.
Cortar, copiar, pegar archivos	Entradas para cortar, copiar o pegar archivos.
Crear un enlace	Crea un vínculo a un archivo en el mismo directorio.
Renombrar	Cambia el nombre de un archivo.
Mover a la papelera	Mueve un archivo al directorio Trash , donde puede eliminarlo después.
Crear archivador	Archiva un archivo utilizando File Roller.
Enviar a	Envía un archivo por correo electrónico.
Propiedades	Despliega el cuadro de diálogo Propiedades de un archivo. Existen tres paneles: Estáticos, Opciones y Permisos.

TABLA 8-4 Menú desplegable Archivo de Nautilus

nombre de un archivo al insertar un nuevo nombre en su cuadro de diálogo Propiedades. Haga clic con el botón derecho y seleccione Propiedades, del menú emergente, para desplegar el cuadro de diálogo Propiedades. En la ficha Básico, cambie el nombre del archivo.

Agrupación de archivos

Las operaciones de archivo pueden realizarse en un grupo selecto de archivos y directorios. Hay varios métodos para seleccionar un grupo de elementos. Puede hacer clic en el primer elemento y después mantener oprimida la tecla MAYÚS, conforme hace clic en el último elemento. También oprima y arrastre el ratón a través de los elementos que quiera seleccionar. Para seleccionar elementos separados, mantenga oprimida la tecla CTRL mientras hace clic en iconos individuales. Si quiere seleccionar todos los elementos del directorio, seleccione la entrada Seleccionar todo, en el menú Editar. Entonces haga clic y arrastre el conjunto de elementos todos a la vez. Esto le permite copiar, mover o incluso eliminar más de un archivo a la vez.

Aplicaciones y archivos: Abrir con

Puede iniciar cualquier aplicación en el administrador de archivos haciendo doble clic en la propia aplicación o un archivo de datos empleando dicha aplicación. Si quiere abrir el archivo con una aplicación específica, haga clic con el botón derecho en el archivo y seleccione la entrada Abrir con otra aplicación. Un submenú desplegará una lista de posibles aplicaciones. Si su aplicación no se muestra en la lista, seleccione Otra aplicación, para abrir el cuadro de diálogo Seleccione una aplicación, donde puede elegir aquella con la que quiere abrir el archivo. Utilice también un visor de texto para desplegar el contenido de un archivo en la ventana de administración de archivos. Las operaciones de arrastre y colocación también funcionan con aplicaciones. Puede arrastrar un archivo de datos al icono de aplicación asociada (digamos, una en el escritorio); entonces se lanzará la aplicación usando ese archivo.

Para cambiar o configurar la aplicación que se empleará de manera predeterminada con cierto tipo de archivo, abra Propiedades del archivo y seleccione el panel Abrir con. Aquí puede seleccionar la aplicación predeterminada que se utilizará con ese tipo de archivo. Por ejemplo, el cambio de la opción predeterminada, para un archivo de imagen, del Visor de imágenes a KView, convertirá

KView en el visor predeterminado para todas las imágenes. Si la aplicación que quiere no se encuentra en la lista, haga clic en el botón Agregar, del panel Abrir con, para desplegar una lista de aplicaciones y seleccionar la de su interés. Esto desplegará el cuadro Agregar aplicación y un botón Explorar. Las aplicaciones utilizadas comúnmente ya se encuentran en la lista. Si ya conoce el nombre de ruta completa para la aplicación, puede ingresarla directamente. Si la aplicación no se muestra en la lista, haga clic en Explorar para desplegar el cuadro Seleccione una aplicación, que mostrará una lista con las aplicaciones que puede seleccionar. Al principio, las aplicaciones del directorio `/usr/bin` se muestran en la lista, aunque puede explorar otros directorios. Una vez seleccione su aplicación, aparecerá en la lista Abrir con para ese archivo.

Si existe una aplicación en el panel Abrir con que no quiere se muestre en la lista de opciones, selecciónela y haga clic en el botón Eliminar.

Por ejemplo, para asociar archivos BitTorrent con la aplicación original BitTorrent, haga clic con el botón derecho en cualquier archivo BitTorrent (con la extensión **.torrent**), seleccione la entrada Propiedades y después elija el panel Abrir con. Se desplegará una lista de aplicaciones instaladas, como Ktorrent, Azureus y BitTorrent. Haga clic en BitTorrent para usar la aplicación original BitTorrent y después cierre el cuadro. BitTorrent será entonces la opción predeterminada para los archivos **.torrent**.

SUGERENCIA La herramienta Aplicaciones preferidas le permitirá configurar aplicaciones predeterminadas para Internet y aplicaciones del sistema, como el explorador Web, cliente mail y consola de la ventana Terminal. Las aplicaciones disponibles se muestran en una lista en menús emergentes. Incluso puede seleccionar de una lista de aplicaciones instaladas para elegir un programa personalizado. Puede acceder a la herramienta Aplicaciones preferidas desde el submenú Personal ubicado en el menú Sistema | Preferencias.

Lanzador de aplicaciones

Ciertos archivos, como las secuencias de comandos de shell, están hechos para ejecutarse como aplicaciones. Para ejecutar el archivo mediante un icono, como haría con otras aplicaciones instaladas, puede crear un lanzador de aplicaciones. Para crear lanzadores de aplicación se utiliza la herramienta Crear un lanzador. Esta herramienta puede accederse desde el menú de escritorio con la entrada Crear un lanzador o desde el cuadro Agregar a, del menú del panel, como la entrada Lanzador de aplicación personalizado. Cuando se crea desde el escritorio, el nuevo lanzador se coloca en el escritorio; cuando se crea desde un panel, se colocará directamente en el panel.

La herramienta Crear un lanzador pedirá el nombre de la aplicación, comando que lo invoca y tipo de lanzador. Para el tipo de lanzador tiene las opciones aplicación, archivo o archivo dentro de una terminal. Para las secuencias de comandos shell, use la opción Aplicación en terminal, que ejecuta la secuencia de comandos dentro de una shell.

En el caso de un archivo de datos, utilice el tipo de archivo con que iniciará automáticamente una aplicación asociada, abriendo el archivo, como una página Web, que iniciará entonces un explorador Web. En vez de un comando, se le pedirá escribir la ubicación del archivo.

Para Aplicación y Aplicación en terminal, se le pedirá elegir el comando que habrá de utilizar. Para hacer esto (la aplicación actual o archivo de secuencia de comandos), escriba el nombre de ruta, si la conoce, o use el botón Explorar, para abrir una ventana de explorador de archivos y seleccionarlo.

Con el fin de asignar un icono para su lanzador, haga clic en el botón Icono, inicialmente etiquetado Sin icono. Esto abrirá la ventana Examinar iconos, mostrando los iconos que podrá seleccionar.

Propiedades de archivo y directorio

Con el cuadro de diálogo Propiedades, puede ver información detallada de un archivo, así como configurar opciones y permisos (véase la figura 8-5). Un cuadro Propiedades tiene cinco paneles:

FIGURA 8-5
Propiedades de
archivo en Nautilus



Básico, Emblemas, Permisos, Abrir con y Notas. El panel Básico muestra información detallada como tipo, tamaño, ubicación y fecha de modificación. El tipo es MIME, indicando el tipo de aplicación asociada con éste. El icono del archivo se despliega en la parte superior y puede editar el nombre de archivo en el cuadro de texto bajo el icono. Si quiere cambiar la imagen del icono utilizada para el archivo o carpeta, haga clic en la imagen del icono, en el panel Básico (después del nombre). Se abrirá el cuadro de diálogo Seleccione el icono personalizado, mostrando iconos disponibles; elija el que desee. El directorio **pixmaps** almacena el conjunto de imágenes predeterminadas actuales, aunque también puede seleccionar imágenes propias. Haga clic en la entrada Imagen para ver sus iconos desplegados en el panel de la derecha. Si hace doble clic, la imagen del icono cambia.

El panel Emblemas le permite configurar el que quiere desplegar para este archivo, mostrando todos los emblemas disponibles. Un emblema aparecerá en la esquina superior derecha del icono, indicando el contenido o importancia del archivo.

El panel Permisos presenta los permisos de lectura, escritura y ejecución para propietario, grupo y otros, como se configuró para el archivo. Puede cambiar cualquiera de los permisos aquí, suponiendo que el archivo le pertenece. Configure el acceso para propietario, grupo y otros, al utilizar los menús emergentes. Configure los permisos de propietario como Sólo lectura o Lectura y escritura. En el caso de grupos y otros, también puede configurar la opción Ninguno, denegando el acceso. El nombre del grupo se expande a un menú emergente mostrando una lista de grupos diferentes; seleccione el que quiera para cambiar el grupo del archivo. Si quiere ejecutar esto como aplicación (digamos, una secuencia de comandos de shell), marque la entrada Permitir ejecutar el archivo como un programa. Esto tiene el efecto de configurar el permiso de ejecución.

El panel Permisos, para directorios, opera casi de la misma forma, pero incluye dos entradas, Acceso a carpeta y Acceso a archivo. La entrada Acceso a carpeta controla el acceso a la carpeta con opciones para Sólo listar archivos, Acceder a archivos y Crear y borrar archivos. Esto corresponde a los permisos leer, leer y ejecutar, además de leer, escribir y ejecutar, dados a los directorios. La entrada Acceder a archivos configura permisos para todos los archivos del directorio. Son los mismos que para archivos: para propietario, Lectura o Lectura y escritura; para el grupo y otros, la entrada agrega una opción Ninguno, para denegar acceso. Para configurar los permisos de todos los archivos de manera correspondiente en el directorio (no sólo para la carpeta), haga clic en el botón Aplicar permisos a los archivos contenidos.

El panel Abrir con, muestra una lista de todas las aplicaciones asociadas con el tipo de archivo. Puede seleccionar el que quiere como predeterminado. Esto es de utilidad para archivos

multimedia, con los que quizás prefiera un reproductor específico para cierto archivo o visor de imágenes particular para imágenes.

El panel Notas mostrará una lista de cualquier nota que quiera hacer para el archivo o directorio. Es una ventana de texto para edición; así puede cambiar sus notas o agregar elementos, directamente.

Cierto tipo de archivos tendrán paneles agregados, que proporcionan información acerca del elemento. Por ejemplo, un archivo de audio tendrá un panel Audio mostrando el tipo de archivo de audio e información como título de la canción o método de compresión utilizado. Un archivo de imagen tendrá un panel Imagen presentando una lista de resolución y tipo de imagen. Un archivo de video contendrá un panel Video mostrando tipo de archivo de video junto con información de compresión y resolución.

Preferencias de Nautilus

Puede configurar las preferencias del administrador de archivos de Nautilus en el cuadro de diálogo Preferencias, al que se accede seleccionando el elemento Preferencias, del menú Editar. El cuadro de diálogo Preferencias muestra un panel principal con una barra lateral incluyendo entradas para configuración: Vistas, Comportamiento, Visualización, Columnas de la lista y Vista Previa. Puede usar estos cuadros de diálogo para configurar propiedades de despliegue predeterminado del administrador de archivos de Nautilus.

- El panel Vistas permite seleccionar la manera en que se desplegarán los archivos, en forma predeterminada, sea lista o icono.
- Comportamiento le permite elegir la manera de seleccionar archivos, administrar la papelera y manejar secuencias de comandos, además de decidir si usa la vista Navegador de manera predeterminada.
- Visualización selecciona que información agregada desplegar en la leyenda de un icono, como tamaño o fecha.
- La vista Columnas de la lista selecciona qué características se desplegarán en la lista detallada y el orden para desplegarlas. Además de Nombre, Tamaño, Fecha y Tipo ya seleccionados, agregue permisos, grupo, tipo de MIME y propietario.
- El panel Vista previa elige si quiere desplegar vistas previas pequeñas del contenido en los iconos, como el principio de los archivos de texto.

Nautilus como explorador de FTP

Nautilus trabaja como explorador FTP. Puede usar el cuadro Lugar (cambie a vista de cuadro) o la entrada Abrir lugar, en el menú Archivo, para acceder cualquier sitio FTP. Sólo escriba el URL del sitio FTP en el cuadro Lugar y oprima ENTER (no necesita especificar **ftp://**). Se desplegarán las carpetas del sitio FTP y podrá arrastrar archivos a un directorio local para descargarlas. La primera vez que se conecta a un sitio, se abrirá un cuadro de diálogo Autenticación, permitiéndole seleccionar acceso Anónimo o como Usuario. Si selecciona Usuario, entonces debe insertar nombre de usuario y contraseña para ese sitio. Después seleccione si quiere se recuerde su contraseña para dicha sesión o si se almacenará permanentemente en un anillo de claves.

Una vez haya accedido al sitio, puede recorrer las carpetas como haría con cualquier carpeta de Nautilus, abriendo directorios o regresando a los principales. Para descargar un archivo, sólo arrástrelo de la ventana FTP a una ventana de directorio local. Aparecerá un pequeño cuadro de diálogo mostrando el progreso de descarga. Para subir un archivo, sólo arrástrelo de su carpeta local a la ventana del directorio FTP abierta. Su archivo se cargará en el sitio FTP (debe tener permiso para hacerlo). También puede eliminar archivos en los directorios del sitio.

NOTA A diferencia del administrador de archivos KDE Konqueror, Nautilus no es un explorador Web funcional. Es preferible usar un explorador Web para accederla.

El panel de GNOME

El *panel* es el centro de la interfaz GNOME. Desde este puede iniciar aplicaciones, ejecutar applets y acceder áreas del escritorio. Puede considerar que el panel de GNOME es un tipo de herramienta aplicable en su escritorio. Es posible tener varios paneles de GNOME desplegados en su escritorio, cada uno con applets y menús colocados en éstos. En ese sentido, GNOME es flexible, permitiéndole configurar sus paneles en la forma que quiera. En realidad, el escritorio predeterminado de GNOME presenta dos paneles, uno de menús en la parte superior, para aplicaciones y acciones (véase la figura 8-6) y otro en la parte inferior, empleado para ventanas minimizadas y el Selector de áreas de trabajo. Puede personalizar un panel para cubrir sus necesidades, almacenando applets y menús elegidos. Puede agregar un nuevo panel, añadir aplicaciones a un panel e integrar applets.

Las tareas de configuración de paneles, como agregar aplicaciones, seleccionar applets, configurar menús y crear nuevos paneles, se administran desde el menú emergente Panel. Sólo haga clic con el botón derecho en cualquier lugar de su panel para desplegar un menú con entradas para Propiedades, Panel nuevo, Añadir al panel y Eliminar este panel, así como Ayuda y Acerca de los paneles. Panel nuevo le permite crear otros paneles; con Añadir al panel agrega elementos: lanzadores de aplicación, applets para tareas simples (como Selector de áreas de trabajo) y menús (como el menú de las aplicaciones principales). La entrada Propiedades desplegará un cuadro de diálogo para configurar características de ese panel, como posición y capacidades de ocultamiento.

Para agregar un nuevo panel, seleccione la entrada Panel nuevo, en el menú emergente de Panel. Se crea automáticamente un nuevo panel expandido y se despliega en un extremo de su pantalla. Luego puede usar el cuadro Propiedades del panel para configurar diferentes características de pantalla y fondo, a describirse en las secciones siguientes.

Propiedades del panel

Para configurar paneles individuales, use el cuadro de diálogo Propiedades del panel. Para desplegar este cuadro, haga clic con el botón derecho en un panel determinado y seleccione la entrada Propiedades en el menú emergente. En el caso de paneles individuales, puede establecer características de configuración general, además del fondo. El cuadro de diálogo Propiedades del panel incluye un cuadro de diálogo con las fichas General y Fondo. En la versión 2.4, GNOME abandonó los diferentes tipos de paneles, a favor de un solo tipo con diferentes características que ofrecen las mismas capacidades de paneles anteriores.

Despliegue de paneles

En el panel General del cuadro Propiedades del panel, puede determinar cómo quiere se despliegue el panel. Aquí tiene opciones para orientación, tamaño y si quiere botones para expandir, ocultar automáticamente o mostrar botones de ocultación. La entrada Orientación permite seleccionar en qué lado de la pantalla quiere se coloque el panel. Luego puede seleccionar si quiere un panel expandido; éste alcanzará los bordes de la pantalla, mientras para un panel no expandido su tamaño dependerá del número de elementos en él, mostrando controladores en cada extremo. Los paneles expandidos permanecerán fijos en los bordes de la pantalla, mientras los no expandidos pueden moverse, siempre y cuando la característica Mostrar botones de ocultación no esté seleccionada.

FIGURA 8-6 El panel de GNOME en la parte superior del escritorio

Desplazamiento y ocultamiento de paneles expandidos

Los paneles expandidos pueden colocarse en cualquier borde de su pantalla. Puede mover paneles expandidos de un borde de una pantalla a otro arrastrándolos allí. Si ya existe un panel en ese lugar, el nuevo panel se ubicará arriba del existente. No puede mover paneles no expandidos de esta forma. Tenga en cuenta que si coloca un panel expandido en el borde, cualquier menú se desplegará a lo largo de la esquina superior para permitir el despliegue correcto de los menús emergentes. Los paneles ubicados en bordes laterales expandirán su tamaño para acomodar menús. Si tiene varios menús o uno con nombres largos, puede terminar con un panel muy largo.

Tiene la opción de ocultar paneles expandidos de manera automática o manual. Estas son características especificadas en el cuadro General, del panel Propiedades, como Ocultar automáticamente o Mostrar botones de ocultación. Para ocultar paneles automáticamente, seleccione la característica Ocultar automáticamente. Para desplegar un panel de nuevo, mueva su ratón al borde donde se localiza el panel. Puede habilitar o deshabilitar el ocultamiento de botones en la ventana Propiedades, del panel.

Si quiere ocultar un panel manualmente, seleccione Mostrar botones de ocultación. Se desplegarán dos controladores en ambos extremo del panel. Puede seleccionar más adelante si quiere que estos controladores desplieguen flechas. Luego puede ocultar el panel en cualquier momento, haciendo clic en alguno de los botones Ocultar, localizados en cada extremo del panel. Los botones Ocultar son delgados y muestran una flecha pequeña. Es la dirección en la que el panel se ocultará.

Paneles no expandidos: móviles y fijos

Mientras un panel expandido siempre se localiza en el borde de la pantalla, un panel no expandido es móvil. Puede localizarse tanto en un borde de la pantalla, trabajando encogido según la versión expandida, como moverlo a cualquier lugar de su escritorio, tal cual haría con un icono.

Un panel no expandido se encogerá según el número de componentes, mostrando controladores en los extremos. Entonces podrá mover el panel arrastrando sus controladores. Para acceder al menú del panel con su entrada Propiedades, haga clic con el botón derecho en cualquiera de los controladores.

Para fijar un panel no expandido en su posición actual, seleccione la característica Mostrar botones de ocultación, en el cuadro Propiedades. Esto reemplazará los controladores con botones Ocultar y hará el panel fijo. Al hacer clic en un botón Ocultar, se ocultará el panel en el borde de la pantalla, igual que con los paneles expandidos. Si un panel expandido ya se encuentra en ese borde, el botón de un panel no expandido oculto estará arriba de él, como con un panel expandido oculto. La característica Ocultar automáticamente funcionará con paneles no expandidos, colocados en el borde de la pantalla.

Si quiere fijar un panel no expandido en el borde de una pantalla, asegúrese de que está colocado en el borde deseado y después configure la característica Mostrar botones de ocultación.

Fondo del panel

Con el Fondo de panel, en el cuadro de diálogo Propiedades, puede cambiar color o imagen de fondo del panel. En el caso de un color de fondo, haga clic en un botón de color, para desplegar una ventana de selección de color, donde podrá elegir el color de un círculo de colores y su intensidad en un triángulo de color interior. Puede escribir el número, si lo conoce. Una vez haya seleccionado el color, puede utilizar la barra de desplazamiento Estilo, para hacerlo más transparente u opaco. En caso de usar una imagen en vez de un color, seleccione la entrada imagen y utilice el botón Examinar para localizar el archivo de imagen que quiere. En el caso de una imagen, también puede arrastrar y colocar un archivo de imagen del administrador de archivos al panel; esa imagen se volverá la imagen de fondo del panel.

Objetos del panel

Un panel puede contener varios tipos diferentes de objetos. Entre éstos se incluyen menús, lanzadores, applets, cajones y objetos especiales.

- **Menús** El menú Aplicaciones es un ejemplo de panel de menús. Los lanzadores son botones usados para iniciar una aplicación o ejecutar un comando.
- **Lanzadores** El icono del explorador Web es un ejemplo de un botón lanzador. Puede seleccionar cualquier entrada de aplicación en el menú Aplicaciones y crear un lanzador para éste en el panel.
- **Applets** Un applet es una aplicación pequeña, diseñada para ejecutarse en un panel. El selector de áreas de trabajo, mostrando los diferentes escritorios, es un ejemplo de applet de GNOME.
- **Cajones** Un cajón es una extensión del panel que puede abrirse o cerrarse. Considere al cajón una parte reductible del panel. Puede agregar cualquier cosa en un panel regular, incluidos applets, menús aun otros cajones.
- **Objetos especiales** Los objetos especiales se usan para tareas especiales que no realizan otros objetos de panel. Por ejemplo, los botones Cerrar sesión y Bloquear son objetos especiales.

Movimiento, eliminación y bloqueo de objetos

Para mover cualquier objeto dentro del panel, haga clic con el botón derecho y seleccione la entrada Mover. En el caso de lanzadores, arrastre el objeto directamente adonde quiera colocarlo. Para eliminar un objeto de un panel, haga clic con el botón derecho para desplegar un menú emergente, y después elija la entrada Quitar del panel. Para evitar que un objeto se mueva o elimine, configure la característica de bloqueo (haga clic con el botón derecho en el objeto y seleccione la entrada Bloquear). Para permitir después que se mueva, primero debe desbloquear el objeto (haga clic con el botón derecho en el objeto y seleccione Desbloquear).

SUGERENCIA En el panel *Añadir a lista*, los objetos comunes como reloj y reproductor de CD se mezclan con tipos de objetos como menús y aplicaciones. Cuando se agrega un tipo de objeto, como una aplicación, deberá buscar en la lista para encontrar la entrada para ese tipo; en el caso de las aplicaciones, es la entrada *Lanzador de aplicación*.

Adición de objetos

Para agregar un objeto a un panel, elíjalo del cuadro de diálogo, *Añadir al panel* (véase la figura 8-7). Para desplegar el cuadro *Añadir al panel*, haga clic con el botón derecho en el panel y seleccione la entrada *Añadir al panel*. Este cuadro despliega una lista extensa de objetos comunes, además de tipos de objeto. Por ejemplo, desplegará un menú Principal, además de una entrada para crear menús personalizados. Puede indicar que se agregue una aplicación presente en el menú Aplicaciones de GNOME o crear un lanzador para una aplicación que no figure en el menú. Los lanzadores pueden agregarse a un panel arrastrándolos directamente. Entre los lanzadores se incluyen aplicaciones, ventanas y archivos.

Lanzadores de aplicaciones

Si una aplicación ya tiene un lanzador, es fácil agregarlo al panel. Sólo arrástrelo al panel. Esto se creará automáticamente una copia del lanzador que podrá utilizarse en ese panel. Los lanzadores pueden ser elementos de menú o iconos de escritorio.

FIGURA 8-7

El cuadro Añadir al panel, con una lista de objetos



Todas las entradas de su menú Aplicaciones, son lanzadores de aplicaciones. Para agregar una aplicación desde el menú, sólo selecciónela y arrástrela al panel. También puede agregar cualquier icono de aplicación de escritorio a un panel, para agregar una copia a ese panel.

Para cualquier elemento del menú, también puede ir a su entrada y hacer clic con el botón derecho en él. Luego seleccione la entrada Añadir este lanzador al panel. Automáticamente, se agrega al panel un lanzador para esa aplicación. Suponga que utiliza gedit con frecuencia y quiere agregar su icono al panel, en vez de dirigirse al menú Aplicaciones todo el tiempo. Haga clic con el botón derecho en la entrada de menú Editor de texto, en el menú Accesorios, y seleccione la opción Añadir este lanzador al panel. Ahora el icono gedit aparecerá en su panel.

También seleccione la entrada Añadir al panel, en el menú del panel, y después seleccione la entrada Lanzador de aplicaciones. Esto desplegará un cuadro con una lista de todas las entradas del menú Aplicaciones junto con los menús Preferencias y Administración, expansibles para sus elementos. Sólo encuentre la aplicación que desea agregar y selecciónela. Éste será un método más sencillo, si está trabajando con muchos paneles diferentes.

Tenga en cuenta que puede arrastrar al panel cualquier lanzador creado previamente en su escritorio, para tener una copia de él colocada en el panel.

Lanzadores de carpetas y archivos

Para agregar una carpeta a un panel, sólo arrástrela directamente de una ventana del administrador de archivos o el escritorio. Para agregar un archivo, también puede arrastrarlo directamente al panel, pero entonces deberá crear un lanzador para éste. Se desplegará la ventana Crear lanzador y puede dar al lanzador de archivos nombre y seleccionar un icono para él.

Añadición de cajones

También puede agrupar aplicaciones en el icono Cajón. Al hacer clic en el icono Cajón, se despliega una lista de diferentes iconos de aplicación para seleccionar uno. Para agregar un cajón a su panel, haga clic con el botón derecho en el panel y seleccione la entrada Agregar al panel, para desplegar la lista correspondiente. En esa lista seleccione la entrada Cajón. Esto creará un cajón en su panel. Entonces puede arrastrar cualquier elemento del escritorio, menús o ventanas al icono Cajón, en el panel para mostrarse en la lista del cajón.

Si quiere agregar, como cajón, un menú completo de aplicaciones en el menú principal de su panel, haga clic con el botón derecho en cualquier elemento de ese menú, seleccione Menú completo,

del menú emergente, y después elija la entrada Añadir esto como cajón al panel. El menú completo aparece como un cajón en su panel, almacenando iconos en vez de entradas de menú. Por ejemplo, suponga que quiere colocar el menú Aplicaciones de Internet en su panel. Haga clic con el botón derecho en cualquier elemento de entrada, en el menú Aplicaciones de Internet, seleccione Menú completo y luego Añadir esto como cajón al panel. Aparecerá un cajón en su panel, con la etiqueta Aplicaciones de Internet y al hacer clic en éste, se despliega una lista emergente de iconos para todas las aplicaciones de Internet.

Adición de menús

La diferencia entre menú y cajón es que un *cajón* almacena iconos de aplicaciones, en lugar de entradas de menú. Puede agregar menús a su panel de la misma forma que añade cajones. Para agregar a su panel un submenú, desde el menú Aplicaciones, haga clic con el botón derecho en cualquier elemento y seleccione Menú completo, luego elija la entrada Agregar esto como menú al panel. El título del menú aparece en el panel; puede hacer clic en éste para desplegar las entradas de menú.

Además, puede agregar un menú desde la lista Añadir al panel, al seleccionar Menú personalizado.

Adición de carpetas

También puede agregar carpetas de directorio a un panel. Haga clic y arrastre el icono Carpeta, de la ventana del administrador de archivos a su panel. Cuando haga clic en este botón de la carpeta, se abrirá una ventana del administrador de archivos, desplegando ese directorio. Ya tiene un botón Carpeta para su directorio home. Puede agregar carpetas de directorio a cualquier cajón de su panel.

Objetos especiales del panel

Los objetos especiales del panel realizan operaciones inaccesibles a otros objetos de panel. Actualmente, estos objetos de panel incluyen los botones Bloquear, Salir y Lanzador, además del área de notificación. El botón Bloquear, que despliega un candado, bloqueará su escritorio, ejecutando en su lugar el protector de pantalla. Para acceder a su escritorio, debe hacer clic e insertar después su contraseña, cuando se le pida. El botón Salir muestra una puerta abierta. Al hacer clic en la puerta se desplegará el cuadro de diálogo Salir y entonces puede salir de su sesión. Es lo mismo que seleccionar Salir, en el menú del escritorio. El botón Lanzador muestra un icono de lanzamiento. Abre el cuadro de diálogo Crear un lanzador, permite insertar o elegir una aplicación que desea ejecutar.

El área de notificación está diseñada para almacenar docklets de estado. Un área de notificación proporciona información actual del estado de cualquier aplicación. Las aplicaciones de KDE aceptando docklets de estatus pueden usar el área de estatus de GNOME, cuando se ejecuta bajo GNOME.

Applets de GNOME

Las *applets* son pequeños programas realizando tareas dentro del panel. Para agregar un applet, haga clic con el botón derecho en el panel y seleccione Añadir al panel, desde el menú emergente. Esto despliega el cuadro Añadir al panel, mostrando una lista de applets comunes, junto con otros tipos de objetos, como lanzadores. Seleccione el applet deseado. Por ejemplo, para agregar el reloj a su panel, seleccione Reloj, en el cuadro Añadir al panel. Una vez agregada, el applet se mostrará en el panel. Si quiere quitar un applet, haga clic con el botón derecho en este y seleccione la entrada Quitar del panel.

GNOME presenta varios applets útiles. Algunos sondean su sistema, como el Monitor de carga de batería, para verificar la batería de computadoras portátiles, y Monitor del sistema, indicador

gráfico del uso de CPU y memoria actual. El applet Control de volumen despliega una barra de desplazamiento pequeña para ajustar niveles de sonido. La herramienta Deskbar busca archivos en su escritorio. Monitor de red indica su conexión de red.

Varios applets de utilidad prácticas ofrecen funcionalidad agregada a su escritorio. El applet Reloj despliega la hora en formato de 12 o 24 horas. Haga clic con el botón derecho en el applet Reloj y escoja la entrada Preferencias, para cambiar su configuración. Monitor de frecuencia de la CPU despliega uso de la CPU para procesadores como AMD y nuevos de Intel, que funcionan a menor velocidad cuando están inactivos.

Selector de áreas de trabajo

El *Selector de áreas de trabajo* aparece en el panel e indica sus escritorios virtuales (véase la figura 8-8). Los escritorios virtuales se definen en el administrador de ventanas. Localizado en el lado derecho del panel, el Selector de áreas de trabajo le permite navegar de un escritorio a otro con un clic. Es un applet de panel que funciona sólo en un panel. Puede agregar el Selector de áreas de trabajo a cualquier panel, si lo selecciona en el cuadro Añadir al panel.

El Selector de áreas de trabajo muestra su escritorio virtual completo como rectángulos separados en una lista. Las ventanas abiertas se resaltan como pequeños rectángulos de colores en estos cuadros. Para mover cualquier ventana de un escritorio virtual a otro, haga clic y arrastre su imagen en el Selector de áreas de trabajo. Para configurar este selector, haga clic con el botón derecho en éste y seleccione Preferencias para desplegar el cuadro de diálogo Preferencias. Aquí, puede seleccionar el número de áreas de trabajo. La opción predeterminada es cuatro.

Lista de ventana de GNOME

Lista de ventanas muestra las ventanas abiertas (véase la figura 8-8). Lista de ventanas ordena las ventanas abiertas en una serie de botones, uno para cada ventana. Una ventana puede incluir aplicaciones, como un explorador Web o puede ser una ventana de administrador de archivos, desplegando un directorio. Para ir de una ventana a otra, haga clic en este botón. Cuando minimiza una ventana, puede restaurarla haciendo clic en su entrada en la Lista de ventanas.

Al hacer clic con el botón derecho en un botón de la Lista de ventanas, se abrirá un menú para Minimizar o Desminimizar, Recoger, Mover, Redimensionar, Maximizar o Desmaximizar, o Cerrar la ventana. La operación Minimizar reduce la ventana a su entrada de la Lista de ventanas. Al hacer clic con el botón derecho en la entrada, se despliega el menú con una opción Desminimizar, en vez de una opción Minimizar, que debe utilizarse para desplegar la ventana. Recoger reduce la ventana a su barra de título. Cerrar cierra la ventana, terminando la aplicación.

Si no hay suficiente espacio en el applet Lista de ventanas para desplegar un botón separado por ventana, entonces las ventanas comunes se agruparán en un botón que se expandirá como un menú, mostrando una lista de cada ventana en ese grupo. Por ejemplo, todas las ventanas de Terminal abiertas se agruparán en un solo botón; cuando haga clic en éste, se abrirá una lista emergente de sus botones.

El applet Lista de ventanas se representa por una barra pequeña, aserrada, al principio de la lista de botones de ventana. Para configurar la Lista de ventanas, haga clic con el botón derecho en ella y seleccione la entrada Propiedades. Allí, puede configurar características como tamaño en píxeles, si se agruparán las ventanas, si todas las ventanas se mostrarán, sólo las del espacio de trabajo actual, o hacia que espacio de trabajo se restaurarán las ventanas.

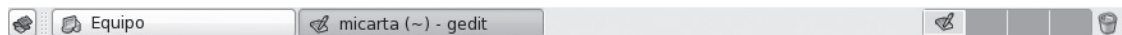


FIGURA 8-8 Panel con Selector de áreas de trabajo y Lista de ventanas, en la parte inferior del escritorio

Configuración de GNOME

Tiene la opción de configurar diferentes secciones de su interfaz GNOME usando las herramientas en la lista del menú Preferencias, del menú Sistema. Este menú desplegará entradas para las preferencias principales de GNOME, organizadas en categorías de submenús, como Hardware y Personal, junto a preferencias mostrando listas de herramientas de tareas específicas, como Dispositivo Palm o Selector de escritorio. Si selecciona una, se abrirá una ventana etiquetada con el nombre de la herramienta, como las preferencias del ratón.

Su sistema GNOME proporciona varias herramientas de escritorio para configurar su escritorio, como Fondo de escritorio, Salvapantallas y Temas. El applet Fondo de escritorio se usa para elegir un color o imagen de fondo, el Salvapantallas selecciona imágenes y tiempo de espera para activar el protector de pantallas; la herramienta Tema sirve para elegir un tema (véase la figura 8-9).

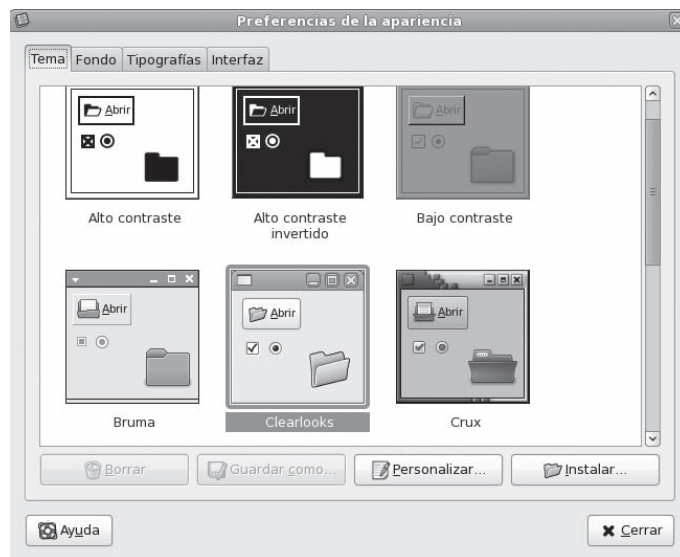
Las herramientas Unidades y soportes extraíbles, así como Preferencias de medios permiten configurar qué acciones realizarán unidades extraíbles, discos y DVD, y cámaras digitales.

Para configurar el audio, la herramienta Sonido le permite seleccionar los archivos de sonido para reproducir eventos en diferentes aplicaciones de GNOME. En el caso de su teclado, puede establecer la sensibilidad de repetición y el sonido del clic con la herramienta Teclado. Puede modificar los botones del ratón para su mano izquierda o derecha y ajustar el movimiento del ratón.

Directorios y archivos de GNOME

Los archivos binarios de GNOME suelen instalarse en el directorio `/usr/bin` de su sistema. Las bibliotecas de GNOME se ubican en el directorio `/usr/lib`. GNOME también tiene su propios directorio `include`, con archivos de encabezado que se usan en compilación y desarrollo de aplicaciones de GNOME, `/usr/include/libgnome-2.0/libgnome` y `/usr/include/libgnomeui` (véase la tabla 8-5). Éstos son instalados por paquetes de desarrollo de GNOME. Los directorios localizados en `/usr/share/gnome` contienen archivos utilizados para configurar su entorno de GNOME.

FIGURA 8-9
Selección de
temas de GNOME



Directorios del sistema GNOME	Contenidos
<code>/usr/bin</code>	Programas de GNOME
<code>/usr/lib</code>	Bibliotecas de GNOME
<code>/usr/include/libgnome-2.0/libgnome</code>	Archivos de encabezado que se utilizarán en aplicaciones de compilación y desarrollo de GNOME
<code>/usr/include/libgnomeui</code>	Archivos de encabezado que se utilizarán en componentes de interfaz de compilación y desarrollo de GNOME
<code>/usr/share/gnome</code>	Archivos utilizados por las aplicaciones de GNOME
<code>/usr/share/doc/gnome*</code>	Documentación para varios paquetes de GNOME, incluidas las bibliotecas
<code>/etc/gconf</code>	Archivos de configuración de GConf
Directorios de usuario de GNOME	Contenidos
<code>.gnome</code> , <code>.gnome2</code>	Archivos de configuración del escritorio de GNOME del usuario y aplicaciones de GNOME; incluye archivos de configuración para panel, fondo, tipos de MIME y sesiones
DESKTOP	Directorio donde residen los archivos, directorios y vínculos colocados en su escritorio
<code>.gnome2_private</code>	El directorio de GNOME privado del usuario
<code>.gtkrc</code>	Archivo de configuración GTK+
<code>.gconf</code>	Base de datos de configuración de GConf
<code>.gconfd</code>	Archivos de administración del daemon gconfd de GConf
<code>.gstreamer</code>	Archivos de configuración multimedia GStreamer de GNOME
<code>.nautilus</code>	Archivos de configuración para el administrador de archivos de Nautilus

TABLA 8-5 Directorios de configuración de GNOME

Directorios de usuario de GNOME

GNOME instala varios archivos y directorios de configuración en el directorio home. Los directorios **.gnome**, **.gnome2** y **.gconf** almacenan archivos de configuración para diferentes componentes del escritorio, como **nautilus** para el administrador de archivos **panel** para los paneles. El directorio **DESKTOP** almacena todos los elementos colocados en el escritorio. El archivo **.gtkrc** es el archivo de configuración del usuario para bibliotecas GTK+, conteniendo las directivas de configuración del escritorio actuales para recursos como combinaciones de teclas, colores y estilos de ventana.

El editor de configuración GConf

GConf ofrece soporte de configuración (no se instala como opción predeterminada). GConf corresponde al Registro utilizado en los sistemas Windows. Consta de una serie de bibliotecas utilizadas para implementar una base de datos de configuración para el escritorio de GNOME. Esta base de datos de configuración estandarizada permite interacciones consistentes entre aplicaciones

de GNOME. Las aplicaciones de GNOME se construyen a partir de varios otros programas, como Nautilus, utilizan GConf para configurar todos los programas de acuerdo con un sólo estándar, manteniendo las configuraciones en una sola base de datos. Actualmente la base de datos GConf está implementada como archivos XML en el directorio del usuario `.gconf`. La interacción y acceso a la base de datos la realiza el daemon de GConf `gconfd`.

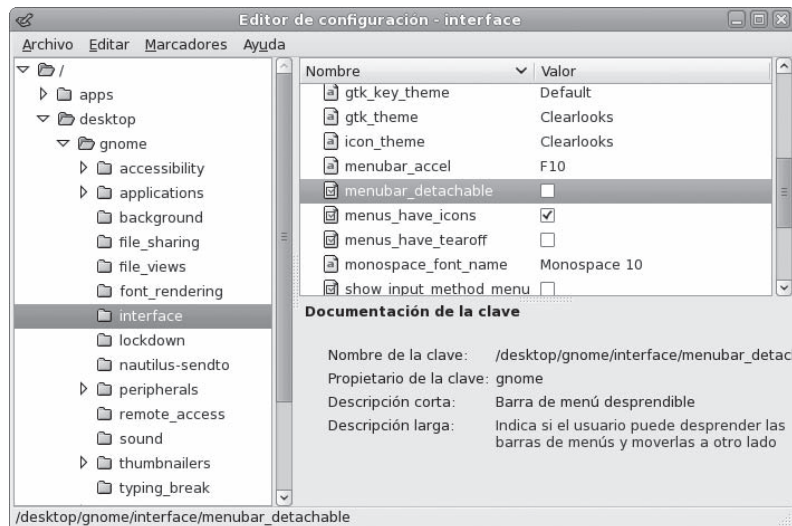
Puede emplear el editor GConf para configurar diferentes aplicaciones y funciones de escritorio de GNOME. Para iniciar el editor GConf, ingrese `gconf-editor` en una ventana de Terminal o seleccione Editor de configuración, del menú Aplicaciones | Herramientas del sistema (menú Aplicaciones). Asegúrese de instalar el paquete `gconf-editor` primero (puede utilizar Pirut, para agregar o quitar software).

Los elementos de configuración son claves específicas organizadas por aplicación y programa. Edite estas claves, cambiando sus valores. En la figura 8-10 se muestran las opciones del editor GConf para las características de despliegue de cuadros de diálogo, utilizadas por la interfaz GNOME.

El editor GConf tiene cuatro paneles:

- **Árbol** El panel para explorar claves, con árboles expansibles para cada aplicación, está localizado a la izquierda. Las entradas de Aplicación se expanden en subentradas, agrupando las claves en diferentes secciones o funciones para la aplicación.
- **Modificación** El panel de la parte superior derecha despliega claves para una entrada seleccionada. El campo de nombre incluirá un icono indicando tipo y el campo Valor es editable, presentando el valor actual. Cambie este valor directamente.
- **Documentación** El panel de la parte inferior derecha despliega información acerca de la clave seleccionada, mostrando nombre de la clave, aplicación a la que pertenece, descripción corta y descripción detallada.
- **Resultados** Este panel, desplegado en la parte inferior, sólo aparece cuando hace la búsqueda de una clave.

FIGURA 8-10
Editor GConf



Una clave tiene un tipo específico, como numérico o de cadena, y sólo podrá hacer cambios empleando el tipo apropiado. Cada clave tiene un icono que especifica su tipo, como una marca de verificación para valores booleanos, un número 1 para valores numéricos y una letra para valores de cadena. Algunas claves tienen menús emergentes con selecciones limitadas para elegir, representadas por un icono con una fila de líneas. Para cambiar el valor de una clave, haga clic en su campo de valor. Entonces podrá editar el valor. En el caso de menús emergentes, haga clic con el botón derecho en un campo de valor para desplegar el menú.

Hay muchas claves distribuidas sobre varias aplicaciones y grupos. Para localizar una, puede usar la función de búsqueda. Seleccione Buscar, del menú Editar, e inserte un patrón. Los resultados se desplegarán en el panel Resultados, donde puede desplazarse para ver las claves coincidentes y seleccionar la deseada.

Los usuarios o administradores pueden hacer los cambios. Los administradores pueden configurar valores predeterminados u obligatorios para las claves. Los valores obligatorios evitan que los usuarios hagan cambios. En el caso de cambios hechos por usuarios, puede abrir la ventana Opciones, al seleccionar Opciones, del menú Archivo. Esto abre una ventana de editor GConf idéntica. Para cambios administrativos, primero debe iniciar sesión como usuario root. Para cambios predeterminados, seleccione la entrada Predeterminado, del menú Archivo y para cambios obligatorios, seleccione la entrada Obligatorios.

K *Desktop Environment* (KDE, entorno de escritorio K) es un escritorio transparente para red que incluye características estándar de un escritorio, como administrador de ventanas y uno de archivos, además de un conjunto extenso de aplicaciones cubriendo casi todas las tareas de Linux. KDE es un sistema que toma en consideración Internet e incluye un conjunto completo de aplicaciones de red e Internet integradas, considerando correo electrónico, lector de noticias y un explorador Web. El administrador de archivos funge como cliente Web y FTP, permitiéndole acceder a sitios de Internet directamente desde su escritorio. KDE se concentra en ofrecer un nivel de funcionalidad de escritorio y facilidad de uso como las encontradas en sistemas Macintosh y Windows, combinado con el poder y flexibilidad del sistema operativo Unix.

KDE Project desarrolla y distribuye el escritorio KDE. Es un grupo grande con cientos de programadores de todo el mundo. KDE es un software abierto y gratuito, proporcionado bajo la licencia pública de GNU, disponible de manera gratuita junto con su código fuente. Un grupo base administra el desarrollo de KDE: KDE Core Team. Cualquiera puede pertenecer, aunque la membresía está basada en méritos.

NOTA *Las aplicaciones de KDE se desarrollan utilizando varias tecnologías que soportan KDE, incluida KIO, con acceso ininterrumpido y modular a archivos y directorios mediante la red. Para comunicación entre procesos, KDE se vale del protocolo de comunicaciones de escritorio (DCOP, Desktop Communications Protocol). KParts es el modelo de objeto de componentes de KDE, empleado para incrustar una aplicación dentro de otra, como una hoja de cálculo dentro en un procesador de palabras. KHTML es un motor de representación y dibujo de HTML.*

Desde el escritorio, es posible acceder fácilmente a varias aplicaciones escritas específicamente para KDE. Entre éstas se incluyen editores, aplicaciones de imágenes de fotografía y dibujo, hojas de cálculo y aplicaciones de oficina. Tales aplicaciones suelen tener la letra K como parte de su nombre (por ejemplo KWord o KMail). Con el escritorio de KDE se ofrecen varias herramientas. Estas incluyen calculadores, ventanas de consola, blocs de notas y administradores de paquetes de software. En cualquier nivel de administración de sistema, KDE ofrece herramientas para su configuración. Con KUser, administra cuentas de usuario, al agregar nuevas o eliminar antiguas. Desde el escritorio de KDE puede realizar casi todas sus tareas de Linux. Las aplicaciones KDE también tiene la aplicación Ayuda integrada. Al seleccionar la entrada Contenido, en el menú Ayuda, inicia el visor Centro de Ayuda de KDE, con una interfaz similar a una página Web, con vínculos para navegación entre

Sitio Web	Descripción
kde.org	Sitio Web de KDE
ftp.kde.org	Sitio FTP de KDE
kde-apps.org	Depósitos de software de KDE
developer.kde.org	Sitio de desarrolladores de KDE
trolltech.com	Sitio para bibliotecas Qt
Koffice.org	Conjunto de oficina KOffice
kde-look.org	Temas de escritorio KDE, seleccione la entrada KDE
lists.kde.org	Listas de correo electrónico de KDE

TABLA 9-1 Sitios Web de KDE

documentos de Ayuda. La versión 3 de KDE incluye soporte para el conjunto de aplicaciones de oficina KOffice, basado en la tecnología KParts de KDE. KOffice incluye una aplicación para presentaciones, hoja de cálculo, ilustrador y procesador de palabra, entre otros componentes. Además, un entorno de desarrollo integrado (IDE, Integrated Development Environment), denominado KDevelop, disponible para ayudar a programadores en la creación de software basado en KDE.

NOTA En KDE, los menús mostrarán más aplicaciones KDE que GNOME, incluido acceso al Centro de control KDE, en el menú principal.

KDE, se inició por Matthias Ettrich, en octubre de 1996, tiene una extensa lista de patrocinadores, incluidos SuSE, Red Hat, Fedora, Mandrake, O'Reilly y otros. KDE se diseñó para ejecutarse en cualquier implementación Unix, incluidos Linux, Solaris, HP-UX y FreeBSD. El sitio Web oficial KDE es **kde.org**, ofreciendo actualizaciones de noticias, vínculos para descarga y documentación. Los paquetes de software de KDE pueden descargarse del sitio FTP de KDE en **ftp.kde.org** y sus sitios espejo. Están disponibles varias listas de correo electrónico de KDE para usuarios y desarrolladores, incluidos anuncios, administración y otros temas (consulte el sitio Web de KDE para suscribirse). Hay gran cantidad de aplicaciones de software disponibles para KDE en **kde-apps.org**. Soporte y documentación para el desarrollo pueden adquirirse en **developer.kde.org**. En la tabla 9-1 se muestran varios sitios Web.

NOTA En la actualidad, nuevas versiones de KDE se liberan con frecuencia, ocasionalmente cada pocos meses. Las versiones de KDE están diseñadas para permitir a los usuarios actualizar sus propias versiones con facilidad. El actualizador de distribuciones pone al corriente KDE desde depósitos de distribución, cuando hay nuevo software disponible. Como opción, puede descargar paquetes de KDE desde el sitio FTP de las distribuciones e instalarlos manualmente. Los paquetes, hechos a la medida de varias distribuciones, también pueden descargarse del sitio Web de KDE en kde.org o directamente del sitio FTP de KDE, en ftp.kde.org y sus sitios espejo en el directorio stable.

La biblioteca Qt

KDE usa como biblioteca de herramientas GUI la biblioteca Qt, desarrollada y mantenida por Trolltech (**trolltech.com**). Qt se considera una de las mejores bibliotecas GUI disponibles para sistemas Unix y Linux. El uso de Qt ofrece la ventaja de que se depende de una biblioteca de

GUI con desarrollo y soporte comercial. Además, el empleo de bibliotecas Qt reduce de manera importante el tiempo de desarrollo para KDE.

Trolltech ofrece las bibliotecas Qt como software de fuente abierta distribuido gratuitamente. Sin embargo, existen ciertas restricciones: las aplicaciones basadas en Qt (KDE) deben ser gratuitas y de fuente abierta, no deben modificarse las bibliotecas Qt. Si desarrolla una aplicación con bibliotecas Qt y quiere venderla, entonces deberá comprar una licencia de Trolltech. En otras palabras, la biblioteca Qt es gratis para las aplicaciones de fuente abierta gratuitas, no para comerciales.

Configuración y administración de acceso con KDE

KDE emplea un conjunto diferente de menús y puntos de acceso distinto de GNOME para acceder herramientas de administración del sistema. También existen varias formas de acceder a las tareas de configuración de KDE, junto con herramientas de administración del sistema KDE, que no están disponibles en GNOME.

- **Centro de control** Se accede a él desde el menú principal Centro de control o la entrada Preferencias, de cualquier menú Ir, en la ventana de administración de archivos. Se trata de una herramienta de configuración de KDE muy completa, mostrando una lista de todos los paneles de configuración de KDE para administrar su escritorio, el administrador de archivos y el sistema, además de herramientas de administración propias de KDE, que pueden usarse en vez de las de GNOME.
- **Sistema** Se accede a él desde la entrada del menú principal Sistema. Se trata de una colección de herramientas del sistema. Aquí también encontrará herramientas de administración de KDE, como KUser para administrar usuarios.
- **Preferencias** Se ingresa a éstas desde la entrada del menú principal Preferencias. Es una colección más pequeña de características de configuración para modificar, por ejemplo, su imagen de inicio de sesión, panel e impresoras.
- **Utilidades** Se entra a esta opción desde el menú principal Utilidades. Aquí encontrará herramientas para tareas específicas como KPilot para handhelds Palm (en Periféricos) y Beagle para búsquedas.

Un motivo de confusión es que Preferencias se usa de manera distinta en el menú principal y el menú Ir del administrador de archivos. En el menú principal se refiere a cualquier colección ad hoc de herramientas de configuración del escritorio, como notificación de correos, mientras el menú Ir, de la ventana del administrador de sistema, se despliega el centro de control, pero en un formato de icono de ventana.

El término *Sistema* también se aplica de manera diferente. En el administrador de archivos despliega recursos de sistema, como su directorio home o los recursos compartidos de red, mientras el menú principal exhibe una lista de herramientas del sistema, como el actualizador de software. Además, en el panel puede agregar un applet Sistema, con una lista de recursos de sistema del administrador de archivos.

Para configurar inicialmente su escritorio, tal vez quiera ejecutar Escritorio, al que se accede desde el menú principal Preferencias. Aquí puede configurar país e idioma, apariencia de escritorio, nivel de características de eye-candy y el tema de escritorio deseado.

Por último, para configurar sólo su escritorio, puede hacer clic con el botón derecho, en cualquier lugar del escritorio, para desplegar un menú emergente con una entrada Configurar escritorio. Al seleccionar esta opción se abrirán los paneles Configurar escritorio. Son los mismos usados en la selección Escritorio, del Centro de control.

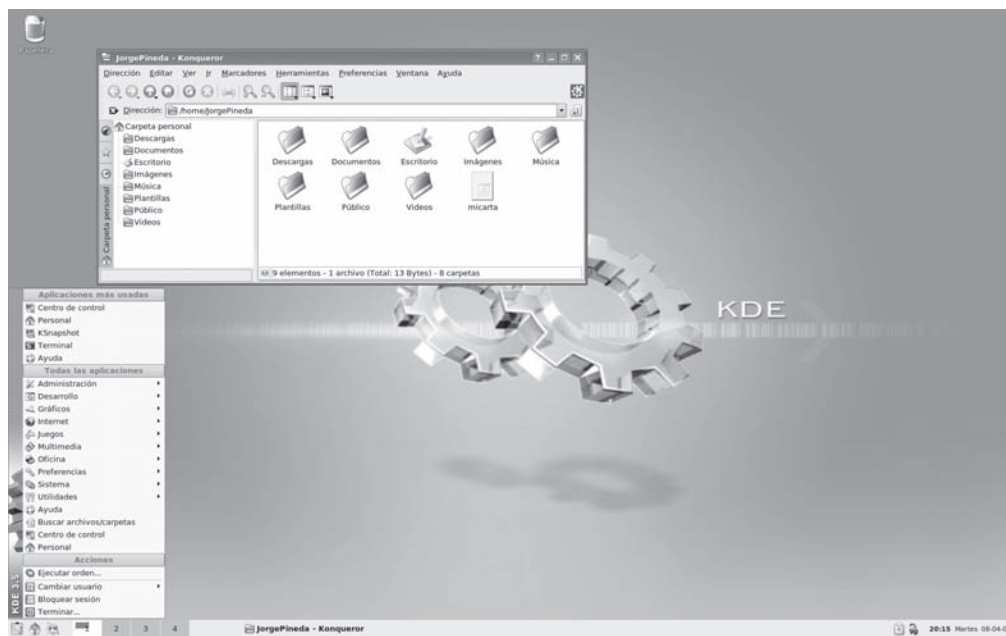


FIGURA 9-1 El escritorio de KDE

El escritorio de KDE

Uno de los objetivos de KDE consiste en proporcionar a los usuarios un escritorio consistente e integrado, donde todas las aplicaciones usan una GUI (véase la figura 9-1). Con este fin, KDE ofrece su propio administrador de ventana (KWM), administrador de archivos (Konqueror), administrador de programas y panel de escritorio (Kicker). Puede ejecutar cualquier otra aplicación compatible con otro X Window System, como Firefox, en KDE, además de cualquier aplicación GNOME. A cambio, es posible ejecutar cualquier aplicación de KDE, incluido el administrador de archivos Konqueror, en GNOME.

NOTA Cuando inicia KDE por vez primera, se le pedirá configure el monitor de dispositivos de red Knemo. Aquí puede especificar sus dispositivos de red, como `eth0` para el primer dispositivo Ethernet, `ppp0` para conexión por marcado telefónico o `wlan0` para conexiones inalámbricas. También especificar los recuadros con información sobre herramientas que habrán de utilizarse, icono para monitoreo y hasta color.

Menús de KDE

Cuando inicia KDE, el panel KDE se despliega en la sección inferior de la pantalla. Hay iconos para menús y programas localizados en el panel, además de botones para diferentes pantallas de escritorio. El botón para el menú principal muestra una *k*, el icono de KDE. Es el botón del menú principal de KDE. Haga clic en este para desplegar el menú de aplicaciones que puede ejecutar (también lo abre al oprimir ALT-F1). Desde el menú de KDE, se accede a varios submenús de diferentes tipos de aplicación. El menú incluye elementos como Terminar, para salir de la sesión de KDE; Bloquear sesión, para bloquear su escritorio; Centro de control, para configurar su escritorio KDE; Cambiar usuario, para iniciar sesión como otro usuario sin salir de la sesión; Ejecutar orden, para correr

programas desde la línea de comandos; Personal, para explorar de manera rápida su directorio home y Ayuda, para iniciar la herramienta de ayuda de KDE.

SUGERENCIA Ejecute Escritorio en el menú Preferencias, para cambiar las opciones de su escritorio de manera sencilla.

Se tiene acceso a las aplicaciones estándar de KDE instaladas a través de este menú. El menú principal tiene casi todas las entradas encontradas en GNOME. Estas se han estandarizado para ambas interfaces. Encontrará entradas para categorías como Internet, Herramientas del sistema, Gráficos y Oficina. Estos menús muestran una lista de aplicaciones que puede usar en GNOME y KDE. Sin embargo, algunos menús de KDE contienen entradas para aplicaciones alternas de KDE, como KMail en el menú Internet. Algunas entradas invocarán la versión KDE de una herramienta, como Terminal en el menú Sistema, que llamará la ventana de terminal de KDE, Konsole. Además, el menú Preferencias está casi vacío; sólo incluye el mismo submenú Más preferencias. En GNOME, el menú Preferencias se utiliza específicamente para configurar GNOME. Para configurar KDE, recurra al Centro de control de KDE referido por el elemento Centro de control, en el menú principal.

SUGERENCIA Si los iconos de sus dispositivos CD o DVD-ROM no se despliegan cuando inserta uno, deberá habilitar el despliegue del icono del dispositivo en su escritorio. Haga clic con el botón derecho en el escritorio y seleccione Configurar escritorio del menú emergente. Esto mostrará sólo aquellas entradas de escritorio del Centro de control. Seleccione Comportamiento y luego, en el panel Iconos de dispositivo, elija la casilla de verificación Mostrar iconos de dispositivos. Se desplegará una larga lista de dispositivos que pueden conectarse; en ella, los dispositivos predeterminados ya estarán seleccionados. Puede seleccionar y dejar de seleccionar los que quiera mostrar u ocultar. Para casi todos los dispositivos, tiene las opciones montada y desmontada. Por ejemplo, una entrada desmontada para DVD-ROM desplegará un icono de DVD-ROM aunque el dispositivo esté vacío.

Salida de KDE

Para salir de KDE, elija Terminar, en el menú principal o haga clic con el botón derecho en cualquier lugar del escritorio y seleccione la entrada Terminar, del menú emergente. Si deja cualquier aplicación o ventana KDE o X11 abierta cuando sale, se restaurarán automáticamente al iniciar de nuevo. Si sólo quiere bloquear su escritorio, seleccione la entrada Bloquear escritorio, en el menú principal y aparecerá el protector de pantalla. Para acceder a un escritorio bloqueado, haga clic en la pantalla; aparecerá un cuadro pidiéndole contraseña de inicio de sesión. Cuando escriba la contraseña, su escritorio aparece de nuevo.

NOTA Puede usar el menú Crear nuevo, para generar nuevas carpetas o archivos en su escritorio, además de vínculos con aplicaciones y dispositivos.

Operaciones de escritorio de KDE

Inicialmente el icono Papelera se muestra en el extremo izquierdo. Este icono opera como la Papelera de reciclaje en Windows o Mac. Arrastre elementos a la papelera para almacenarlos con miras a su eliminación. Puede utilizar los iconos de disco flexible, DVD y CD-ROM para montar, desmontar y desplegar el contenido de sus discos CD-ROM y flexibles. Los iconos DVD y CD-ROM aparecerán cuando se inserten y monten discos, asimismo desaparecerán cuando se expulsen. Se tiene acceso a su directorio home inicialmente desde Personal, en el menú principal o un icono en el panel. Para colocar esta carpeta en el escritorio, haga clic con el botón derecho en la entrada

Personal, el menú principal y elija Añadir un elemento al escritorio del menú emergente. Un icono del directorio home aparece permanentemente en su escritorio. También tiene opción de desplegar el directorio home en su panel, al seleccionar Añadir un elemento al panel principal.

El panel KDE, desplegado a lo largo de la sección inferior de la pantalla, muestra pequeños botones para el menú principal de KDE, directorio home del usuario, explorador Web, herramientas de oficina, reloj y botones para escritorios virtuales, entre otros. El escritorio soporta operaciones de arrastre y colocación. Por ejemplo, para imprimir un documento, basta arrastrarlo al icono Impresora. Puede colocar cualquier directorio en el escritorio, arrastrándolo de una ventana del administrador de archivos al escritorio. Aparecerá un pequeño menú con opciones para copiar o vincular la carpeta. Para crear un icono en el escritorio para la misma carpeta, elija la entrada vínculo.

El escritorio también permite operaciones de copiado y pegado, almacenando el texto copiado de una aplicación en un portapapeles del escritorio, para después pegarlo en otra aplicación. Incluso es posible copiar y pegar desde una ventana Konsole. Por ejemplo, puede copiar la dirección de una página Web y luego pegarla en un mensaje de correo electrónico o documento de procesador de palabras. Esta característica es soportada por la utilería Klipper, localizada en el panel.

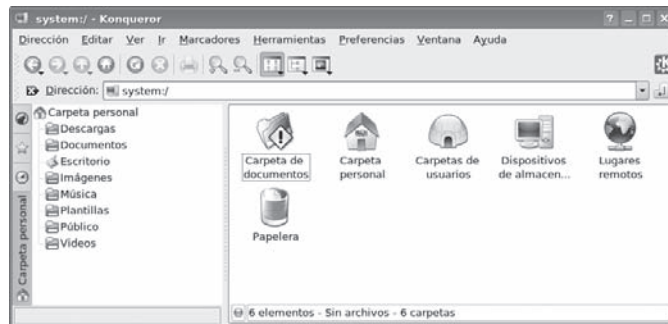
Para crear nuevos directorios en el escritorio, haga clic con el botón derecho en cualquier lugar del escritorio y seleccione Crear nuevo y después Carpeta, del menú emergente. Todos los elementos en el escritorio se localizan en el directorio **Escritorio**, de su directorio home. Ahí encontrará el directorio **Trash**, junto con otros que coloque en su escritorio. También se crean archivos de texto simples y archivos HTML utilizando el mismo menú.

Acceso a los recursos del sistema desde el administrador de archivos

Se tiene acceso a los recursos del sistema, como recursos compartidos de red, directorios de usuario o medios de almacenamiento como CD-ROM, desde cualquier administrador de ventana (véase la figura 9-2). Algunos de estos recursos pueden abrirse directamente, como los iconos Papelera y CD-ROM en su escritorio o la entrada del directorio Personal, en el menú principal. Para abrir una ventana del administrador de archivos inicial, elija la entrada Personal en el menú principal. Después haga clic en el icono del sistema, en la esquina inferior izquierda de la ventana del administrador de archivo (el último icono). Los iconos se despliegan para acceder diferentes recursos del sistema, como medios de almacenamiento, hosts remotos, recursos compartidos de red de Samba y su carpeta Papelera. También puede acceder cualquiera de éstos directamente desde el menú Ir, del administrador de archivo. El icono Dispositivos de almacenamiento se expandirá para mostrar una lista de todos sus medios de DVD y CD, medios extraíbles y particiones de discos duros. Bajo Lugares remotos encontrará iconos para su red local, servicios de red, recursos compartidos de Samba y una herramienta para agregar carpetas de red. Mediante el icono Comparticiones Samba, accede a sus carpetas e impresoras compartidas de Windows. El icono

FIGURA 9-2

Acceso de recursos del sistema desde el administrador de archivos



Dispositivos de almacenamiento muestra una lista de medios de almacenamiento, como sus CD-ROM. Puede abrir éstos para acceder contenidos. Ciertos recursos tienen sus propios URL, que puede escribir en un cuadro de lugares del administrador de archivos para acceder a ellos directamente; por ejemplo, Lugares remotos tiene el URL **remote:/**, Samba utiliza **smb:/** y Dispositivos de almacenamiento utiliza **media:/**.

Configuración de su escritorio

Para configurar su escritorio, haga clic con el botón derecho en el escritorio y seleccione la entrada Configurar escritorio. Se desplegará una ventana con entradas para Comportamiento, Escritorios múltiples, Pantalla, Fondo y Salvapantallas. Todas estas características pueden configurarse usando los paneles Apariencia & Temas, del Centro de control KDE:

- Pantalla le permite configurar resolución y orientación de la pantalla.
- Comportamiento activa el despliegue de ciertas características, como desplegar un menú de escritorio a lo largo de la parte superior de la pantalla o mostrar iconos en su escritorio. También puede seleccionar operaciones para hacer clic en su escritorio. El clic con el botón derecho, actualmente despliega el menú del escritorio. También puede especificar qué dispositivos desplegar en su escritorio.
- El panel Escritorios múltiples elige número de escritorios virtuales a ser desplegados.
- Fondo selecciona color de fondo o imagen para cada escritorio virtual.
- Salvapantallas elige protector de pantalla, así como tiempo de activación. Varios protectores de pantalla ya están configurados.

En el caso de su escritorio, también puede elegir temas diferentes. Un *tema* cambia la apariencia de su escritorio, afectando el aspecto de los elementos GUI, como las barras de desplazamiento, botones e iconos. Por ejemplo, use el tema Mac Os para que su K Desktop parezca escritorio Macintosh. Emplee el Gestor de temas en el Centro de control KDE (Apariencia y Temas), para elegir un tema e instalar nuevos. Es posible que ya estén instalados varios, incluidos Bluecurve (Fedora) o Default (el tema de KDE). Temas adicionales para K Desktop pueden descargarse del sitio Web **kde-look.org**.

Archivos de vínculos y ubicaciones URL del escritorio

En el escritorio KDE, se usan archivos especiales, llamados archivos de *vínculo*, para acceder diversos elementos, incluidos sitios Web, programas de aplicaciones y dispositivos. Cree un archivo de vínculo haciendo clic con el botón derecho en el escritorio y elija Crear nuevo. Desde este menú, escoja el tipo de archivo de vínculo que quiera crear.

La entrada Enlace a aplicación lanza aplicaciones. La entrada Enlace a dirección (URL) almacena una dirección URL que puede usar para acceder a un sitio Web o FTP. El submenú Enlazar a dispositivo crea vínculos con diferentes tipos de dispositivos, incluidos CD-ROM, discos duros y cámaras. Tenga en cuenta que estos sólo son vínculos. Rara vez necesita utilizarlos. Ahora, udev y HAL generan de manera automática y directa los iconos de dispositivo desplegados en su escritorio, conforme se necesiten.

Para crear un archivo de escritorio de URL, haga clic con el botón derecho, seleccione el menú Crear nuevo y el submenú Archivo. Después indique la entrada Vincular a dirección (URL). Aparecerá una ventana mostrando un cuadro pidiéndole inserte un nombre para el archivo y dirección URL. Asegúrese de colocar el protocolo apropiado antes del URL, como **http://** para páginas Web. Como

opción, arrastre y coloque una URL, directamente desde el cuadro Dirección, en un explorador Web, como Firefox. Después edite el archivo del escritorio haciendo clic con el botón derecho en éste y seleccione Propiedades. Se desplegará un cuadro de diálogo de escritorio para acceso a URL. Este cuadro de diálogo tiene tres paneles con fichas: General, Permisos y URL. En el panel General está el nombre de su archivo de escritorio, que será el escrito. Un botón Icono, en este panel mostrará el que aparecerá en su escritorio para ese archivo. Para seleccionar el icono, haga clic en el botón Icono, a fin de abrir una ventana mostrando una lista de iconos entre los que puede seleccionar. Haga clic en Aceptar, cuando termine. Entonces el archivo de escritorio aparecerá en su escritorio con ese icono. En el panel URL, verá un cuadro etiquetado URL, con una dirección ya escrita. Cámbiela si así lo desea. Por ejemplo, en el caso de temas en línea, la URL sería <http://www.kde-look.org>.

En su escritorio, puede hacer clic en el icono URL cuando quiera acceder a ese sitio Web. Una opción y manera más sencilla de crear un archivo de escritorio URL, consiste en arrastrar a su escritorio la URL de una página Web, desplegada en su administrador de archivos. Una ventana emergente le permitirá elegir Copiar o Enlazar. Seleccione Enlazar para crear un archivo de escritorio URL (Copiar creará una copia local de esa página). Un archivo de escritorio se genera automáticamente con esa URL. Para cambiar el icono predeterminado, haga clic con el botón derecho en el archivo y elija Propiedades, para desplegar el cuadro de diálogo del escritorio.

Ventanas de KDE

Tiene la misma función que encontrará en otros administradores de ventanas y escritorios. Puede cambiar el tamaño de una ventana haciendo clic y arrastrando cualquiera de sus esquinas. La operación de clic y arrastre hacia un lado extiende la ventana en esa dimensión, mientras si la aplica en una esquina, se extenderán alto y ancho simultáneamente. Observe que las esquinas están un poco mejoradas. La parte superior de la ventana tiene una barra de título mostrando el nombre de la ventana, nombre del programa, en el caso de aplicaciones, y nombre del directorio actual, para ventanas del administrador de archivos. La ventana activa tiene la barra de título resaltada. Para mover la ventana, haga clic en su barra de título y arrástrela adonde quiera. Haga clic con el botón derecho en la barra de título de la ventana, para desplegar un menú emergente con entradas para operaciones de ventana, como cerrar y cambiar el tamaño de la ventana. Dentro de ésta, se despliegan menús, iconos y barras de herramientas para la aplicación particular.

Puede configurar apariencia y operación de una ventana, seleccionando la entrada Configurar comportamiento de la ventana, en el menú Ventana (haga clic con el botón derecho en la barra de título). Aquí puede configurar apariencia (Decoración de ventana); operaciones de botón y teclas (Acciones); directiva de enfoque, como un clic del ratón o sólo pasar el ratón sobre él (Foco); cómo se despliega la ventana cuando se mueve (Moviendo); y características avanzadas, como desplazar una ventana directamente a otro escritorio virtual (Avanzado). Todas estas características se configuran en los paneles Apariencia y Temas, del Centro de control KDE.

Las ventanas abiertas se muestran como botones en la barra de herramientas de KDE, localizada en el panel. La barra de tareas muestra diferentes programas en ejecución o ventanas abiertas. Se trata, en esencia, de un mecanismo de anclado para cambiar una ventana o aplicación con sólo hacer clic en su botón. Cuando minimiza (convierte en un icono) una ventana, se reduce a un botón de la barra de tareas. Esta se restaura haciendo clic en su botón de la barra de tareas.

A la derecha de la barra de título se encuentran tres botones pequeños para minimizar, maximizar y cerrar la ventana. Cambie a una ventana cuando lo desee, haciendo clic en su botón de la barra de tareas. Desde el teclado, la combinación de teclas ALT-TAB despliega una lista de aplicaciones actuales. Para recorrer la lista, deje oprimida la tecla ALT y luego oprima TAB.

Algunas ventanas de aplicación muestran un botón Ayuda, junto al botón convertido en icono y presentando un signo de interrogación. Al hacer clic en este botón, su cursor toma la forma de un

signo de interrogación. Entonces mueva el cursor a un elemento, un icono en la barra de herramientas y haga clic en éste, para desplegar una pequeña nota de ayuda explicando qué hace el elemento. Por ejemplo, al mover el ratón al botón Adelante, en la barra de herramientas del administrador de archivos, se desplegará una nota explicando que el botón realiza una operación de exploración hacia delante.

SUGERENCIA Barra de tareas y paginador tienen tres estilos: elegante, clásico y transparente.

Escritorios virtuales: Paginador de escritorio KDE

KDE, como casi todos los administradores de ventanas de Linux, permite escritorios virtuales. En efecto, esto extiende el área del escritorio en que trabaja. Puede tener Mozilla ejecutándose en un escritorio y usar un editor de texto en otro. KDE soporta 16 escritorios virtuales, aunque la opción predeterminada es 4. Sus escritorios virtuales se despliegan y tiene acceso a ellos usando Paginador de escritorio KDE, localizado en el panel. Paginador de escritorio KDE representa su escritorio virtual, como una pantalla miniatura mostrando pequeños cuadros para cada escritorio. Está hecho para asemejarse al Selector de áreas de trabajo de GNOME. Los cuatro cuadros predeterminados están numerados 1, 2, 3 y 4. Para pasar de un escritorio a otro, haga clic en el cuadro para el escritorio destino. Al hacer clic en 3, se despliega el tercer escritorio y clic en 1 lo devuelve al primer escritorio. Si quiere mover una ventana a un escritorio diferente, primero abra el menú de ventanas, haciendo clic con el botón derecho en la barra de título. Después seleccione la entrada Al escritorio, que mostrará una lista de escritorios disponibles. Elija el deseado.

También es posible configurar KDE de manera que, si mueve el ratón sobre el borde de una pantalla de escritorio, pasará automáticamente al escritorio contiguo. Tal vez necesite imaginar los escritorios están ordenados en una configuración de cuatro cuadros, con dos escritorios en la parte superior, juntos, y dos escritorios bajo ellos. Esta configuración se habilita en Bordes de escritorio activos, en el panel Escritorio | Comportamiento de ventana | Avanzado, en el Centro de control KDE.

Para cambiar el número de escritorios virtuales, use la entrada Escritorio, del Centro de control KDE. Elija Configurar escritorio, en el menú emergente del escritorio (haga clic con el botón derecho en cualquier lugar del escritorio) y seleccione Escritorios múltiples o elija Centro de control, del menú principal, y abra Escritorio, para seleccionar Escritorios múltiples. La barra visible controla número de escritorios. Desplace esta a la derecha para aumentar cantidad o a la izquierda para reducirla. Puede cambiar cualquiera de los nombres de escritorio haciendo clic en el nombre e insertar uno nuevo. En la entrada Fondo de Aspecto y Temas, puede cambiar la apariencia de determinados escritorios, como color del fondo y fondo (primero quite la selección de Todos los escritorios).

SUGERENCIA Utilice las teclas `CTRL-TAB` para pasar al siguiente escritorio y `CTRL-MAYÚS-TAB` para regresar al escritorio anterior. Use la tecla `CTRL`, en combinación con una tecla de función para ir a un escritorio específico; por ejemplo, `CTRL-F1` cambia al primer escritorio y `CTRL-F3` cambia al tercero.

El panel KDE: Kicker

El panel KDE (Kicker), ubicado en la parte inferior de la pantalla, accede a casi todas las funciones de KDE (véase la figura 9-3). El panel incluye iconos para menús, ventanas de directorio, programas específicos y escritorios virtuales. En el extremo izquierdo del panel, se encuentra un botón para el menú principal (también conocido como menú K), un icono K de KDE.



FIGURA 9-3 El panel KDE

Para agregar una aplicación al panel, haga clic con el botón derecho, en cualquier lugar de éste y seleccione **Añadir**, del menú emergente. **Añadir** despliega el tipo de objetos que puede agregar, incluidos applets, aplicaciones y extensiones de panel. En el caso de aplicaciones de KDE, elija la entrada **Aplicaciones**. En esta lista se muestran todas las aplicaciones instaladas de KDE en su menú principal. Haga clic en el nombre de la aplicación, para agregar un botón de aplicación al panel. También puede arrastrar aplicaciones desde una ventana de administrador de archivos o el menú principal hasta el panel, directamente, y hacer que se coloquen en el panel automáticamente. El panel sólo despliega archivos de escritorio. Cuando arrastra y coloca un archivo en el panel, se genera automáticamente un archivo de escritorio.

Kicker también soporta varias applets y extensiones de panel, además de botones especiales.

- Las applets están diseñadas para ejecutarse como iconos en el panel. Incluyen reloj, paginador y monitor de sistema.
- Las extensiones de panel agregan componentes a su escritorio (seleccione **Panel**, en el menú **Añadir**). Por ejemplo, la extensión **KasBar** configura su propio panel y listas de iconos para cada ventana abierta. También puede ir fácilmente de una ventana a otra haciendo clic en el icono correspondiente del panel de la extensión **KasBar**.
- Entre los botones especiales se incluyen los de uso para operaciones específicas de KDE, como la lista **KDE Window**, una ventana de terminal **Kterm**, el administrador de impresora **KDE** y preferencias de KDE.

Para configurar posición y comportamiento del panel, haga clic con el botón derecho en el panel y escoja la entrada **Configurar Panel**. Esto despliega una ventana personalizada del módulo de control recolectando las entradas de configuración del panel desde el Centro de Control KDE. Existen cinco ventanas de configuración. Las primeras cuatro le permiten determinar la manera en que se despliega el panel, y la última, **Barra de tareas**, configura la manera en que se muestran las ventanas en la barra de tareas. Éstas se ajustan a las entradas **Escritorio** | **Paneles** | **Barra de tareas** del Centro de control KDE.

Los primeros cuatro paneles son **Arreglo**, **Ocultar**, **Menús** y **Aspecto**. **Arreglo** especifica los bordes de la pantalla y dónde quiere se desplieguen panel y barra de tareas. También puede reducir y agrandar su tamaño. El panel **Ocultar** elige el modo de ocultamiento, para permitir ocultamiento automático o manual y desplegar la barra de tareas. El panel **Menús** controla tamaño de sus menús, y si quiere desplegar documentos abiertos recientemente, como elementos de menú. También puede elegir entradas predeterminadas como **Preferencias** y **Marcadores**, además de editar el menú **K** directamente, agregando o eliminando elementos. El panel **Aspecto** configura colores de los botones e imágenes de fondo en la barra de tareas. Con el panel **Barra de tareas**, se controlan ventanas y tareas desplegadas en la barra de tareas, además de configurar las acciones de botón.

El centro de ayuda de KDE

Ofrece una interfaz similar a un explorador, para acceder y desplegar archivos de ayuda de KDE, archivos **Man** y de información de Linux. Puede iniciar el Centro de ayuda eligiendo la entrada **Ayuda** en el menú principal (un salvavidas), o al hacer clic con el botón derecho en el escritorio y seleccionar la entrada **Help**. La ventana de **Ayuda** se divide en dos cuadros. El que se encuentra a la izquierda de la pantalla de **Ayuda** almacena paneles con fichas; una muestra contenido y otra ofrece un glosario. El cuadro de la derecha despliega documentos seleccionados actualmente. Un árbol de ayuda en el panel de contenido elige el tipo de documentos de **Ayuda** que quiere acceder. Aquí puede seleccionar manuales, páginas **Man** o documentos de información y manuales de

aplicaciones. El Centro de ayuda incluye un manual de usuario detallado, preguntas más frecuentes y acceso al sitio Web KDE.

Una barra de navegación le permite desplazarse por documentos vistos previamente. Los documentos de Ayuda de KDE usan formato HTML con vínculos en que puede hacer clic para acceder a otros documentos. Los comandos Atrás y Adelante, lo llevan por la lista de documentos vistos previamente. El sistema de Ayuda de KDE ofrece una herramienta de ayuda efectiva, para buscar patrones en documentos de Ayuda, incluidas páginas Man y de información. Seleccione la entrada Buscar, del menú Editar, para desplegar una página donde puede escribir su patrón.

Aplicaciones

Una aplicación en KDE se inicia de varias formas. Si existe entrada en el menú principal, elíjala para iniciar la aplicación. Algunas también tienen botones en el panel KDE; para iniciarlas, basta hacer clic en ellas. Dependiendo de la distribución, el panel almacenará al principio aplicaciones como el explorador Web Firefox y aplicaciones de Office.org. También puede usar el administrador de archivos para localizar un archivo que use dicha aplicación. Haciendo clic en el icono del archivo, se iniciará la aplicación. Otra forma consiste en abrir una ventana shell, escribir el nombre de la aplicación en el indicador de comandos y oprimir ENTER. También puede seleccionar Ejecutar orden, en el menú principal (u oprimir ALT-F2), para abrir una pequeña ventana con un cuadro para insertar un solo comando. Se accede a los comandos anteriores desde un menú emergente. Un botón Opciones mostrará una lista de opciones para programas en ejecución, como prioridad o si se encuentra dentro de una ventana de terminal.

NOTA *Puede crear un archivo de escritorio en su escritorio para cualquier aplicación que ya esté en su menú de KDE, arrastre y coloque la entrada del menú en el escritorio. Seleccione Copiar para crear en el escritorio un archivo para esa aplicación, mostrando su icono.*

Para crear un nuevo archivo de escritorio para una aplicación, haga clic con el botón derecho, en cualquier lugar vacío del escritorio, seleccione Crear nuevo, del menú emergente, y en el submenú Archivo, elija Enlazar aplicación. Inserte el nombre del programa y aparecerá un archivo de escritorio para éste con ese nombre. Luego se abrirá un cuadro de diálogo Propiedades con cuatro paneles: General, Permisos, Aplicación y Previsualización. El panel General desplegará el nombre del vínculo. Para seleccionar la imagen de un icono para el archivo de escritorio, haga clic en el icono. Se mostrará la ventana Seleccionar icono, mostrando una lista de iconos que puede escoger.

En el panel Permisos, asegúrese de asignar permisos para que el programa pueda ejecutarse. Es posible configurar permisos para usted, su grupo o cualquier usuario del sistema. El panel Info Meta, mostrará una lista del tipo de sistema de archivos utilizado.

Para especificar la aplicación que ejecuta el archivo de escritorio, vaya al panel Aplicación y escriba el nombre del programa de la aplicación en el cuadro Orden o haga clic en Examinar y selecciónelo. En este panel, también puede especificar una descripción y comentario. Para la descripción, introduzca el nombre de la aplicación. Éste se utilizará en el vínculo, mediante el administrador de archivos para desplegarlo. El comentario es una nota de ayuda mostrada cuando pasa el ratón sobre el icono.

En el panel Aplicación, también puede especificar tipo de documentos asociados con esta aplicación. El panel inferior muestra los botones Añadir o Eliminar. Para especificar un tipo MIME, haga clic en Añadir. Esto desplegará una lista de tipos de archivos y sus descripciones. Elija la que desee asociar con este programa. No es necesario que los archivos de escritorio residan en el escritorio.

Colóquelos en cualquier directorio y acceda a ellos mediante el administrador de archivos. Después, para cambiar un archivo de escritorio, haga clic con el botón derecho en su icono y seleccione Propiedades, del menú emergente. Esto despliega nuevamente el cuadro de diálogo para este archivo. Puede cambiar su icono e incluso la aplicación que lo ejecuta.

El botón Opciones avanzadas contiene opciones de ejecución para la aplicación, como ejecutarla en una ventana shell o asumiendo una identidad de usuario. Para ejecutar un programa basado en shell, como Vi, ponga una marca en la casilla de verificación Ejecutar en Terminal y especifique la opción de terminal. Las opciones de inicio muestran una lista del programa en la bandeja del sistema.

SUGERENCIA KDE puede desplegar automáticamente directorios seleccionados o lanzar ciertas aplicaciones cuando inicie. Para ello, coloque vínculos con estas ventanas y aplicaciones en el directorio *AutoStart*, localizado en su directorio *.kde*.

Montaje de dispositivos desde el escritorio

Para acceder un CD-ROM, colóquelo en su unidad CD-ROM y haga doble clic en el icono CD-ROM. La ventana del administrador de archivos se abrirá, desplegando el contenido del directorio más alto del CD-ROM. Para expulsarlo, haga clic con el botón derecho en el icono del CD-ROM y seleccione Expulsar, del menú emergente (también puede seleccionar que sólo se desmonte el CD-ROM).

Para acceder unidades USB, inserte una en cualquier puerto USB. La unidad se detectará automáticamente, abriendo una ventana del administrador de archivos, que mostrará el contenido de la unidad. También puede leer, copiar, mover y eliminar archivos de la unidad USB. Un icono de unidad USB aparecerá en su escritorio. Al mover el cursor sobre el icono, se desplegará información detallada acerca de la unidad, dónde está montada y cuánta memoria usa. Al hacer clic con el botón derecho y seleccionar Propiedades, se desplegará el cuadro con información General, Permisos, Info Meta (espacio utilizado) y Montado. El menú de la unidad USB también incluye una entrada para transferir un archivo de imagen a la herramienta digiKam (Desgargar fotos con digiKam). Para quitar una unidad USB, primero haga clic con el botón derecho en el icono de USB y elija la entrada Extracción segura. El icono de la unidad USB desaparecerá del escritorio. Entonces podrá retirar la unidad.

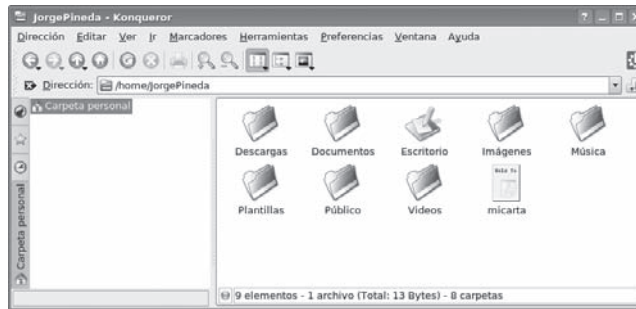
Para acceder un disco flexible, colóquelo en la unidad de disco y haga clic con el botón derecho en el icono Disco floppy. Esto despliega una ventana de administrador de archivos, con el contenido del disco flexible. Tenga cuidado de no quitar el disco antes de desmontarlo. Para esto, haga clic con el botón derecho en el icono y seleccione Desmontar, del menú emergente del icono. Es posible realizar operaciones adicionales en el disco flexible. Si coloca un disco en blanco, puede formatearlo. Seleccione entre varios formatos de sistemas de archivos, incluido MS-DOS. Para formatear un sistema de archivo Linux estándar, elija la entrada ext3.

SUGERENCIA Nunca retire una unidad USB directamente, como hace en Windows. En caso de retirarla, no se aplicará ningún cambio hecho, como agregar archivos. Primero haga clic con el botón derecho en el icono de la unidad USB y seleccione Extracción segura. El icono de la unidad USB desaparecerá del escritorio y entonces podrá quitar la unidad.

Administrador de archivos KDE y cliente de Internet: Konqueror

El administrador de archivos KDE, conocido como Konqueror, es una utilidad con varias funciones usadas para administrar archivos, iniciar programas, explorar Web y descargar archivos de sitios remotos (véase la figura 9-4). Tradicionalmente, el término “administrador de archivos” se usaba

FIGURA 9-4
El administrador de
archivos de KDE



para referirse a la administración de archivos en un disco duro local. El administrador de archivos KDE extiende su funcionalidad más allá, porque interactúa con Internet, desplegando sistemas de archivos remotos como si fueran propios, además de ver páginas Web con las capacidades de un explorador. Puede desplegar gran cantidad de archivos diferentes, incluidos imágenes, PostScript y archivos de texto. Las aplicaciones de KOffice se ejecutan en la ventana de Konqueror. Incluso puede abrir un panel separado dentro de una ventana de administrador de archivos, para ejecutar una terminal, donde escribir comandos de shell (el menú Ventana).

La ventana de Konqueror

Una ventana del administrador de archivos de KDE incluye barra de menús, barra de herramientas de navegación, campo de dirección, barra de estado y barra lateral para diferentes vistas de los recursos de usuarios, como un árbol para ver iconos de archivos y directorios de home. Cuando despliega por primera vez una ventana del administrador de archivos, se muestran los iconos de archivos y subdirectorios de su directorio home. Los archivos y directorios se actualizan automáticamente. Por tanto, si agrega o elimina directorios, no tiene que actualizar manualmente la ventana del administrador de archivos. Su lista se actualiza automáticamente, mostrando archivos añadidos o eliminados. Los archivos mostrados en la lista de un directorio pueden verse de varias formas, como iconos, columnas (iconos pequeños), árboles expansibles, información de archivo o una lista detallada. Las diferentes vistas se muestran en una lista del submenú Modo de vista, en el menú Ver y las más comunes en una lista de iconos al final de la barra de iconos. El modo Árbol presenta una lista de subdirectorios como árboles expansibles, cuyo contenido se despliega haciendo clic en los signos más (+). El modo Lista de información presenta información de archivos, como número de líneas y caracteres en el archivo. La lista detallada indica permisos, propietario, grupo y tamaño de la información. La vista Texto hace lo mismo pero sin desplegar un icono junto al nombre del archivo.

Konqueror también soporta despliegues con fichas. En vez de abrir una carpeta en la misma ventana del administrador de archivos o una nueva, puede abrir una nueva pestaña para usar la misma ventana del administrador de archivos. Una pestaña puede desplegar la carpeta inicial abierta y usar otras pestañas para carpetas abiertas después. Entonces, para ir de una carpeta a otra, basta con hacer clic en la pestaña de la carpeta correspondiente. De esta forma, puede ver varias carpetas en una sola ventana del administrador de archivos. Para abrir una carpeta como una pestaña, haga clic en su icono y seleccione Abrir en una pestaña nueva. Para cerrar una carpeta, haga clic con el botón derecho en la etiqueta de su pestaña y seleccione Cerrar pestaña. También puede desprender una pestaña, abriéndola en su propia ventana del administrador de archivos.

SUGERENCIA Los archivos de configuración, conocidos como archivos ocultos, no suelen desplegarse. Para que un administrador de archivos despliegue estos archivos, seleccione *Mostrar archivos ocultos*, del

menú Ver. Konqueror también da soporte a vistas divididas, permitiéndole ver diferentes directorios en la misma ventana (el menú Ventana). Tiene la opción de dividir de forma vertical u horizontal.

Para abrir un archivo, haga clic en él o selecciónelo y después elija la entrada Abrir, en el menú Dirección. Si quiere seleccionar un archivo o directorio, necesita dejar oprimida la tecla CTRL mientras hace clic o un solo clic (porque al hacerlo doble se abre el archivo). Si el archivo es un programa, se lanzará éste. Si es un archivo de datos, como un archivo de texto, se ejecutará la aplicación asociada utilizando ese archivo de datos. Por ejemplo, si hace doble clic en un archivo de texto, se iniciará la aplicación Kate y desplegará ese archivo. Si Konqueror no puede determinar la aplicación que debe usarse, abrirá un cuadro de diálogo pidiéndole escriba el nombre de la aplicación. Haga clic en el botón Examinar en este cuadro, para usar un árbol de directorios en que podrá localizar el programa de la aplicación deseada.

El administrador de archivos también extrae archiveros tar. Un *archivero* es un archivo con terminación **.tar.gz**, **.tar** o **.tgz**. Al hacer clic en el archivero se muestra una lista de archivos contenidos. Para extraer un archivo determinado, sólo debe arrastrarlo fuera de la ventana. Si hace clic en un archivo de texto en el archivero, se abre con Kate, mientras un archivo de imagen, se despliega con KView. En el caso de distribuciones incluyendo paquetes de software como RPM y DEB, si selecciona el paquete, se abrirá con la utilería de instalación de software de la distribución, usada para instalar el paquete.

Si se trata de una carpeta CVS, para administrar diferentes versiones de un proyecto, puede recurrir a la herramienta Cervisia, mostrada en la lista del submenú Modo de vista, para desplegar y examinar archiveros CVS.

Panel Navegación

El panel Navegación es una barra lateral exhibiendo listas de diferentes recursos a los que puede acceder el usuario con Konqueror. Active o desactive el panel Navegación seleccionando su entrada en el menú Ventana. La barra lateral está configurada según la herramienta panel de Navegación, a la que se accede mediante el primer botón de la barra de botón del panel Navegación.

SUGERENCIA *Konqueror también ofrece una barra lateral de reproductor para archivos de multimedia, seleccionados en la ventana del administrador de archivos.*

El panel Navegación presenta una barra de botones vertical para desplegar elementos, como sus marcadores, dispositivos, el directorio home, servicios y recursos de red, en un árbol expansible. Al arrastrar el ratón sobre el icono del recurso se despliega su nombre completo. Cuando hace clic en un elemento, su icono se expandirá para abarcar el nombre de dicho recurso. Haga doble clic para acceder a éste con Konqueror. Por ejemplo, para ir a un subdirectorio, expanda su entrada del directorio home y después haga doble clic en el subdirectorio deseado. Konqueror desplegará ese subdirectorio. Para ir a un directorio marcado o página Web anterior, encuentre su entrada en la lista Marcadores y selecciónela. El botón de red muestra recursos a los que tiene acceso, como FTP y sitios Web. El botón de la carpeta raíz muestra su directorio y subdirectorios raíz del sistema.

Para configurar el panel Navegación, haga clic en el botón Configurar, en la barra de botones lateral. Seleccione la entrada Vistas múltiples, para permitir el despliegue de varias listas de recursos a la vez, cada uno con su barra lateral secundaria. También puede agregar una nueva lista de recursos, seleccionando desde un marcador, un historial o tipo de directorio. Aparecerá un botón para la nueva lista. Puede hacer clic con el botón derecho para seleccionar un nuevo icono o elegir un URL, ya sea nombre de ruta de directorio o dirección de red. Para eliminar un botón y su lista, haga clic en éste y seleccione la entrada Eliminar.

SUGERENCIA Si la característica de vistas múltiples está habilitada en la configuración del panel de Navegación, puede desplegar varios de estos recursos a la vez, sólo haciendo clic en los que quiera. Si la característica no está habilitada, la lista previa se remplaza al seleccionar una. Deshabilite un despliegue al hacer clic de nuevo en su botón.

Búsqueda

Para buscar archivos, seleccione la entrada Buscar, en el menú Herramientas. Esto abre un panel dentro de la ventana del administrador de archivos, en que puede buscar nombres de archivos utilizando símbolos de relación, como *. Haga clic en Buscar para ejecutar la búsqueda y Detener para detenerla. Los resultados de la búsqueda se despliegan en un panel en la mitad inferior de la ventana del administrador de archivos. Puede hacer clic en un archivo y abrirlo con la aplicación apropiada. El editor de texto Kate despliega archivos de texto. Las imágenes se muestran con KView, mientras los archivos PostScript con KGhostView. Las aplicaciones se ejecutan. El programa de búsqueda también permite guardar resultados de su búsqueda para futura referencia. Tiene incluso la opción de seleccionar archivos de búsqueda y agregarlos a un archivero.

Directorios de navegación

En una ventana de administrador de archivos, al hacer doble clic en un icono de directorio ese directorio se mueve y despliegan sus iconos de archivos y subdirectorios. Para regresar al directorio principal, haga clic en el botón con la flecha hacia arriba, ubicado en la parte izquierda de la barra de herramientas de navegación. Al hacer doble clic en un icono de directorio, desciende por el árbol de directorios, de uno en uno. Para ir directamente a un directorio específico, escriba el nombre de ruta en el cuadro Dirección, localizado arriba del panel desplegando los iconos de archivo y directorio. Como un explorador Web, el administrador de archivos recuerda directorios anteriores desplegado. Use los botones con flechas hacia atrás y adelante para recorrer esta lista de directorios. También puede usar métodos abreviados de teclado para realizar dichas operaciones, como se indica en la lista de la tabla 9-2.

Teclas	Descripción
ALT-FLECHA A LA IZQUIERDA, ALT-FLECHA A LA DERECHA	Va hacia adelante y atrás en el Historial
ALT-FLECHA HACIA ARRIBA	Un directorio hacia arriba
ENTER	Abre un archivo/directorio
ESC	Abre un menú emergente para el archivo actual
FLECHAS HACIA LA IZQUIERDA, DERECHA, ARRIBA Y ABAJO	Se mueve entre iconos
BARRA ESPACIADORA	Selecciona o deja de seleccionar un archivo
AV PÁG, RE PÁG	Se desplaza hacia arriba o abajo de manera rápida
CTRL-C	Copia en el portapapeles el archivo seleccionado
CTRL-V	Pega en el directorio actual archivos en el portapapeles
CTRL-S	Selecciona archivos por patrón
CTRL-L	Abre un nuevo lugar
CTRL-F	Busca archivoss
CTRL-W	Cierra una ventana

TABLA 9-2 Métodos abreviados de teclado del administrador de archivos de KDE

Si sabe que quiere acceder de nuevo directorios particulares, puede convertirlos en marcadores, de la misma forma que hace con una página Web. Sólo abra el directorio y elija la entrada Añadir marcadores, en el menú Marcadores. Entonces, se coloca una entrada para ese directorio en el menú Marcadores del administrador de archivos. Para ir nuevamente al directorio, elija la entrada del directorio en el menú Marcadores. Para ir de un directorio a otro, use el campo Dirección o el árbol de directorio. En el campo Dirección, puede escribir el nombre de ruta de un directorio, si lo conoce y oprimir ENTER. El árbol de directorios presenta una lista en rama de todos los directorios de su sistema y home. Para desplegar el árbol de directorios, seleccione Vista en árbol, del submenú Modo de vista, del menú Ver o haga clic en el icono Vista en árbol en la barra de iconos. Para acceder a la Vista de árbol desde su directorio home o raíz directamente, use los recursos de la carpeta Personal o raíz del panel Navegación.

Operaciones de copiado, movimiento, eliminación, renombre y vinculación

Para realizar una operación en un archivo o directorio, primero debe seleccionarlo. Para escoger un archivo o directorio, haga clic en el icono o lista del archivo. Para seleccionar más de un archivo, mantenga oprimida la tecla CTRL, mientras hace clic en los archivos deseados. Use también las teclas de flechas del teclado, para ir de un icono de archivo a otro y después use la tecla ENTER para seleccionar el archivo deseado.

Para copiar o mover archivos, emplee el método estándar de arrastre y colocación con su ratón. Para copiar un archivo, ubíquelo con el administrador de archivos. Abra otra ventana del administrador de archivos en el directorio donde quiere copiar el archivo. Después haga clic y mueva el icono del archivo a esa ventana. Aparecerá un menú emergente con selecciones para Mover, Copiar o Enlazar. Seleccione Copiar aquí. Para mover un archivo a otro directorio, siga el mismo procedimiento, pero escoja Mover aquí, del menú emergente. Para copiar o mover un directorio, use el mismo procedimiento que para los archivos. Todos los archivos y subdirectorios del directorio también se copian o mueven.

Para cambiar el nombre de un archivo, haga clic en su icono y oprima F2 o haga clic en el icono y seleccione Renombrar, del menú emergente. El nombre bajo el icono se convertirá en un cuadro, con texto que puede modificar.

Un archivo se elimina al borrarlo directamente o colocarlo en la carpeta Papelera para eliminarlo después. Para eliminar un archivo, selecciónelo y después elija la entrada Eliminar, en el menú Editar. O haga clic con el botón derecho en el icono para seleccionar Eliminar. Para colocar un archivo en la carpeta Papelera, haga clic y arrástrelo al icono Papelera en su escritorio o seleccione Mover a la papelera, del menú Editar. Después abra la carpeta Papelera y elimine los archivos. Para eliminar todos los archivos en la carpeta Papelera, haga clic con el botón derecho en el icono Papelera y seleccione Vaciar la papelera, del menú emergente. Para restaurar cualquier archivo de la Papelera, abra ésta y arrástrelo fuera de ella.

Todos los archivos o directorios tienen propiedades asociadas incluidos permisos, nombre de archivo y directorio. Para desplegar la ventana Propiedades de un archivo dado, haga clic con el botón derecho en el icono del archivo y seleccione la entrada Propiedades. En el panel General, se ve el nombre del archivo desplegado. Para cambiar nombre, reemplace éste con uno nuevo. Los permisos se configuran en el panel Permisos. Aquí, se configuran los permisos de escritura, lectura y ejecución para usuario, grupo u otro acceso al archivo. La entrada Grupo cambia el grupo de un archivo. El panel Info Meta muestra una lista de información para tipos de archivo específicos, por ejemplo, el número de líneas y caracteres en un archivo de texto. Un archivo de imagen mostrará una lista con características como resolución, profundidad de bits y color.

SUGERENCIA KDE busca automáticamente y lee un archivo **.directory** localizado en un directorio. Éste almacena información de configuración de KDE, usada para determinar la forma en que se desplegará el directorio. Puede crear este tipo de archivos en un directorio y colocar en él una configuración para establecer características de despliegue, como el icono que usará para desplegar la carpeta del directorio.

Acceso Web y FTP

El administrador de archivos KDE funge como explorador Web y cliente FTP con gran cantidad de características. Incluye un cuadro para escribir el nombre de ruta de un archivo local o URL para una página Web en Internet o intranet. Una barra de herramientas de navegación se usa para desplegar páginas Web o directorios previos. El botón Navegar hasta la URL de inicio siempre lo devolverá al directorio de inicio. Cuando accede una página Web, la página se despliega como en cualquier explorador Web. Con la barra de herramientas de navegación, puede moverse hacia adelante o atrás, a través de la lista de páginas desplegadas previamente en esa sesión.

El administrador de archivos de KDE también opera como cliente FTP. Cuando accede al sitio FTP, navega por directorios remotos como si fueran suyos. Las operaciones para descargar un archivo son iguales a copiar un archivo en su sistema local. Sólo escoja el icono del archivo o entrada en la ventana del administrador de archivos y arrástrelo a otra ventana mostrando un directorio local donde quiere se descargue. Después, seleccione la entrada Copiar, del menú emergente. Konqueror también incluye KSSL, que ofrece soporte SSL completo para conexiones seguras, presentando un despliegue del estado de la conexión segura.

SUGERENCIA KDE presenta la herramienta KGet para Konqueror, que administra descargas FTP, para seleccionar, poner en cola, suspender y programar descargas, conforme se despliega información de estado en las descargas actuales.

Configuración de Konqueror

Como explorador de archivos, explorador Web, FTP y parte integral del escritorio de KDE, Konqueror tiene gran cantidad de opciones de configuración. Para modificar Konqueror, abra la ventana Configurar Konqueror, del menú Preferencias en una ventana de Konqueror (véase la figura 9-5). La ventana despliega presentando una lista de categorías en una barra lateral. Las categorías iniciales trabajan con opciones de administración de archivos básica: apariencia,

comportamiento, vista previa y asociaciones de archivos. En Comportamiento, se especifican acciones como desplegar información sobre herramientas y abrir carpetas en nuevas ventanas. En Aspecto selecciona la fuente y su tamaño. Con Previsualización y Meta-datos configura tamaño de los iconos de vista previa, además de especificar

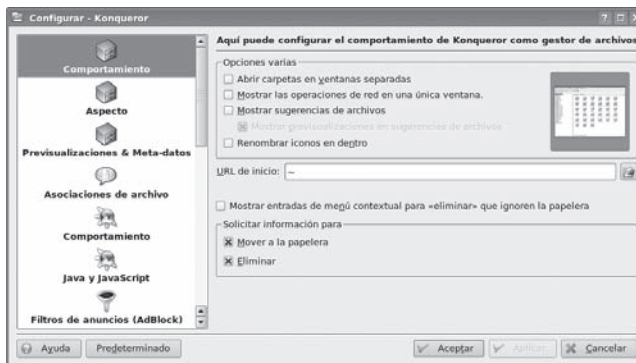


FIGURA 9-5
La ventana Configurar
Konqueror

tipo de archivos de los que quiere recuperar metadatos. Asociaciones de archivo modifica aplicaciones predeterminadas para diferentes tipos de archivos (igual que Asociación de archivo, en Componentes de KDE, en el Centro de control).

Las demás categorías trabajan con parámetros del explorador Web, incluidas configuración de proxies y despliegues de páginas Web, además del comportamiento básico, como resaltar URL, fuentes que se utilizarán, administración de cookies y selección de métodos de codificación. La categoría Historial especifica número de elementos del historial y fecha de expiración. Con la categoría Complementos puede ver una lista de complementos de explorador actual, además de rastrear nuevos.

Configuración de KDE: Centro de control KDE

Con el Centro de control KDE, puede configurar escritorio y sistema, cambiando la forma en que se despliega y características soportadas (véase la figura 9-6). El centro de control puede iniciarse directamente seleccionando Centro de control, en el menú principal.

La ventana del Centro de control se divide en dos paneles. El panel de la izquierda muestra una vista de árbol de todos los componentes a ser configurados, mientras el panel de la izquierda despliega ventanas de diálogo de los componentes seleccionados. Consulte el visor de ayuda para una lista actual de módulos de configuración de K Desktop.

En el panel de la izquierda, los componentes se ordenan en categorías cuyos títulos puede expandir o encoger. El encabezado Internet y Red almacena entradas para configurar herramientas de red del administrador de archivos de KDE, incluidas características de explorador Web, acceso de Samba (Windows) y conectividad inalámbrica. Bajo Aspecto y temas, puede configurar diferentes características para desplegar y controlar su escritorio. Por ejemplo, la entrada Fondo selecciona un color o imagen diferente de fondo para cada uno de sus escritorios virtuales. Otras entradas modifican componentes como el protector de pantallas, lenguaje utilizado y estilo de ventana. El encabezado Periféricos almacena entradas para configurar su ratón, teclado e impresora. El encabezado Sonidos y multimedia contiene paneles para configurar componentes de sonido. Desde el Centro de control, también se accede a un conjunto de herramientas de configuración de sistema KDE especializadas. Actualmente incluyen un administrador de inicio de sesión y un administrador de fuentes.

La categoría Componentes de KDE cambia el comportamiento de su interfaz KDE. El Selector de componentes selecciona componentes predeterminados para aplicaciones, incluidos cliente de correo, herramienta de terminal y explorador Web. Asociaciones de archivo vincula tipos MIME con aplicaciones predeterminadas. La entrada Gestor de archivos establece características del administrador de

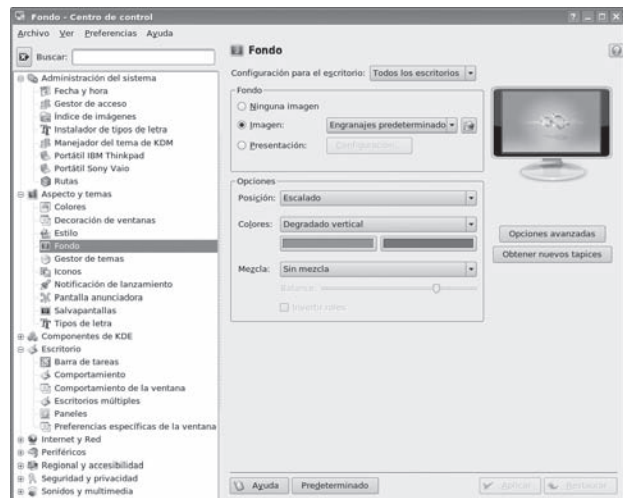


FIGURA 9-6
Centro de control KDE

archivos: fuente utilizada y archivos para vista previa. Con Gestor de sesiones, establece acciones de inicio de sesión y apagado, por ejemplo, restaurar sesiones previas al inicio o apagar el sistema automáticamente, cuando sale de KDE. El Gestor de servicios mostrará una lista de daemons de KDE, cargados bajo pedido y al inicio. También puede seleccionar si quiere un daemon se ejecute al iniciar, además de iniciar o detener manualmente los daemons. Actualmente, los daemons para compartir archivos de KDE e Internet se inician automáticamente, pero puede seleccionar se apaguen e inicien manualmente, cuando quiera ese tipo de conectividad.

También puede acceder entradas del Centro de control desde cualquier ventana del administrador de archivos (véase la figura 9-7). Seleccione Preferencias, desde el menú Ir, del administrador de archivos. Esto abre una carpeta mostrando una lista de iconos para todas las categorías de configuraciones de KDE, como iconos y carpetas. La configuración KDE usa **the URL settings:/**.

Los directorios .kde y Escritorio del usuario

Su directorio **.kde** almacena archivos y directorios para mantener su escritorio KDE. Al igual que GNOME, el directorio Escritorio almacena archivos de escritorio de KDE cuyos iconos se despliegan en el escritorio. Los archivos de configuración se localizan en el directorio **.kde/share/config**. Aquí puede encontrar archivos de configuración general para diferentes componentes de KDE: **kwinrc** contiene comandos de configuración para el administrador de ventanas, **kmailrc** para correo y **kickerrc** para su panel, mientras que **kdeglobals** aloja métodos abreviados de teclado, así como otras definiciones globales. Puede colocar directivas de configuración directamente en cualquiera de estos archivos; **.kde/share/mimelnk** almacena las entradas de menú para archivos de escritorio agregadas por el usuario. El directorio **.kde/share/apps** contiene archivos y directorios para configuración de aplicaciones KDE, incluidos **koffice**, **kmail**, incluso **konqueror**.

Tipos MIME y aplicaciones asociadas

Mientras instala nuevos programas, puede utilizar archivos de cierto tipo. En ese caso, debe registrar el tipo con KDE para asociarlo con una aplicación dada o un grupo de aplicaciones. Por ejemplo, el tipo MIME para imágenes GIF es **image/gif**, asociado con programas para ver imágenes. Puede utilizar el Centro de control de KDE para configurar un nuevo tipo MIME o cambiar las asociaciones del tipo MIME con aplicaciones. Seleccione la entrada Asociación de archivo bajo Componentes de KDE. Esto mostrará una lista de tipos MIME y sus extensiones de nombre de archivo asociadas. Seleccione una entrada para editarla, donde puede cambiar aplicaciones asociadas a ésta. KDE guarda su información tipo MIME en un archivo separado llamado **mimelnk**, en el directorio de configuración KDE.

FIGURA 9-7
Acceso de Preferencias de sistema (Centro de control) desde el administrador de archivos



Directorios y archivos de KDE

Cuando KDE está instalado en su sistema, los archivos de aplicación, configuración y soporte abarcando todo el sistema pueden instalarse en los mismos directorios del sistema que otras GUI y aplicaciones de usuario. En Red Hat Enterprise Linux y Fedora, KDE se instala en el sistema de directorios estándar, con variaciones, como `/usr/bin` para archivos de programas de KDE, `/usr/lib/kde3`, alojando las bibliotecas de KDE, y `/usr/include/kde` contiene archivos de encabezados de KDE empleados para el desarrollo de aplicaciones.

Los directorios localizados en `share` contienen archivos utilizados para configurar opciones predeterminadas de sistema, para el entorno KDE (el directorio del sistema `share` se localiza en `/usr/share`). El directorio `share/mimelnk` correlaciona sus directorios con iconos de KDE, especificando además las definiciones del tipo MIME. El contenido consta de archivos de escritorio con extensión `.desktop`, uno para cada entrada de menú. EL directorio `share/apps` contiene archivos y directorios configurados por aplicaciones de KDE; `share/config` maneja los archivos de configuración para aplicaciones particulares de KDE. Estas configuraciones son las predeterminadas de todo el sistema, prevalecientes sobre configuraciones del usuario en sus propios directorios `.kde/share/config`. El directorio `share/icons` almacena iconos predeterminados utilizados en su escritorio KDE y por aplicaciones KDE, además de la interfaz Bluecurve. Como ya se observó, en el directorio home del usuario, el directorio `.kde` almacena la configuración de KDE propia del usuario, para el escritorio y aplicaciones.

Cada usuario tiene un directorio `Desktop` almacenando archivos de vínculos KDE, para todos los iconos y carpetas en el escritorio del usuario (véase la tabla 9-3). Estos incluyen las carpetas Papelera y vínculos del directorio de CD-ROM y home.

Directorio del sistema KDE	Descripción
<code>/usr/bin</code>	Programas de KDE
<code>/usr/lib/kde3</code>	Bibliotecas de KDE
<code>/usr/include/kde</code>	Archivos de encabezado a usarse para compilar y desarrollar aplicaciones de KDE
<code>/usr/share/config</code>	Archivos de configuración de escritorio y aplicación de KDE
<code>/usr/share/mimelnk</code>	Archivos de escritorio utilizados para construir el menú principal
<code>/usr/share/apps</code>	Archivos utilizados por aplicaciones de KDE
<code>/usr/share/icons</code>	Iconos utilizados en el escritorio y aplicaciones de KDE
<code>/usr/share/doc</code>	Sistema de Ayuda de KDE
Directorio KDE del usuario	Descripción
<code>.kde/AutoStart</code>	Aplicaciones que se inician automáticamente con KDE
<code>.kde/share/config</code>	Archivos de configuración de escritorio y aplicación de KDE del usuario, para características especificadas por el usuario
<code>.kde/share/mimelnk</code>	Archivos de escritorio utilizados para construir las entradas del menú del usuario en el menú principal de KDE
<code>.kde/share/apps</code>	Directorios y archivos usados por aplicaciones de KDE
<code>Desktop</code>	Archivos de escritorios para iconos y carpetas desplegadas en el escritorio de KDE del usuario
<code>Desktop/Trash</code>	Carpeta Papelera para archivos marcados para eliminación

TABLA 9-3 Directorios de instalación de KDE

IV PARTE

Software de Linux

CAPÍTULO 10

Administración de software

CAPÍTULO 11

Aplicaciones de oficina
y bases de datos

CAPÍTULO 12

Herramientas gráficas
y multimedia

CAPÍTULO 13

Clientes de correo y noticias

CAPÍTULO 14

Clientes Web, FTP y Java

CAPÍTULO 15

Herramientas de red

Administración de software

Instalar, desinstalar o actualizar paquetes de software siempre ha sido un proceso simple en Linux, debido al uso extendido de formatos de paquetes como Red Hat Package Manager (RPM) o el administrador de paquetes de Debian (DEB). En vez de usar un archivo tar estándar, el software se empaqueta en un archivo con formato especial que puede manejarse mediante un administrador de paquetes. Un archivero de administrador de paquetes contiene todos los archivos de programas, configuración, de datos e incluso documentación integrando una aplicación de software. Con una simple operación, el administrador de paquetes instala todo esto. También revisa cualquier otro paquete de software que el programa pueda necesitar para ejecutarse correctamente. Incluso, es posible crear sus propios paquetes.

Ahora muchas distribuciones de Linux administran paquetes de software utilizando depósitos en línea. Todo el software se descarga e instala directamente usando instalador y actualizador de software de la distribución. Este método anuncia el cambio de la idea de que la mayor parte del software de Linux se incluya en pocos discos; a ver el disco de instalación como punto de partida desde donde expandir el software instalado, en la manera que desea, recurriendo a depósitos en línea. Con la integración del acceso a depósitos de software desde su sistema Linux, puede pensar ahora en ese software como extensión instalada de manera sencilla en su colección actual. Depender de discos para su software se vuelve, en cierto sentido, obsoleto.

También es posible descargar versiones de código fuente de las aplicaciones y luego compilarlas e instalarlas en su sistema. Aunque este proceso alguna vez fue complejo, ahora se ha vuelto más eficaz con la adición de *secuencias de comandos de configuración*. Casi todo el código fuente actual, incluido el software GNU, se distribuye con una secuencia de comandos de configuración. Ésta detecta automáticamente la configuración de su sistema y genera un *Makefile* (el archivo de secuencia de comandos, producido por el uso del comando *make*) empleado para compilar la aplicación y crear un archivo binario compatible con su sistema. En la mayoría de los casos, con algunas operaciones de *Makefile* puede compilar e instalar código complejo en cualquier sistema.

Puede descargar software de Linux desde cualquier recurso en línea. También encontrará sitios para tipos particulares de aplicaciones, como GNOME y KDE, además de distribuciones en particular. Con depósitos habilitados para distribución, Linux descarga y actualiza automáticamente software instalado desde paquetes de software.

Extensión	Archivo
.rpm	Paquete de software creado con Red Hat Software Package Manager, usado en las distribuciones Fedora, Red Hat, Centos y SuSE
.deb	Paquete de Linux Debian
.src.rpm	Paquetes de software, versiones de código fuente de aplicaciones, creados con Red Hat Software Package Manager
.gz	Archivo gzip -comprimido (use gunzip para descomprimirlo)
.bz2	Archivo bzip2 -comprimido (use bunzip2 para descomprimirlo; use también la opción j con tar , como en xvzf)
.tar	Archivo de archivero tar (use tar con xvf para extraer)
.tar.gz	Archivo gzip -comprimido de archivero tar (use gunzip para descomprimir y tar para extraer; emplee la opción z con tar , como en xvzf , para descomprimir y extraer en un solo paso)
.tar.bz2	Archivo bzip2 -comprimido de archivero tar (extraiga con tar -xvzj)
.tz	Archivo de archivero tar, comprimido con el comando compress
.Z	Archivo comprimido con el comando compress (use el comando decompress para descomprimir)
.bin	Archivo de software que se extrae solo
.torrent	Archivo de BitTorrent, para realizar descargas distribuidas por BitTorrent (sólo información de torrent)

TABLA 10-1 Extensiones de archivo de paquete de software de Linux

Tipos de paquetes de software

Los paquetes de software en sitios RPM como **freshrpms.net** y **rpmfind.net**, tendrán la extensión de archivo **.rpm**. Los paquetes RPM, conteniendo código fuente, manejan la extensión **.src.rpm**. Otros paquetes, en forma de código fuente que necesita compilar, vienen en diversos archivos comprimidos. Generalmente, tienen extensiones **.tar.gz**, **.tgz** o **.tar.bz2**. Se explican con más detalle, adelante en este capítulo. En la Tabla 10-1 se muestran varias extensiones de archivo comunes, que encontrará para gran variedad de paquetes de software de Linux a su disposición.

Descarga de imágenes de distribución ISO y DVD con BitTorrent

Archivos muy grandes, como las imágenes de distribución ISO, pueden descargarse mediante BitTorrent. Se trata de una operación de descarga distribuida, donde varios usuarios en Internet participan en la misma descarga, cada uno subiendo partes que otros pueden descargar, a su vez. El archivo se corta en pequeños paquetes IP y cada paquete se sube y descarga individualmente, como si fuera un archivo separado. Su cliente BitTorrent combinará automáticamente estos paquetes en un archivo completo. No existe espacio en disco compartido, como en métodos para compartir archivos. No se permite acceso para otros usos. Un usuario sólo pide que otros le envíen el paquete. Se trata, estrictamente, de una operación de transmisión, como si diferentes usuarios participaran en la misma transmisión, en vez de uno solo.

Necesitará acudir a un cliente BitTorrent. Tiene varios para escoger: **azureus**, **rtorrent**, **ctorrent**, **ktorrent**, incluido el BitTorrent original. Para el BitTorrent original, existen dos paquetes,

bittorrent y **bittorrent-gui**, con una interfaz GNOME. El sitio de BitTorrent original ahora es un sitio comercial para descargar películas requiriendo el Reproductor Windows Media y administración de derechos digitales (DRM, Digital Rights Managment).

Para iniciar un torrent, basta hacer clic en la entrada del torrent en un archivo desde su explorador Web. Firefox pedirá que decida si quiere iniciar la aplicación directamente o descargar el archivo. Si ejecuta la aplicación, se lanzará el cliente BitTorrent e iniciará su descarga. Puede detener el torrent en cualquier momento e iniciarlo después. Iniciará automáticamente donde se quedó, conservando lo descargado hasta el momento. El cliente BitTorrent automáticamente se ajustará a la escala de descarga y subida apropiada, pero puede ajustar esto como desee. Hay botones para pausar y detener la descarga, además de recuperar información detallada acerca del torrent. Un icono de barra mostrará progreso y tiempo restante estimado, aunque puede acortarse conforme avance la descarga. El cliente mostrará todos los torrents en proceso, mostrando cuánto se ha descargado para cada uno, permitiéndole seleccionar cuál quiere activo. Las entradas de configuración permitirán ajustar comportamiento: qué puerto utilizar, directorio de descarga predeterminado y permitir que los torrents se ejecuten en paralelo.

El archivo torrent por sí solo no es la imagen ISO o DVD. Ésa la descargará BitTorrent. En cambio, es un archivo pequeño de BitTorrent que almacena información sobre la manera de acceder a ese torrent en particular e iniciarlo. Primero puede descargar el archivo torrent y luego iniciarlo para descargar el torrent. Es posible coleccionar archivos torrent para iniciar y detener cuando quiera. Tendrá un archivo torrent pequeño de tipo **.torrent** en su disco. Haga doble clic en éste para iniciar el cliente BitTorrent y la descarga para las imágenes ISO binarias de la distribución.

BitTorrent sirve para cualquier archivo del que encuentre un torrent asociado, pero la descarga es sensible al tiempo, porque depende de cuántos usuarios participen en el torrent. Los torrents de distribuciones Linux suelen tener buen mantenimiento y usuarios contribuyendo al soporte de subida constante, conocidos como semillas. Es posible que otros dejen de existir, si ya no hay usuarios participando en ese torrent. La velocidad de descarga dependerá directamente del número de usuarios concurrendo en el torrent. Por esto el torrent es mejor para las distribuciones en momentos específicos, como el lanzamiento de una nueva versión.

NOTA El paquete BitTorrent también ofrece herramientas para crear su propio torrent y distribuir un archivo. Ahora se dispone de dos métodos de distribución: con tracker y sin él. Un método sin tracker no requiere soporte de un servidor.

Red Hat Package Manager (RPM)

Varias distribuciones de Linux, incluidas Fedora, Red Hat y openSUSE, utilizan RPM para organizar el software de Linux en paquetes que se instalan, actualizan y eliminan automáticamente. RPM es un sistema de administración de paquetes orientado a línea de comandos, para instalar, desinstalar, consultar, verificar y actualizar paquetes de software existentes en su sistema. RPM opera como su propio programa de instalación para una aplicación de software. Una aplicación de software de Linux suele incluir archivos que necesitan instalarse en diferentes directorios. Generalmente, el programa se coloca por sí sólo en un directorio llamado **/usr/bin**; archivos de manuales en línea, como las páginas Man, van en otros directorios, al igual que archivos de bibliotecas. El paquete de software RPM realiza todas estas tareas por sí solo. Además, si después decide que no quiere una aplicación específica, puede desinstalar paquetes, para eliminar todos los archivos e información de configuración de su sistema. RPM trabaja de manera similar al Asistente para instalación de Windows, instalando software automáticamente, incluidos configuración, documentación, imagen, ejemplos y archivos de programas, junto con cualquier otro archivo y aplicación que se pueda

utilizar. Todos se instalan en los directorios apropiados de su sistema. RPM mantiene una base de datos del software instalado, dando seguimiento a todos los archivos instalados. Esto permite el uso de RPM para desinstalar software, eliminando automáticamente todos los archivos de la aplicación.

NOTA Red Hat, Fedora y otras distribuciones de Linux que usan paquetes RPM pueden recurrir a Yum (Yellowdog Updater Modified) para descargar, instalar y actualizar software automáticamente, desde depósitos RPM en línea (linux.duke.edu/projects/yum). Revise la documentación Yum de su distribución para adquirir más información. La instalación y actualización es una operación simple, consistente en apuntar y hacer clic; Yum detectará la versión exacta del paquete que necesita para su sistema. Yum puede utilizarse para cualquier depósito compatible con él.

Para instalar y desinstalar paquetes RPM, emplee el comando **rpm**, directamente desde un indicador de comandos shell. A pesar de que debe descargar paquetes RPM para su distribución particular, varios paquetes de software RPM están diseñados para ejecutarse en cualquier sistema Linux. Aprenderá más acerca de RPM en su sitio Web, en rpm.org y en wiki.rpm.org. Estos sitios contienen versiones y documentación actualizada para RPM.

NOTA RPM ha sido reconocido como un proyecto independiente y ya no se considera sólo una herramienta de Red Hat.

Las convenciones de asignación de nombres para paquetes RPM varían entre una distribución y otra. El nombre del paquete incluye la versión de éste y su plataforma (**i386** para PC Intel), además de la extensión **.rpm**. Aquí se muestra un ejemplo del paquete RPM, del editor Emacs para sistemas Intel:

```
emacs-21.4-3.i386.rpm
```

SUGERENCIA Los paquetes RPM con el término **noarch** indican paquetes de arquitectura independiente. Esto significa que están diseñados para instalarse en cualquier sistema Linux. Es posible que los paquetes sin **noarch** sean dependientes de distribuciones o arquitecturas.

El comando rpm

Con el comando **rpm**, puede mantener paquetes, consultarlos, generar sus propios paquetes y verificar los que tiene. El mantenimiento de paquetes incluye instalación de nuevos paquetes, actualización a nuevas versiones y desinstalación. El comando **rpm** involucra un conjunto de opciones para determinar qué acciones tomar. Además, ciertas tareas, como instalación y consulta de paquetes, tienen opciones propias, que más adelante califican el tipo de acción a tomar. Por ejemplo, la opción **-q** pide un paquete, pero combinado con la opción **-l**, muestra listas de todos los archivos de ese paquete. En la Tabla 10-2 se presenta una lista del conjunto de opciones **rpm**. La sintaxis para el comando **rpm** es la siguiente (*rpm-paquete-nombre*, es el nombre del paquete de software que desea instalar):

```
rpm opciones rpm-paquete-nombre
```

Una descripción completa de **rpm** y sus capacidades se muestran en el manual en línea:

```
# man rpm
```

Modo de operación	Efecto
rpm -i <i>opciones archivo-paquete</i>	Instala un paquete; requiere el nombre completo del archivo del paquete.
rpm -e <i>opciones archivo-paquete</i>	Desinstala (elimina) un paquete; sólo necesita el nombre del paquete, a menudo una palabra.
rpm -q <i>opciones archivo-paquete</i>	Consulta un paquete. Una opción puede ser el nombre del paquete, opciones adicionales y nombre del paquete u opción aplicada a todos los paquetes
rpm -U <i>opciones archivo-paquete</i>	Actualiza; igual a instalar, pero cualquier versión previa se elimina.
rpm -F <i>opciones archivo-paquete</i>	Actualiza, pero sólo si el paquete está instalado.
rpm -verify <i>opciones</i>	Verifica que un paquete esté instalado correctamente; utiliza el mismo tipo de opciones que las consultas. Puede utilizar -v o -y en lugar de -verify .
--percent	Despliega el porcentaje de los paquetes durante la instalación.
--replacepks	Instala un paquete ya instalado.
--replacefiles	Reemplaza los archivos instalados por otros paquetes.
--redhatprovides <i>archivos-dependientes</i>	Busca paquetes dependientes.
--oldfiles	Instala una versión más antigua de un paquete ya está instalado.
--test	Prueba la instalación; no instala, sólo revisa si hay conflictos.
-h	Despliega símbolos # cuando el paquete se instala.
--excludedocs	Excluye archivos de documentación.
--nodeps	Instala sin revisión de dependencias (peligroso).
--force	Impone la instalación, pese a los conflictos (peligroso).
Opciones de desinstalación (se utilizará con -e)	
--test	Prueba la instalación. No elimina, sólo revisa lo que se eliminará.
--nodeps	Desinstala sin revisar dependencias.
--allmatches	Elimina todas las versiones de un paquete.
Opciones de consulta (se utilizará con -q)	
nombre-paquete	Consulta el paquete.
-qa	Consulta todos los paquetes.
-qf <i>nombredearchivo</i>	Consulta un paquete que pertenece a <i>nombredearchivo</i> .

TABLA 10-2 Opciones de Red Hat Package Manager (RPM)

Opciones de consulta (se utilizará con -q)	
-qR	Lista paquetes de que depende el panel.
-qp <i>nombre-paquete</i>	Consulta un paquete desinstalado.
-qi	Despliega toda la información de un paquete.
-ql	Lista archivos en un paquete.
-qd	Lista archivos de documentación en un paquete.
-qc	Lista archivos de configuración en un paquete.
-q --dump	Muestra sólo archivos con detalles completos.
Opciones generales (se utilizará con cualquier opción)	
-vv	Depura; despliega descripciones de todas las acciones tomadas.
--quit	Despliega sólo mensajes de error.
--version	Despliega un número de versión RPM.
--help	Despliega un mensaje de uso detallado.
--root <i>directorio</i>	Usa un directorio como directorio de nivel alto, para todas las operaciones (en vez de raíz).
--dbpath <i>directorio</i>	Utiliza una base de datos RPM en el directorio específico.
--dbpath <i>cmd</i>	Canaliza la salida RPM con el comando cmd .
--rebuilddb	Regenera la base de datos de RPM; puede usarse con las opciones -root y -dbpath .
--initdb	Genera una nueva base de datos de RPM; puede utilizarse con las opciones -root y -dbpath .
Otras fuentes de información	
rpm.org	El sitio Web de RPM con documentación detallada.
RPM Man page (man rpm)	Lista detallada de opciones.

TABLA 10-2 Opciones de Red Hat Package Manager (RPM) (*continuación*)

Consulta de información desde paquetes RMP y software instalado

La opción **-q** indica si un paquete ya está instalado y **-qa** despliega una lista de todos los paquetes instalados. Es mejor canalizar la salida a una utilería de paginador, como **more**.

```
# rpm -qa | more
```

En el siguiente ejemplo, el usuario revisa si Emacs ya está instalado en su sistema. Observe que no es necesario el nombre de archivo completo del archivero RPM. Si el paquete está instalado, su sistema ya ha registrado el nombre y lugar donde se localiza.

```
# rpm -q emacs
emacs-22.0.95-1
```

Opción	Significado
-q aplicación	Revisa si una aplicación está instalada.
-qa aplicación	Lista todas las aplicaciones RPM instaladas.
-qf nombredearchivo	Lista las aplicaciones a que pertenece <i>filename</i> .
-qR aplicación	Lista las aplicaciones de que depende esta aplicación.
-qi aplicación	Despliega toda la información de aplicación.
-ql aplicación	Lista archivos en la aplicación.
-qd aplicación	Lista los archivos de documentación en la aplicación, únicamente.
-qc aplicación	Lista los archivos de configuración en la aplicación, únicamente.

TABLA 10-3 Opciones de consulta para software instalado

Puede combinar la opción **q** con **i** o **l**, para desplegar información del paquete. **-qi** despliega información del software, como número de la versión o autor (**-qpi** consulta un archivo de paquete desinstalado). La opción **ql** lista todos los archivos en el paquete de software. La opción **-h** proporciona una lista completa de opciones **rpm**. Las opciones de consulta comunes se muestran en la tabla 10-3.

Para desplegar información, tomada directamente de un paquete RPM, agregue el calificador **p** a las opciones **q**, como se aprecia en la tabla 10-4. La combinación **-qpi** despliega información de un paquete específico y **-qpl** lista de archivos contenidos en el paquete RPM dado. En este caso, debe especificar nombre de archivo completo del paquete. Para evitar esto, inserte una parte única del nombre y use el carácter comodín para nombres de archivo ***** para generar el resto.

Si su consulta RPM produce una lista de datos larga, como una lista extensa de archivos, puede canalizar la salida al comando **less**, para ver la lista pantalla por pantalla o incluso redirigir la salida a un archivo.

```
# rpm -ql emacs | less
# rpm -qpl emacs-22.0.95-1.i386.rpm > mitemp
```

Opción	Significado
-qpi archivo-RPM	Despliega toda la información del paquete RPM.
-qpl archivo-RPM	Lista archivos en el paquete RPM.
-qpd archivo-RPM	Lista los archivos de documentación en el paquete RPM, únicamente.
-qpc archivo-RPM	Lista los archivos de configuración en el paquete, únicamente.
-qpr archivo-RPM	Lista los paquetes de que depende el paquete RPM.

TABLA 10-4 Opciones de consulta para paquetes RPM

Instalación y actualización de paquetes con rpm

Utilice la opción **-i** para instalar nuevos paquetes y **-U** para actualizar los paquetes instalados actualmente con la nueva versión. Con **-e**, **rpm** desinstala el paquete. Si intenta usar **-i**, para instalar una nueva versión de un paquete instalado, recibirá un error indicando que el paquete ya está instalado. Cuando un paquete está instalado, RPM revisa su firma, utilizando claves públicas insertas desde un proveedor de software. Si la revisión de la firma falla, se despliega un mensaje de error, especificando **NOKEY**, en caso de no tener la clave pública apropiada. Si quiere instalar sobre un paquete ya instalado, puede forzar la instalación con la opción **--replacepks**. Algunas veces un paquete incluirá un archivo, como una biblioteca, también instalado por otro paquete. Para permitir que un paquete sobrescriba el archivo instalado por otro paquete, utilice la opción **--replacefiles**. Muchos paquetes dependen de bibliotecas instaladas por otros. Si estos paquetes dependientes no están instalados, primero debe instalarlos. RPM informa sobre los archivos dependientes faltantes y sugiere la instalación de paquetes. Si no se sugieren paquetes, puede usar la opción **--redhatprovides** para archivos faltantes y buscar los paquetes necesarios.

La opción **-U** también instala un paquete, si no está instalado, mientras la opción **-F** sólo actualiza los paquetes instalados. Si el paquete incluye archivos de configuración que sobrescribirán los instalados, guardará una copia de cada archivo de configuración actual, con un archivo cuya terminación es **.rpmsave**, como **/etc/mttools.conf.rpmsave**. Este archivo preserva cualquier cambio de configuración personalizado hecho al archivo. Asegúrese de revisar también la compatibilidad de configuración entre versiones previas y actualizadas. Si intenta instalar un paquete más antiguo que el ya instalado, entonces debe usar la opción **--oldpackages**.

```
# rpm -Uvh emacs-22.0.95-1.i386.rpm
```

Eliminación de paquetes de software de RPM

Para eliminar un paquete de software desde su sistema, primero utilice **rpm -q**, para asegurarse de que realmente está instalado. Después use la opción **-e**, para desinstalarlo. No es necesario emplear el nombre completo del archivo instalado; sólo necesita el nombre de la aplicación. Por ejemplo, si decide que no necesita Gnumeric, puede eliminarlo con la opción **-e** y el nombre del software, como se muestra aquí:

```
# rpm -e gnumeric
```

RPM: verificación de una instalación RPM

Emplee la opción de verificación (**-V**), para ver si ocurrieron problemas con la instalación. RPM compara los atributos actuales de archivos instalados con la información de éstos, alojada en la base de datos RPM, cuando se instaló el paquete. Mientras no haya discrepancias, RPM nada produce. De otro modo, RPM genera una secuencia de ocho caracteres, uno para cada atributo, para cada archivo del paquete que falla. Los que no difieren tienen un punto. Los que difieren tienen un código de carácter correspondiente, como se indica en la Tabla 10-5.

En el siguiente ejemplo se verifica el paquete ProFTPD:

```
[root@tortuga mispaquetes]# rpm -V proftpd
```

Para comparar los archivos instalados directamente con los que se encuentran en un paquete RPM, utilice la opción **-Vp**, de manera muy parecida a la opción **-qp**. Para revisar todos los paquetes, utilice la opción **-Va**, como se muestra aquí:

Atributo	Explicación
5	Suma de verificación de MD5
S	Tamaño de archivo
L	Vínculo simbólico
T	Fecha de modificación del archivo
D	Dispositivo
U	Usuario
G	Grupo
M	Modo (incluye permisos y tipos de archivo)

TABLA 10-5 Códigos de discrepancia RPM

```
# rpm -Va
```

Si quiere verificar un paquete, pero sólo conoce el nombre de un archivo en éste, puede combinar la verificación con la opción **-f**. En el siguiente ejemplo, se comprueba que el paquete RPM contiene el comando **ftp**:

```
# rpm -Vf /bin/ftp
```

Reconstrucción de la base de datos RPM

RPM mantiene un registro de paquetes instalados en su base de datos **RPM**. Algunas veces, tal vez necesite reconstruir su base de datos para asegurar que RPM cuenta con información actual de qué está instalado y no. Use la opción **--rebuilddb** para reconstruir su archivo base de datos:

```
# rpm --rebuilddb
```

Para crear una nueva base de datos RPM, use la opción **--initdb**. Esta puede combinarse con **--dbpath** para especificar la ubicación de la nueva base de datos.

Debian

Entre la mayor parte de distribuciones Linux, existen básicamente dos empaquetadores de software principales, RPM y DEB, utilizado principalmente en distribuciones Debian y Ubuntu. El formato DEB tiene mayor capacidad que su contraparte RPM. Por ejemplo, un paquete Debian resolverá automáticamente las dependencias, instalando cualquier otro paquete necesario, en vez de sólo reportar su ausencia, como hace RPM. Debian también emplea un formato de nombre de paquete diferente de RPM. El nombre de los paquetes se integra con el nombre del software, número de versión y extensión **.deb**. Consulte debian.org/doc para adquirir más información.

Hay dos administradores básicos de paquetes para usarse con paquetes Debian: Advanced Package Tool (APT) y la herramienta Debian Package (dpkg). Para APT, se utiliza la herramienta **apt-get** en la administración de paquetes. Esta incluso puede descargar paquetes de software, además de versiones de código fuente. La herramienta **apt-get** toma dos argumentos: el comando que se utilizará y nombre del paquete. Otras herramientas del paquete APT siguen el mismo formato. El comando es un término como **install**, para instalar paquetes o **remove**, para desinstalar un paquete. Para instalar el paquete de imagen de kernel usaría:

```
apt-get install kernel-image-2.6.21.deb
```

Actualizar sólo es cuestión de usar el comando **upgrade**. Si no especifica un paquete, apt-get con el comando **upgrade** actualizará todo su sistema, descargando desde un sitio FTP o copiando desde un CD-ROM e instalando los paquetes necesarios. Agregue la opción **-u** para mostrar una lista de paquetes mientras se actualizan.

```
apt-get -u upgrade
```

Incluso puede actualizar a una nueva versión con el comando **dist-upgrade**.

```
apt-get -u dist-upgrade
```

Existen varias “fachadas” populares para apt-get, que administran su software de manera sencilla, como synaptic, gnome-apt, aptitude y dselect. La configuración para APT se almacena en el directorio **/etc/apt**. Aquí, el archivo **sources.list** señala una lista de depósitos de distribución, desde donde se instalan los paquetes. Las listas de fuentes para depósitos de terceros (como Wine) se almacenan en el directorio **/etc/sources.list.d**. Los archivos de bases de datos GPG contienen claves de validación para dichos depósitos. Las opciones específicas para apt-get se almacenan en un archivo **/etc/apt.conf** o varios archivos ubicados en el directorio **/etc/apt.conf.d**.

También puede usar la herramienta **dpkg** para administrar software, aunque se utiliza principalmente para obtener información de un paquete. Su versión más compleja, **dpkg-deb**, se usa para generar paquetes Debian. Los archivos de configuración dpkg se ubican en el directorio **/etc/dpkg**. La configuración se almacena en el archivo **dpkg.cfg** y sus fuentes en el directorio **origins**.

Instalación de software desde archivos comprimidos: .tar.gz

Las aplicaciones de software de Linux, en forma de código fuente, están disponibles en diferentes sitios de Internet. Puede descargar cualquiera e instalarlo en su sistema. Las versiones recientes suelen estar disponibles en forma de archivos comprimidos, si no tienen versión RPM. Esto es particularmente cierto para versiones recientes de paquetes GNOME y KDE. Los paquetes RPM sólo se generan de manera intermitente.

Descompresión y extracción de software en un solo paso

Aunque puede descomprimir y extraer software en operaciones separadas, encontrará que el método más común consiste en realizar ambas acciones con un solo comando. La utilidad **tar** proporciona opciones de descompresión que puede utilizar para que **tar** primero descomprima un archivo, al invocar la utilidad de descompresión especificada. La opción **z** invoca

automáticamente **gunzip**, para desempacar un archivo **.gz** y la opción **j** desempaca un archivo **.bz2**. Utilice la opción **z** para archivos **.Z**. Por ejemplo, para combinar la operación de descompresión y desempacado de un archivo **.tar.gz** en un comando **tar**, escriba **z** en la lista de opciones, **xzvf** (consulte la sección “Extracción de software”, adelante, para conocer más acerca de estas opciones). En el siguiente ejemplo se muestra cómo combinar descompresión y extracción en un solo paso.

```
# tar xvzf htdig-3.1.6.tar.gz
```

En el caso de un archivo comprimido, use la opción **j** en vez de **z**.

```
# tar xvjf htdig-3.1.6.tar.gz
```

Descompresión de software de manera separada

Es probable que muchos paquetes de software en desarrollo o designados para implementación en plataformas cruzadas, no estén en formato RPM. En cambio, tal vez estén archivados y comprimidos. Los nombres de esos archivos terminan con la extensión **.tar.gz**, **.tar.bz2** o **.tar.Z**. Las diferentes extensiones indican distintos métodos de descompresión mediante varios comandos: **gunzip** para **.gz**, **bunzip2** para **.bz2** y **decompress** para **.Z**. En realidad, casi todo el software con formato RPM también tiene formato **.tar.gz** correspondiente. Después de descargar dicho paquete, debe descomprimirlo y desempacarlo con el comando **tar**. Los archiveros comprimidos pueden almacenar código fuente que necesita compilar o, en el caso de paquetes Java, binarios listos para ejecutarse.

Un *archivero comprimido* es un archivo creado con **tar** y después comprimido con una herramienta de compresión **gzip**. Para instalar dicho archivo, necesita descomprimirlo primero con una utilería de descompresión, como **gunzip** y después utilizar **tar** para extraer archivos y directorios integrando el paquete. En vez de la utilería **gunzip**, también puede utilizar **gzip -d**. En el siguiente ejemplo se descomprime el archivo **htdig-3.2.6.tar.gz**, reemplazándolo con una versión descomprimida llamada **htdig-3.2.6.tar**.

```
# ls
  htdig-3.2.6.tar.gz
# gunzip htdig-3.2.6.tar.gz
# ls
htdig-3.2.6.tar
```

Puede descargar archiveros comprimidos de muchos sitios diferentes, incluidos los mencionados previamente. Las descargas pueden realizarse con clientes FTP como NcFTP y gFTP o cualquier explorador Web. Una vez descargado, cualquier archivo terminando con **.Z**, **.bz2**, **.zip** o **.gz** es un archivo comprimido que debe descomprimirse.

En el caso de archivos terminando con **.bz2**, use el comando **bunzip2**. En el siguiente ejemplo se descomprime una versión **bz2**:

```
# ls
  htdig-3.2.6.tar.bz2
# bunzip2 htdig-3.2.6.tar.bz2
# ls
htdig-3.2.6.tar
```

Los archivos cuya terminación es **.bin**, son archiveros que se extraen solos. Ejecute el archivo bin como si fuera comando. Tal vez tenga que utilizar **chmod** para volverlo ejecutable. El paquete de software j2sdk se distribuye como archivo bin de autoextracción.

```
# j2sdk-1.4.2-FCS-linux-i386.tar.bin
# ls
j2sdk-1.3.0-FCS-linux-i386.tar
```

Selección de un directorio para instalación

Antes de desempacar el archivero, muévelo al directorio donde lo quiere tener. Los paquetes de código fuente suelen colocarse en un directorio como **/usr/local/src**, y los paquetes binarios van en directorios designados. Cuando los archivos de fuente abierta se desempacan, generan sus propios subdirectorios desde donde compilar e instalar el software. Una vez el paquete se instala, puede eliminar dicho directorio, dejando el archivo de paquete de código fuente original (**.tar.gz**).

Los paquetes que almacenan programas binarios listos para ejecutarse, como Java, deben extraerse en ciertos directorios. Generalmente, se trata del directorio **/usr/local**. Casi todos los archiveros, cuando están desempacados, crean subdirectorios con el nombre de la aplicación y su versión, colocando en ese subdirectorio todos los archivos o directorios integrando el paquete de software. Por ejemplo, el archivo **htdig-3.2.6.tar** desempaca un subdirectorio llamado **htdig-3.2.6**. En ciertos casos, el paquete de software conteniendo binarios precompilados está diseñado para desempacarse directamente en el subdirectorio del sistema donde se utilizará. Por ejemplo, es recomendable que **j2sdk-1.4.2-FCS-linux-i386.tar** se desempaque en el directorio **/usr/local**, donde creará un subdirectorio llamado **j2sdk-1.4.2**. El directorio **/usr/local/j2sdk-1.4.2/bin** almacena programas binarios Java.

Extracción de software

Primero, utilice **tar** con la opción **t** para revisar el contenido del archivero. Si la primera entrada es un directorio, entonces, cuando extraiga el archivero, se creará ese directorio y colocarán en él los archivos extraídos. Si la primera entrada no es directorio, primero debe crear uno y después copiar el archivo del archivero en éste. Después extraiga el archivero dentro de ese directorio. Si no existe un directorio como primera entrada, los archivos se extraen en el directorio actual. Primero debe crear un directorio para almacenar estos archivos.

```
# tar tvf htdig-3.1.6.tar
```

Ahora está listo para extraer los archivos del archivero tar. Utilice **tar** con la opción **x** para extraer archivos, la opción **v** para desplegar nombres de ruta de los archivos mientras se extraen y **f**, seguida por el nombre de un archivero:

```
# tar xvf htdig-3.2.6.tar
```

También puede descomprimir y extraer en un solo paso, usando la opción **-z** para archivos **gz** y **-j** archivos **bz2**.

```
# tar xvzf htdig-3.1.6.tar.gz
```

El proceso de extracción crea un subdirectorio incluyendo nombre y versión del software. En el ejemplo anterior, la extracción creó un subdirectorio llamado **htdig-3.2.6**.

Puede cambiar a este subdirectorio y examinar sus archivos, como **readme** e **INSTALL**.

```
# cd htdig-3.2.6
```

La instalación de su software puede diferir para cada paquete. Las instrucciones suelen proporcionarse con un programa de instalación. Asegúrese de consultar los archivos **readme** e **INSTALL**, si se incluyen. Consulte la siguiente sección acerca de la compilación de software para conocer información sobre la manera de crear e instalar aplicaciones en su sistema.

Compilación de software

Algún software pueden estar en forma de código fuente que necesita compilar antes de instalarlo. Esto es particularmente cierto en programas diseñados para implementaciones de plataforma cruzada. Los programas diseñados para ejecutarse en varios sistemas Unix, como Sun, además de Linux, pueden distribuirse como código fuente para su descarga y compilación en dichos sistemas. La compilación de dicho software se ha simplificado en gran medida en años recientes, mediante secuencias de comandos de configuración que detectan automáticamente la configuración de hardware y software del sistema; en consecuencia, después permiten compilar el programa. Por ejemplo, el nombre del compilador C en un sistema podría ser **gcc** o **cc**. Las secuencias de comandos de configuración detectan cual está presente y lo eligen para usarlo con el programa de compilación.

Una secuencia de comandos de configuración trabaja generando un Makefile personalizado, diseñado para ese sistema específico. Makefile contiene comandos detallados para compilar un programa, incluido cualquier proceso previo, vincular las bibliotecas requeridas y compilar los componentes de programas en su propio orden. Es probable que muchos Makefiles de aplicaciones complejas deban acceder varios subdirectorios de software, cada uno con componentes separados para compilar. El uso de secuencias de comandos de configuración y Makefile automatizan ampliamente el proceso de compilación, reduciendo el procedimiento a unos cuantos pasos.

En primer lugar, cambie al directorio donde se ha extraído el código fuente del software:

```
# cd /usr/local/src/htdig-3.2.6
```

Antes de compilar su software, lea los archivos **readme** o **INSTALL** incluidos. Estos archivos dan instrucciones detalladas sobre cómo compilar e instalar estos programas.

Casi todo el software puede compilarse e instalarse en tres simples pasos. El primer paso es el comando **./configure**, para generar un Makefile personalizado. El segundo paso es el comando **make**, que utiliza un Makefile en su directorio de trabajo (en este caso, el Makefile generado con el comando **./configure**) para compilar su software. El paso final también utiliza el comando **make**, pero esta vez con la opción **install**. El Makefile generado por el comando **./configure** también contiene instrucciones para instalar el software en su sistema. Al utilizar la opción **install**, se ejecutan los comandos de instalación. Para realizar la instalación, necesita iniciar sesión como usuario root, que le da la capacidad para agregar archivos de software en los directorios del sistema, según necesite. Si el software utiliza secuencias de comandos de configuración, la compilación e instalación suele incluir sólo los tres comandos siguientes:

```
# ./configure
# make
# make install
```

En el ejemplo anterior, el comando `./configure` detecta la configuración. El comando `make` compila, utilizando la secuencia de comandos Makefile generada por la operación `./configure`. El comando `make install` instala el programa en su sistema, colocando el ejecutable en un directorio, como `/usr/local/bin` y cualquier archivo de configuración en `/etc`. Cualquier biblioteca compartida creada puede ir en `/usr/local/lib`.

Una vez compilada e instalada su aplicación, y que ha revisado trabaja correctamente, puede quitar el directorio de código fuente creado cuando extrajo el software. Mantenga el archivo tar en caso de que necesite extraer el software nuevamente. Utilice `rm` con las opciones `-rf` para que todos los subdirectorios se eliminen y no deba confirmar cada eliminación.

***SUGERENCIA** Asegúrese de escribir el punto y la diagonal antes del comando `configure`. El `./` hace referencia al comando en el directorio de trabajo actual, en vez de otro comando de Linux con el mismo nombre.*

Configure las opciones del comando

Es probable que cierto software tenga opciones específicas para configurar la operación `./configure`. Para encontrar cuáles son, utilice el comando `./configure` con la opción `--help`:

```
# ./configure --help
```

Una opción común y útil es `-prefix`, para especificar el directorio de instalación:

```
# ./configure -prefix=/usr/bin
```

***SUGERENCIA** Algunas aplicaciones X antiguas utilizan `xmkmf` directamente, en vez de una secuencia de comandos de configuración para generar el Makefile necesario. Aunque oficialmente se ha reemplazado a `xmkmf`, en este caso, inserte el comando `xmkmf` en lugar de `./configure`. Asegúrese de consultar los archivos `readme` e `INSTALL` para el software.*

Bibliotecas de desarrollo

Si está compilando un programa X basado en GNOME o KDE, asegúrese de haber instalado las bibliotecas de desarrollo. En el caso de aplicaciones X, cerciórese de que el programa `xmkmf` también está instalado. Si elige una instalación estándar cuando instala su sistema de distribución, lo más probable es que no se instalen. En el caso de distribuciones usando paquetes RPM, estas bibliotecas vienen en forma de conjunto de paquetes RPM de desarrollo, generalmente con la palabra “development” o “develop” en su nombre. Necesita instalarlos con `rpm`. GNOME, en particular, tiene un conjunto extenso de paquetes RPM de bibliotecas de desarrollo. Muchas aplicaciones X demandan bibliotecas compartidas especiales. Por ejemplo, es probable que algunas aplicaciones necesiten la biblioteca `xforms` o `qt`. Tal vez tenga que obtener algunas de sitios en línea.

Bibliotecas compartidas y estáticas

Las bibliotecas son estáticas, compartidas o dinámicas. Una biblioteca *estática* es una cuyo código se incorpora en el programa cuando se compila. Sin embargo, una biblioteca *compartida*, carga el código para acceder a él cuando el programa está en ejecución. Al compilarse, el programa sólo detecta la biblioteca necesaria. Después, cuando el programa se ejecuta, esa biblioteca se carga

y el programa puede acceder a sus funciones. Una biblioteca *dinámica* es una variación de una biblioteca compartida. Al igual que ésta, puede cargarse cuando el programa se ejecuta. Sin embargo, en realidad sólo se carga hasta que el programa le indica hacerlo. También se deja de cargar mientras el programa se ejecuta, y otra biblioteca puede cargarse en su lugar. Las bibliotecas compartidas y dinámicas dan lugar a un código mucho más pequeño. En vez de un programa incluyendo la biblioteca como parte de su archivo ejecutable, sólo necesita una referencia a ésta.

Las bibliotecas disponibles en su sistema residen en los directorios `/usr/lib` y `/lib`. Los nombres de estas bibliotecas siempre comienzan con el prefijo **lib**, seguido del nombre de la biblioteca y un sufijo. El sufijo es diferente para bibliotecas compartidas o estáticas. Una biblioteca compartida tiene la extensión **.so**, seguida por números de versiones menor o mayor. Una biblioteca estática sólo tiene la extensión **.a**. Se hace una distinción adicional para bibliotecas compartidas en el antiguo formato **a.out**. Tienen la extensión **.sa**. La sintaxis para el nombre de la biblioteca es el siguiente:

```
libnombreso.mayor.minor  
libnombre.a
```

libnombre puede ser cualquier cadena y sólo identifica una biblioteca. Puede tratarse de una palabra, unos cuantos caracteres o incluso una sola letra. El nombre de la biblioteca math compartida es **libm.so.5**, donde math sólo se identifica por la letra **m** y la versión mayor es **5**, mientras **libm.a** es una biblioteca math estática. El nombre de la biblioteca X Window es **libX11.so.6**, donde la biblioteca X Window sólo se identifica con las letras **X11** y su versión mayor es **6**.

Casi todas las bibliotecas compartidas se encuentran en los directorios `/usr/lib` y `/lib`. Siempre se busca primero en estos directorios. Algunas bibliotecas compartidas se ubican en un directorio especial propio. Una lista se encuentra en el archivo de configuración `/etc/ld.conf`. En estos directorios siempre se buscará una biblioteca dada. Como opción predeterminada, Linux busca primero las bibliotecas compartidas, luego las estáticas. Siempre que se actualiza una biblioteca compartida o se instala una nueva, necesita ejecutar el comando **ldconfig** para actualizar sus entradas en el archivo `/etc/ld.conf`, además de los vínculos con éste (si instala desde un paquete RPM, esto suele hacerse de manera automática).

Archivo Makefile

Si no existe secuencia de comandos de configuración ni el programa usa **xmkmf**, tal vez deba editar directamente el Makefile del software. Asegúrese de revisar la documentación de dicho software, para ver si deben hacerse cambios a Makefile. Es probable que sólo se necesiten unos cuantos cambios, pero cambios más detallados requieren conocimientos de programación C y del funcionamiento de **make**. Si configura correctamente el archivo Makefile, tal vez sólo deba insertar los comandos **make** y **make install**. Un posible problema consiste en localizar las bibliotecas de desarrollo para C y X Window System. Las bibliotecas X están en el directorio `/usr/X11R6/lib`; las C estándar, en el directorio `/usr/lib`.

Directorios de comandos y programas: PATH

Los programas y comandos suelen instalarse en varios directorios de sistema estándares, como `/bin`, `/usr/bin`, `/usr/x11R6/bin` o `/usr/local/bin`. Sin embargo, algunos paquetes colocan comandos en subdirectorios, creados dentro de uno de estos directorios estándar o un directorio completamente separado. En tales casos, es probable no pueda ejecutar tales comandos porque su sistema no logra

localizarlos en el nuevo subdirectorio. Su sistema mantiene un conjunto de directorios en que busca comandos cada vez que ejecuta uno. Este conjunto de directorios se almacena en una variable de sistema llamada **PATH**, creada cuando inicia su sistema. Si un comando se encuentra en un directorio que no se localiza en esta lista, su sistema no podrá localizarlo y ejecutarlo. Para utilizar dichos comandos, primero necesita agregar el nuevo directorio al conjunto de directorios en la variable **PATH**. Las herramientas de instalación como RPM actualizarán automáticamente **PATH**, con los directorios apropiados para su caso.

La variable **PATH** es agregada por servicios diferentes que inician cuando el sistema arranca. Un método más seguro consiste en agregar una definición **PATH** en el archivo **/etc/profile**, mediante la función **pathmunge** del archivo, si está disponible.

/etc/profile

Para que una aplicación esté disponible para todos los usuarios, agregue el directorio del software a la entrada de ruta, en la secuencia de comandos **/etc/profile**. Se trata de una secuencia de sistema ejecutada para cada usuario cuando inicia sesión. Edite con cuidado el archivo **/etc/profile** con un editor de texto, como KEdit, Gedit, Emacs o Vi (tal vez quiera hacer una copia de seguridad primero, con el comando **cp**). En algunas distribuciones, puede agregar de manera sencilla un directorio a la variable **PATH**, mediante la función **pathmunge**, también definida en **/etc/profile**. Por ejemplo, si quiere instalar Java 2 SDK, los comandos de Java se instalan en un subdirectorio llamado **j2sdk-1.4.2/bin**, en el directorio **/usr/local**. El nombre de ruta completo de este directorio es **/usr/local/j2sdk-1.4.2/bin**. Necesita usar **pathmunge** para agregar este directorio a la lista de directorios asignados a **PATH** en el archivo **/etc/profile**.

```
pathmunge /usr/local/ j2sdk-1.4.2/bin
```

Verá otros usos de **pathmunge** en **/etc/profile**, como agregar **/sbin** al usuario root. Después de crear sus cambios, puede ejecutar el archivo **profile** para que surtan efecto:

```
$ . /etc/profile
```

NOTA En sistemas antiguos tenía que crear una nueva entrada de asignación para la variable **PATH**, agregando una línea que comenzaba con **PATH**, seguida por un signo **=**, el término **\$PATH**, dos puntos y después el directorio que agregaría. El **\$** antes de **PATH** extraía el nombre de ruta de la variable **PATH**. Si agregaba más de un directorio, debía usar dos puntos para separarlos y otros dos puntos tenían que incluirse al final. En el siguiente ejemplo se muestra la variable **PATH**, con su lista de directorios y el directorio **/usr/local/j2sdk-1.4.2/bin** agregado. Observe el **\$** antes de **PATH** y después del signo **=**, **PATH=\$PATH. PATH=\$PATH:/usr/local/j2sdk-1.4.2/bin**:

.bash_profile

Los usuarios individuales pueden personalizar sus variables **PATH** colocando una asignación **PATH** en su archivo **bashrc** o **.bash_profile**. De esta forma, los usuarios pueden acceder a comandos y programas que crean o instalan para su propio uso en directorios de usuario. Los archivos **.bash_profile** del usuario ya contiene la siguiente definición **PATH**. Observe el uso de **\$PATH**, conservando todos los directorios agregados a **PATH** con secuencias de comandos de inicio anteriores como **/etc/profile**.

```
PATH=$PATH:$HOME/bin
```

La siguiente entrada del archivo **.bash_profile** agrega el directorio **newbin** de un usuario a la variable **PATH**. Observe los dos puntos colocados antes del nuevo directorio y el uso de la variable **\$HOME** para especificar el nombre de ruta del directorio home del usuario.

```
PATH=$PATH:$HOME/bin/:$HOME/newbin
```

Para el usuario **root**, la definición de **PATH** también incluye directorios **sbin**, almacenando programas de administración del sistema, a los que el usuario **root** necesita acceder. Aquí se muestra la **PATH** del usuario **root**:

```
PATH=/usr/local/sbin:/usr/sbin:/sbin:$PATH:$HOME/bin
```

Subversion y CVS

Subversion y el sistema de versión concurrente (CVS, Concurrent Version System), son métodos de software de desarrollo que permiten a los desarrolladores trabajar desde ubicaciones remotas, en software almacenado en un servidor central. Subversion es una versión mejorada de CVS, diseñada para reemplazarlo. Al igual que CVS, Subversion trabaja con depósitos CVS, que acceder software casi de la misma forma. Subversion agrega características: mejor acceso a directorios y archivos, además de soporte para información de metadatos.

Los sitios CVS permiten a varios desarrolladores trabajar en un archivo al mismo tiempo. Esto significa que soportan desarrollo paralelo, de modo que programadores de todo el mundo trabajen en la misma tarea simultáneamente mediante una conexión a Internet. Se ha vuelto popular entre desarrolladores de Linux como medio de creación de software, mediante el uso de Internet. Los sitios CVS también son fuente de casi todas las versiones actualizadas de diferente software. Proyectos en curso como KDE y GNOME usan servidores Subversion o CVS para publicar versiones recientes de sus aplicaciones de escritorio, sobre todo porque es fácil de usar para el desarrollo de programas en Internet. El sitio **sourceforge.net** proporciona un depósito CVS para muchos proyectos de Linux en curso. Numerosos sitios CVS ahora soportan ViewCVS (versión mejorada de webCVS), interfaz de explorador Web con un depósito CVS para explorar y elegir versiones de software más sencillas. Aprenderá más acerca de CVS en **cvshome.org** y acerca de Subversion en **subversion.tigris.org**.

El uso de un depósito CVS para desarrollo de software abarca procedimientos para acceder una versión de software, haciendo cambios de manera local en su sistema y, después, subiendo su versión cambiada de regreso al depósito CVS. En efecto, usted revisa el software, hace sus cambios de tal forma que se guarden cuidadosamente y después regresa sus versiones al depósito. CVS fue desarrollado originalmente como portal para el sistema de control de revisión (RCS, Revision Control System), más antiguo y que comparte muchos de los mismos comandos.

Empaquetamiento de su software con RPM

Muchos entornos de investigación y empresariales desarrollan software personalizado para distribuirlo en su organización. A veces los paquetes de software se descargan y después personalizan para utilizarlos en una empresa determinada. Para instalar de manera más sencilla dicho software, los administradores empaquetan programas en paquetes propios RPM. Tales paquetes pueden incluir versiones propias de archivos de configuración, documentación y fuentes, así como binarios modificados. RPM instala automáticamente el software en su sistema en los directorios designados, junto con cualquier documentación, bibliotecas o programas de soporte.

El proceso de creación de paquetes está diseñado para llevar los programas a través de varios pasos, empezando por desempacarlos de un archivero, después compilar su código fuente y, por último, generar el paquete RPM. Puede saltarse cualquiera de estos pasos, hasta llegar al último. Si su software ya está empacado, puede empezar por compilarlo. Si su software está compilado, puede iniciar por instalarlo. Si ya está instalado, puede pasar directamente a crear el paquete RPM.

Los procesos de generación de RPM solían incluirse con el comando **rpm**. Ahora se incorporan en una herramienta separada llamada **rpmb**. Esta herramienta, junto con las bibliotecas de soporte y documentación, está localizada en el paquete rpm-build. Asegúrese de que este paquete está instalado antes de que trate de generar sus paquetes RPM. De todas formas puede ejecutar el comando **rpm** con las opciones para generar, pero sólo son alias para los comandos **rpmb** correspondientes.

Aplicaciones de oficina y bases de datos

Hay varias *suites* de oficina disponibles para Linux (véase la tabla 11-1). Entre éstas se incluyen procesadores de textos profesionales, administradores de presentaciones, herramientas de dibujo y hojas de cálculo. En el presente capítulo se describen las versiones adquiribles en forma gratuita. Sun ha iniciado el desarrollo de un conjunto de oficina de fuente abierta empleando código StarOffice. Las aplicaciones, conocidas como OpenOffice.org, proporcionan aplicaciones de oficina integradas en GNOME. OpenOffice.org es actualmente la principal aplicación de oficina soportada por casi todas las distribuciones de Linux. KOffice es un conjunto de oficina libre diseñado para utilizarse con KDE. La *suite* de oficina GNOME integra aplicaciones GNOME en un conjunto de productividad de acceso gratuito. CrossOver Office, de CodeWeavers, ofrece soporte confiable para ejecutar aplicaciones de MS Office para Windows directamente en Linux, integrándolas con KDE y GNOME. También puede comprar *suites* de oficina comerciales como StarOffice, de Sun. Para edición de publicaciones de escritorio, sobre todo la generación de PDF, puede utilizar Scribus, una herramienta de plataforma cruzada disponible en el depósito de Fedora.

Se pueden adquirir varios sistemas de administración de bases de datos para Linux. Entre éstos se incluyen sistemas de administración de bases de datos de gran capacidad de nivel comercial, como Oracle, DB2 de IBM y Sybase. También hay bases de datos de fuente abierta para Linux, como MySQL y PostgreSQL. Estos son de los sistemas más utilizados en Linux. Casi todos los sistemas de administración de bases de datos disponibles para Linux, responden a un diseño para trabajo con grandes bases de datos relacionales. En el caso de pequeñas bases de datos personales, use sistemas de administración de bases de datos de escritorio desarrollados por KDE y GNOME. Además, hay software para bases de datos que pueden accederse con el lenguaje de programación de bases de datos Xbase. Existen bases de datos pequeñas empleando formatos desarrollados originalmente para dBase en PC. En la tabla 11-6, en páginas posteriores de este capítulo, se muestran varios sistemas de bases de datos disponibles para su ejecución en Linux.

Linux también ofrece varios editores de texto, que van desde procesadores de palabras simples para notas, hasta editores con características más complejas, como un revisor ortográfico, uso de búfer y coincidencia de patrones. Todos generan archivos de texto de caracteres y pueden usarse para editar cualquier archivo de texto de Linux. Los editores de texto suelen emplearse en tareas de

Sitio Web	Descripción
openoffice.org	Conjunto de oficina de fuente abierta basado en StarOffice
koffice.org	Conjunto KOffice para KDE
gnome.org/gnome-office	GNOME Office para GNOME
sun.com/staroffice	Conjunto StarOffice
codeweavers.com	CrossOver Office (soporte para MS Office)
scribus.net	Herramienta de escritorio para edición de publicaciones Scribus

TABLA 11-1 Conjuntos de oficina de Linux

administración de sistema para cambiar o agregar entradas en archivos de configuración de Linux, ubicados ya sea en el directorio */etc*, archivos *dot* de inicialización o aplicación del usuario, alojados en el directorio home del usuario. Puede usar cualquier editor de texto para trabajar en archivos de código fuente de cualquier lenguaje de programación o secuencias de comandos de programa en shell.

Ejecución de Microsoft Office en Linux: CrossOver

Una de las preocupaciones principales para los usuarios nuevos de Linux es qué tipo de acceso tendrán a sus archivos de Microsoft Office, en especial los archivos de Word. El sistema operativo Linux y muchas aplicaciones están diseñados para ofrecer acceso íntegro a archivos de MS Office. Las principales *suites* de oficina de Linux, incluidas KOffice, OpenOffice.org y StarOffice, leen y manejan cualquier archivo de Microsoft Office. Además, estas suites de oficina alcanzan de manera rápida el mismo nivel de características y soporte a tareas de oficina encontradas en Microsoft Office.

Si quiere usar cualquier aplicación de Windows en Linux, tres opciones importantes son el soporte de API a ventanas virtuales Wine, la tecnología de plataforma virtual VMware y CrossOver Office de Code Weavers. VMware y CrossOver son paquetes comerciales.

Wine posibilita ejecutar directamente muchas aplicaciones de Windows, usando soporte a la API de ventanas virtuales. Visite la página de Wine, winehq.com, para conocer una lista de aplicaciones a que se da soporte. Las aplicaciones bien escritas pueden ejecutarse directamente desde Wine, como el lector de noticias newsbin. A menudo deberá tener un sistema Windows trabajando para copiar las DLL de sistema necesarias para aplicaciones particulares. También puede importar sus fuentes de Windows copiándolas directamente en el directorio de fuentes de Wine. Cada usuario instala sus propias versiones de Wine con partición C: propia, simulada donde se instalarán las aplicaciones de Windows. La unidad simulada se instala como **drive_c** en su directorio **.wine**. El directorio **.wine** es un directorio oculto. Generalmente, no se despliega con el comando **ls** ni el administrador de archivos de GNOME.

En una ventana de terminal, el uso del comando **wine** con un programa de instalación instalará automáticamente la aplicación de Windows en la unidad C: simulada. En el siguiente ejemplo se instala la aplicación **newsbin.exe**.

```
$ wine newsbin.exe
```

Una vez instalada, junto con las DLL necesarias, aparecerá un icono en su escritorio de Linux para la aplicación. La aplicación iniciará normalmente. Incluso puede usar cualquiera de sus directorios Linux para archivos de datos de su aplicación de Windows, en vez de la unidad C: simulada.

CrossOver Office es un producto comercial que permite instalar y ejecutar la mayor parte de aplicaciones de Microsoft Office. CrossOver Office fue desarrollado por CodeWeavers, que también da soporte para plug-ins de exploradores Web de Windows, además de varias aplicaciones para

Windows como Adobe Photoshop. CrossOver ofrece versiones profesionales y estándar, facilitando soporte confiable a las aplicaciones. Encontrará más acerca de CrossOver Office en codeweavers.com.

CrossOver puede instalarse en modo multiusuario privado o administrado. En el modo multiusuario privado, cada usuario instala su propio software de Windows, como las versiones completas de Office. En el modo multiusuario administrado, el software de Windows se instala una vez y todos los usuarios lo comparten. Cuando instala nuevo software, primero abre la herramienta de inicio de CrossOver y después en el panel Agregar o quitar verá una lista de software soportado. Esto incluirá aplicaciones de Office, además de Adobe, incluidas versiones antiguas de Photoshop. Un panel Instalar programas dará la opción para elegir si quiere instalar desde un CD-ROM o archivo **.exe**. Para el caso de Office en un CD-ROM, seleccione CD-ROM, coloque el CD-ROM de Windows en la unidad correspondiente y después haga clic en Siguiente. El instalador de Windows Office iniciará en una ventana de Linux y procederá como si fuera un sistema Windows. Cuando la instalación requiera reiniciar el sistema, CrossOver simulará el reinicio. Una vez el software esté instalado, verá un menú Aplicaciones de Windows en el menú principal, desde donde podrá iniciar su software de Windows. Las aplicaciones se ejecutarán dentro de una ventana de Linux, como si se ejecutaran en Windows. También puede probar CrossOver con aplicaciones sin soporte. Hay probabilidad de que se ejecuten.

Con VMware, es posible ejecutar Windows en Linux, lo que permite ejecutar aplicaciones de Windows, incluido Microsoft Office, en su sistema Linux. Para conocer más información, consulte el sitio Web de VMware en vmware.com.

NOTA Aunque Linux permite a los usuarios montar y acceder directamente a cualquiera de las particiones antiguas de DOS o FAT32, usadas en Windows 95, 98 y ME, puede montar particiones NTFS (Windows Vista, XP, 2000 y NT) como sólo lectura, con soporte para escritura. Existen dos controladores para montar NTFS, **ntfs-3g** y el soporte de proyecto NTFS original. Los controladores **ntfs-3g** soportan particiones NTFS.

OpenOffice.org

OpenOffice.org (OO) es una *suite* de aplicaciones de oficina totalmente integrada, desarrollada como proyecto de fuente abierta de distribución gratuita para todos. Se incluye como la principal *suite* de oficina para la mayor parte de distribuciones Linux, puede accederse desde un menú Oficina. Contiene aplicaciones de procesamiento de palabras, hoja de cálculo, presentaciones y dibujo (véase la tabla 11-2). Existen versiones de OpenOffice.org para Linux, Windows y Mac OS. Puede obtener información, como manuales en línea y FAQ, además de versiones actualizadas, en el sitio Web de OpenOffice.org, en openoffice.org.

Aplicación	Descripción
Calc	Hoja de cálculo de OpenOffice.org
Draw	Aplicación de dibujo de OpenOffice.org
Writer	Procesador de palabras de OpenOffice.org
Math	Compositor de formulas matemáticas de OpenOffice.org
Impress	Administrador de presentaciones de OpenOffice.org
Base	Interfaz de base de datos para acceder y administrar varias bases de datos distintas

TABLA 11-2 Aplicaciones de OpenOffice.org

NOTA El desarrollo de OpenOffice.org se da como proyecto de fuente abierta llamado openoffice.org. El código base proviene originalmente de StarOffice. El desarrollo del código en el proyecto openoffice.org se incorporará más adelante en versiones futuras de StarOffice.

OpenOffice.org es un conjunto de aplicaciones integrado. Puede abrir el escritor, la hoja de cálculo o la aplicación de presentaciones directamente. Además, en casi todas las aplicaciones OpenOffice, puede seleccionar Nuevo, del menú Archivo, y seleccionar una aplicación diferente, si lo desea. El procesador de palabras Write soporta características de procesador de palabras estándar, como cortar y pegar, revisión ortográfica y formación de texto, además de estilos de párrafo (véase la figura 11-1). Es posible incrustar objetos en los documentos (por ejemplo, usar Draw para crear figuras que puede arrastrar y colocar en el documento de Writer). Encontrará más información acerca de cada componente en las páginas de producto respectivas, en openoffice.org/product.

Calc es una hoja de cálculo de nivel profesional. Con Math, creará formulas para incrustarlas en un documento de texto. El administrador de presentación (Impress), es capaz de crear imágenes para presentaciones, como círculos, rectángulos y elementos de conexión como flechas, además de ilustraciones basadas en vectores. Impress soporta características avanzadas: transformación morfológica de objetos, agrupación de objetos y definición de degradados. Draw es una herramienta de dibujo sofisticada que incluye herramientas de modelado 3-D. Sirve para crear imágenes simples o complejas, incluido texto animado alineado en curvas. OpenOffice.org incluye una herramienta de configuración de impresoras capaz de seleccionar impresoras, fuentes, tamaños de papel y formatos de páginas.

NOTA StarOffice es un conjunto de aplicaciones de oficina totalmente integrado y compatible con Microsoft Office, desarrollado y soportado por Sun Microsystems, sun.com/staroffice. Sun proporciona StarOffice como producto comercial, aunque el uso educativo es gratuito.

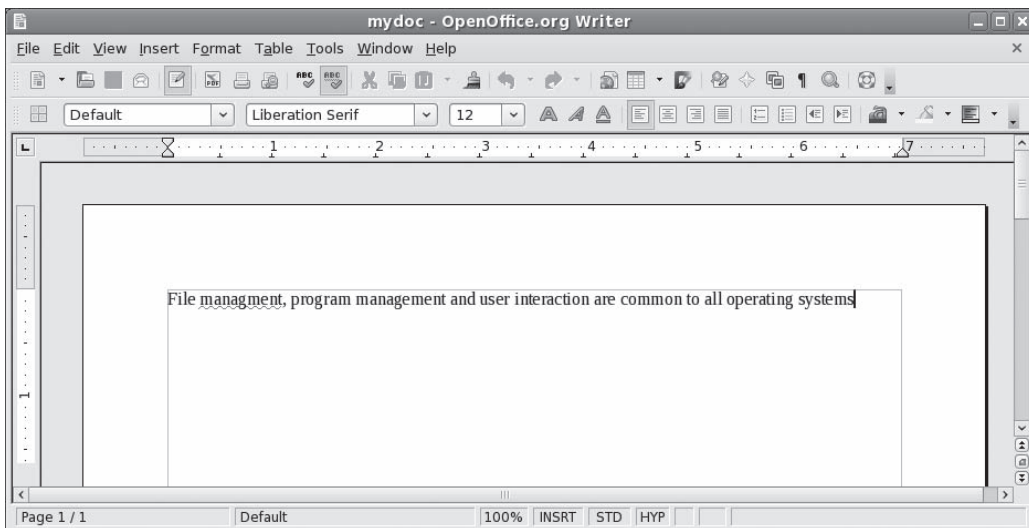


FIGURA 11-1 Procesador de palabras OpenOffice.org's

OpenOffice.org también brinda acceso a varios archivos de base de datos. Entre los tipos de archivo soportados se incluyen los de base de datos ODBC 3 (Open Database Connectivity), JDBC (Java), ADO, MySQL, dBase, Concurrent Versions System (CVS), PostgreSQL y MDB (Microsoft Access). Consulte las páginas Base y Project de OpenOffice.org (dba.openoffice.org) para conocer información detallada sobre controladores y bases de datos soportadas.

OpenOffice.org presenta un modelo base de componentes, susceptible de ser programado para desarrollar aplicaciones personalizadas. Revise el proyecto API de OpenOffice.org para más detalles (api.openoffice.org). Las herramientas para desarrollo de software (SDK, Software Development Kit) de OpenOffice.org, ofrecen soporte para usar componentes de OpenOffice.org en aplicaciones escritas en C++ o Java. El modelo de objetos unificados de red (UNO, Unified Network Objects) es un modelo de componentes para OpenOffice.org, facilita la interacción entre lenguajes de programación, otros modelos de objeto y conexiones de red.

KOffice

KOffice es un conjunto de oficina integrado para KDE (K Desktop Environment, entorno de escritorio K) consistente de diversas aplicaciones para oficina, incluidos procesador de palabras, hoja de cálculo y aplicaciones gráficas. Puede descargarlo al utilizar Pirut/Yum (Agregar o quitar software). Todas las aplicaciones están escritas para el modelo de componentes KOM, habilitando que los componentes de cualquier aplicación se utilicen en otra. Esto significa que puede incrustar una hoja de cálculo de KSpread o diagramas de Karbon14 en un documento KWord. Obtendrá más información acerca de KOffice en el sitio Web de KOffice, koffice.org.

SUGERENCIA Las aplicaciones de KOffice tienen filtros para importar o exportar archivos desde aplicaciones populares como AbiWord, aplicaciones de OpenOffice.org, MS Word e incluso documentos de Palm. La confiabilidad de estos filtros varía y debe revisar la página Web de filtros KOffice para conocer listas de varios filtros y su estabilidad.

Aplicaciones de KOffice

Actualmente, KOffice incluye KSpread, KPresenter, KWord, Karbon14, KFormula, KChart, Kugar, Krita y Kivio (véase la tabla 11-3). La aplicación para contactos, Kontact, surgió de un proyecto aparte. Kontact es una aplicación de agenda integrada compuesta por KMail, KOrganizer, KAddressbook y KNotes. Kspread es una hoja de cálculo, KPresenter sirve para presentaciones, Karbon14 trabaja con gráficos vectoriales, KWord es un procesador de palabras parecido a Publisher, Kformula funciona como editor de fórmulas y KChart genera gráficas y diagramas. Kugar genera de informes, Krita edita imágenes de mapas de bits y Kivio crea gráficas de flujo. Kexi integra bases de datos con aplicaciones KOffice, mismo que soporta actualmente PostgreSQL y MySQL.

KSpread, la aplicación de hoja de cálculo, incorpora operaciones básicas encontradas en la mayor parte de las hojas de cálculo, con fórmulas similares a las utilizadas en Excel. También puede incrustar gráficas, imágenes o fórmulas usando KChart, Krita, Karbon14 o KFormula.

Con KChart, se crean diferentes tipos de gráficas, como gráficas de barra, circulares y de líneas, además de diagramas. Para generar una gráfica, puede manipular datos en KSpread o insertar propios. Con KPresenter, se crean presentaciones constando de texto e imágenes modeladas mediante diferentes fuentes, orientaciones y atributos como colores. Puede agregar elementos como burbujas de diálogo, flechas e imágenes prediseñadas, además de incrustar cualquier componente de KOffice. Karbon14 es un programa gráfico basados en vectores, muy similar a Adobe Illustrator

Aplicación	Descripción
KSpread	Hoja de cálculos
KPresenter	Programa de presentación
KOShell	Área de trabajo de Koffice para aplicaciones de KOffice.
Karbon14	Programa para gráficos vectoriales
KWord	Procesador de palabras (edición de publicaciones de escritorio)
KFormula	Editor de fórmulas matemáticas
KChart	Herramienta para dibujar gráficas y diagramas
Kugar	Generador de informes
Krita	Programa para manipulación de dibujos e imágenes
Kivio	Generador y editor de gráficas de flujo (similar a Vivio)
Kexi	Integración de base de datos
KPlato	Administración y planeación de proyectos
Kontact (proyecto separado)	Aplicación de agenda que incluye correo, libreta de direcciones y organizador.

TABLA 11-3 Aplicaciones de KOffice

y Draw de OpenOffice.org. Soporta operaciones de gráficos estándar: giro, cambio de tamaño y alineación de objetos

KWord se describe mejor como un editor de publicaciones de escritorio, con muchas características encontradas en aplicaciones similares a Microsoft Publisher y FrameMaker. Aunque también es un procesador de palabras funcional, KWord no se basa en una estructura de página como Word o WordPerfect. En cambio, el texto se configura en marcos (*frames*) dentro de la página como objetos. Los marcos, como los objetos de un programa de dibujo, se mueven, cambian de tamaño e incluso tienen la capacidad para ser reorientados. Puede organizar los marcos en un conjunto, haciendo que el texto fluya de un marco al otro.

KParts

Los componentes incrustados facilitan actualizaciones en tiempo real. Por ejemplo, si usa KChart para generar una gráfica en un documento de KWord, a través de datos en una hoja de cálculo KSpread y después cambiando los datos seleccionados en la hoja de cálculo, KChart actualiza la gráfica automáticamente en el documento de KWord. En efecto, está creando un documento compuesto (integrado por varias aplicaciones). Esta capacidad se implementa con el modelo de componentes de KDE conocido como KParts. KParts proporciona comunicación entre objetos distribuidos. A este respecto, considere que una aplicación funciona también como servidor, proporcionando servicios a otras aplicaciones especializadas. Un procesador de palabras, preparado para servicios relativos a la formación de párrafos o revisión ortográfica, ofrece estos servicios a todas las aplicaciones de KOffice. En ese sentido, no es necesario que cada aplicación posea funciones propias de formación.

KParts se implementa con el protocolo de comunicaciones de escritorio (DCOP, *Desktop Communications Protocol*). Se trata de un mecanismo IPC/RPC simple, pequeño y rápido para

comunicación entre procesos (IPC, InterProcess Communication), basado en el protocolo ICE (Inter-Client Exchange, intercambio entre clientes) de X Window System.

Ahora las aplicaciones de KDE usan bibliotecas DCOP para administrar sus comunicaciones con los demás. DCOP hace que el desarrollo de aplicaciones KOffice sea más sencillo y estable.

GNOME Office

El proyecto GNOME Office incluye tres aplicaciones de oficina: AbiWord, Gnumeric y GNOME-DB. Integrantes previos de GNOME Office todavía brindan ciertas tareas de Office, como el correo electrónico y cliente de contactos Evolution, de Novell. Muchos miembros anteriores todavía son proyectos de GNOME y en gnome.org/projects se encuentra información y listas de éstos. Conocerá más al respecto en el sitio de GNOME Office, gnome.org/gnome-office. En la tabla 11-4 se muestra una lista de aplicaciones actuales de GNOME Office, entre las que se incluyen aquellas ajenas al conjunto de oficina GNOME. Todas implementan el modelo CORBA para componentes incrustados, asegurando la capacidad para arrastrar y colocar objetos en la interfaz GNOME.

Gnumeric, una de las aplicaciones de GNOME Office, es una hoja de cálculo de GNOME, programa profesional diseñado para remplazar las hojas de cálculo comunes. Como GNOME, Gnumeric está disponible de manera gratuita bajo la licencia pública de GNU. Gnumeric se incluye con GNOME y lo encontrará instalado con éste en cualquier distribución que soporte GNOME. Descargue las versiones actuales de gnome.org/projects/gnumeric. Gnumeric permite características de hoja de datos GUI estándar, incluidos relleno automático y formación de celdas, además de facilitar amplia cantidad de formatos. Soporta operaciones de arrastre y colocación, permitiéndole seleccionar y después mover o copiar celdas a otro lugar. Gnumeric también da soporte a plug-ins, con lo que permite extender y personalizar fácilmente sus capacidades.

AbiWord, otra aplicación de GNOME Office, es un procesador de palabras de fuente abierta que pretende ser una solución completa de plataforma cruzada; se ejecuta en Mac, Unix y Windows, además de Linux. Es parte de un conjunto de aplicaciones de productividad desarrolladas por el proyecto AbiSource (abisource.com).

Aplicación	Descripción
GNOME Office	
AbiWord	Procesador de palabras de plataforma cruzada
Gnumeric	Hoja de cálculo
GNOME-DB	Conectividad de bases de datos
Otras aplicaciones de oficina para GNOME	
Evolution	Correo electrónico, calendario y organizador personal integrado (Novell)
Dia	Editor de gráficas y diagramas de flujo (proyecto GNOME)
GnuCash	Administrador de finanzas personales (proyecto GNOME)
Balsa	Cliente de correo electrónico (proyecto GNOME)
Planner	Administrador de proyectos (proyecto GNOME)
OpenOffice.org	Conjunto de oficina OpenOffice.org

TABLA 11-4 GNOME Office y otras aplicaciones de oficina para GNOME

El proyecto GNOME-DB ofrece una biblioteca GNOME Data Access (GDA, acceso a datos de GNOME) sustenta varios tipos de bases de datos como PostgreSQL, MySQL, Microsoft Access y unixODBC. Proporciona una API a la que se pueden conectar bases de datos. Estas conexiones de servidor se basan en CORBA. A través de esta API, las aplicaciones GNOME acceden a la base de datos. Encontrará más acerca de GNOME-DB en gnome-db.org.

Dia es un programa de dibujo diseñado para crear diagramas (proyecto de GNOME). Puede seleccionar diferentes tipos de diagramas, como una base de datos, circuitos, gráficas de flujo y diagramas de red. Los elementos se crean de manera sencilla, junto con líneas y arcos con diferentes tipos de terminaciones como flechas o diamantes. Los datos se guardan en formato XML, por lo que pueden transportarse de manera sencilla a otras aplicaciones.

GnuCash (gnucash.org) es una aplicación de finanzas personales para administrar cuentas, acciones y gastos (proyecto GNOME). Incluye soporte a actividades bancarias en casa con la interfaz OpenHBCI. Esta es la interfaz computacional de fuente abierta para actividades bancarias en casa (openhbc.sourceforge.net).

Visores de documentos (PostScript, PDF y DVI)

Los visores PostScript, PDF y DVI son los más utilizados en aplicaciones de oficina (véase la tabla 11-5). Evince y Ghostview despliegan archivos PostScript (.ps) y PDF (.pdf). La interfaz X Window System de Ghostsview es **gv**. KPDF y Xpdf son visores para PDF. KPDF incluye muchas características estándar de Adobe Reader, como acercamiento, despliegue a dos páginas y modo de pantalla completa. Como opción, puede descargar desde Adobe Acrobat Reader para Linux, para desplegar archivos PDF. Todos estos visores también cuentan con la capacidad para imprimir documentos. Si quiere generar documentos PDF, use el editor de publicaciones de escritorio Scribus (scribus.net) y para editar documentos PDF **pdfedit**.

Linux también presenta una herramienta de configuración de composición tipográfica a nivel profesional, llamada TeX, que suele utilizarse para componer fórmulas matemáticas complejas. TeX genera un documento DVI que después se despliega con visores de DVI, de los cuales existen varios para Linux. Los archivos DVI generados por la aplicación de documento TeX se ven con KDVI, un plug-in para la herramienta KViewShell. Ésta última puede desplegar e imprimir cualquier tipo de documento para el que cuente con el plug-in correspondiente.

Visor	Descripción
Evince	Visor de documentos para archivos PostScript y PDF
KPDF	Herramienta KDE para desplegar archivos PDF
KGhostView	Interfaz KDE para desplegar archivos PostScript y PDF
xpdf	Herramienta de X Window System sólo para desplegar archivos PDF
KDVI	Herramienta de KDE para desplegar archivos DVI de TeX (plug-in para KViewShell)
Acrobat Reader	Aplicación de despliegue de PDF y PostScript de Adobe
Gnome-gv	Gnome Ghostscript viewer

TABLA 11-5 Visores para PostScript, PDF y DVI

Acceso a PDA

Para muchas unidades PDA (*Personal Digital Assistant*), puede emplear las herramientas pilot para acceder a su handheld, transfiriendo información entre su unidad y el sistema. El paquete **pilot-link** contiene herramientas que utiliza para acceder a su PDA. Visite **pilot-link.org** para acceder a documentación detallada y vínculos útiles. Los nombres de las herramientas suelen comenzar con “pilot”; por ejemplo, **pilot-addresses** lee campos de una libreta de direcciones. Otras herramientas cuyos nombres comienzan con “read”, permiten convertir datos de la Palm/PDA para ser usadas en otras aplicaciones; **read-expenses**, por ejemplo, envía la salida de datos de gastos como texto estándar. Una de las herramientas más útiles es **pilot-xfer**, usada para respaldar su Palm.

En vez de utilizar comandos directamente desde una ventana de terminal, puede recurrir a las aplicaciones J-Pilot, KPilot y GNOMEpilot para acceder a su Palm/PDA. Para utilizar su PDA en GNOME, se valdría de la applet gnome-pilot desde el panel GNOME para configurar sus conexiones. En la ventana Preferencias de la applet gnome-pilot (haga clic con el botón derecho en la applet), el panel Conduits le permite habilitar varias operaciones de sincronización de host para que se realicen automáticamente, incluidas de correo electrónico, memos y archivos de instalación. Haga clic en el botón Ayuda para leer un manual detallado.

J-Pilot proporciona una GUI que permite realizar tareas básicas como sincronizar libretas de direcciones y escribir memos. KPilot se incluye con el paquete **kpim**, instalado como un componente del escritorio KDE. Cuando inicia **kpilot**, primero se sincronizará automáticamente con su PDA. Después tiene la opción de usar Evolution o KContact con su PDA, o sólo realizar copias de seguridad. Enseguida puede realizar operaciones como sincronizar el host, ver direcciones e instalar archivos. Para conversiones de formato de texto y de Palm, utilice KPalmDoc. Esta herramienta convertirá sus archivos de texto en archivos Palm y viceversa.

SUGERENCIA El nombre de dispositivo generado para su PDA es *lddev/pilot*, administrado por *udev*. Si necesita especificar manualmente un puerto para su handheld, tiene que modificar las reglas de *udev*, no cambiar directamente el archivo *dev/pilot*.

Sistemas de administración de bases de datos

El software de bases de datos suele organizarse en tres categorías: SQL, Xbase y bases de datos de escritorio. Las *bases de datos de SQL* son bases de datos relacionales de nivel profesional, cuyos archivos se administran por un programa servidor de base de datos central. Las aplicaciones usando la base de datos no acceden a esos archivos directamente. En cambio, envían solicitudes al servidor de base de datos, que después realiza el acceso real. SQL es un lenguaje de consultas designado para estas bases de datos de tamaño industrial. Los proyectos son de fuente abierta y están disponibles gratuitamente. En la tabla 11-6 se muestra una lista de sistemas de administración de bases de datos (DBMS, *DataBase Management Systems*) disponibles para Linux.

El *lenguaje Xbase* es una versión mejorada del lenguaje de programación dBase, utilizado para acceder a archivos de base de datos cuyos formatos fueron desarrollados originalmente por dBase en PC. Con Xbase, los DBMS tienen acceso directo a archivos de base de datos. Xbase se utiliza principalmente en bases de datos personales más pequeñas, con archivos localizados en un sistema propiedad de un usuario.

Bases de datos SQL (RDMS)

Las bases de datos SQL son sistemas de administración de base de datos relacionales (RDMS, *Relational Database Management System*) diseñadas para tareas de administración de bases de datos

Sistema	Sitio
PostgreSQL	La base de datos PostgreSQL: postgresql.org
MySQL	Base de datos MySQL: mysql.com
Oracle	Base de datos Oracle: oracle.com
Sybase	Base de datos Sybase: sybase.com
DB2	Base de datos de IBM: software.ibm.com/data/db2/linux
Informix	Base de datos Informix: informix.com/linux
MaxDB	Base de datos Oracle: mysql.com
GNU SQL	La base de datos GNU SQL: ispras.ru/~kml/gss
Flagship	Interfaz para archivos de base de datos Xbase: fship.com/free.html
Xbase	Herramientas y bibliotecas de Xbase: linux.techass.com/projects/xdh

TABLA 11-6 Sistemas de administración de base de datos para Linux

extensas. Algunas de las bases de datos SQL ahora con versiones para Linux son Oracle, Informix, Sybase e IBM (pero no Microsoft, por supuesto). Se trata de sistemas de administración de bases de datos profesionales y comerciales de orden superior. Linux ha probado que puede apoyar tareas de administración de bases de datos complejas y demandantes. Además, muchas bases de datos SQL gratuitas disponibles para Linux, ofrecen casi la misma funcionalidad. Gran parte de las bases de datos comerciales también ofrecen versiones personales gratuitas, como hace Oracle, Adabas D y MySQL.

PostgreSQL

PostgreSQL se basa en DBMS POSTGRES, aunque utiliza SQL como lenguaje de consulta. POSTGRES es un prototipo de investigación de nueva generación, desarrollado en la Universidad de California, en Berkeley. Las versiones Linux de PostgreSQL se incluyen en casi todas las distribuciones. Encontrará más información acerca de PostgreSQL en su sitio postgresql.org. PostgreSQL es un proyecto de fuente abierta desarrollado bajo licencia GPL.

MySQL

MySQL es un servidor de base de datos SQL multiusuario y multiprocesos, con soporte de MySQL AB. MySQL es un producto de fuente abierta disponible de manera gratuita bajo licencia GPL. Obtendrá información actual en el sitio mysql.com. El sitio incluye documentación detallada, abarcando manuales y FAQ.

Oracle

Oracle ofrece una versión totalmente funcional de su DBMS Oracle9i para Linux, además del Oracle Application Server. Descargue una versión de prueba del sitio Web de Oracle, oracle.com. Oracle es un DBMS profesional para bases de datos grandes, diseñado específicamente para tareas de negocios electrónicos en Internet. Oracle Application Server proporciona soporte a aplicaciones en tiempo real y comerciales en Web. Debido a que Linux es una versión completamente funcional de Unix, Oracle es particularmente efectivo en Linux. Fue diseñado originalmente para operar en Unix y Linux es mucho mejor plataforma para éste que cualquier otro sistema operativo de PC.

Oracle ofrece documentación extensa para su versión de Linux; puede descargarla de su página de documentación, a la que se vincula desde las páginas de soporte técnico en su sitio Web. Entre la

documentación disponible se incluye una guía de instalación, una referencia para el administrador y notas de versión, así como documentación genérica. Encontrará información específica sobre la instalación y configuración de Oracle para Linux en la base de datos de Oracle, HOWTO.

Informix

Informix (ahora controlado por IBM) ofrece una plataforma integrada para aplicaciones de Internet llamada Informix Internet Foundation.2000 en Linux. Incluye Informix Dynamic Server, su base de datos de servidor. Informix Dynamic Server presenta Dynamic Scalable Architecture, capacitada para usar de manera efectiva cualquier configuración de hardware. Informix sólo proporciona productos comerciales. No existen versiones gratuitas, aunque la compañía ofrece promociones especiales para productos Linux. Encontrará más información acerca de Informix en www-4.ibm.com/software/data/infomix. Informix apoya fuertemente el desarrollo para Linux en su línea Informix. Encontrará más información acerca de Informix para Linux en www-306.ibm.com/software/data/infomix/linux.

Sybase

Para Linux, Sybase ofrece el servidor Sybase Adaptive Server Enterprise (visite sybase.com). Puede descargar el servidor Adaptive Server Enterprise desde el sitio Web. La base de datos Sybase Enterprise presenta integración de datos para coordinar toda la información de recursos de una red. SQL Anywhere es un sistema de base de datos diseñado para bases de datos pequeñas, aunque con el mismo nivel de complejidad encontrado en las más grandes.

DB2

IBM proporciona una versión para Linux de su software DB2 Universal Database. Se descarga de manera gratuita de la página Web de IBM DB2 para Linux, software.ibm.com/data/db2/linux. DB2 Universal Database para Linux incluye funcionalidad de Internet, junto con soporte para Java y Perl. Con Web Control Center, los administradores mantienen sus bases de datos desde un explorador Web. DB2 presenta escalabilidad para expandir la base de datos de manera sencilla, soporte para objetos grandes binarios y optimización basada en costos para acceso rápido. DB2 es todavía una base de datos para mainframe, aunque IBM trabaja para refinar sus versiones Unix y Linux.

MaxDB

MaxDB es una base de datos certificada y desarrollada originalmente por SAP. Proporciona capacidades equiparables a muchas bases de datos profesionales. MaxDB ahora es desarrollada por el proyecto MySQL, mysql.com. Recientemente, el proyecto MySQL también agregó MAX DB, antes denominado SAP DB.

GNU SQL

GNU SQL es la base de datos relacional GNU desarrollada por un grupo del Instituto para Programación de Sistemas, de la Academia Rusa de Ciencias, respaldado por la organización GNU. Es una DBMS multiusuario portátil con estructura cliente/servidor soportando SQL. El servidor procesa solicitudes y realiza operaciones administrativas básicas, como la desactivación de porciones frecuentemente inactivas de una base de datos. Los clientes pueden residir en cualquier computadora de una red local. GNU SQL utiliza un dialecto SQL basado en el estándar SQL-89, diseñado para utilizarse en un ambiente similar a Unix. Puede descargar el software de la base de datos del sitio FTP de GNU en ftp.gnu.org. Para informarse más, visite el sitio Web de GNU SQL en ispras.ru/~kml/gss.

Bases de datos Xbase

Las bases de datos a que se tiene acceso con Xbase son de pequeña escala y diseñadas para redes de dimensiones moderadas o para uso personal. Muchos son programas de bases de datos para PC, como dBase III, Clipper, FoxPro y Quicksilver. Actualmente, sólo Flagship proporciona una interfaz para acceder a los archivos de la base de datos Xbase.

Flagship es un compilador con el que pueden crear interfaces para consultar archivos de bases de datos Xbase. Las interfaces presentan menús y cuadros de diálogo; integran llamadas a función que ejecutan ciertas consultas de bases de datos. Flagship puede compilar el código de dBase III+ y superior. Es compatible con dBase y Clipper, además de soportar acceso a casi todos los formatos de archivo Xbase, como **.dbf**, **.dbt**, **.fmt** y **.frm**. Una característica clave de Flagship es que sus interfaces pueden adjuntarse a una página Web, permitiendo que los usuarios actualicen la base de datos. Flagship es software comercial, aunque puede descargar una versión gratuita personal del sitio Web fship.com/free.html.

Editores

Tradicionalmente, la mayor parte de distribuciones de Linux instalan editores Vim y Emacs, basados en interfaz de cursor. *Vim* es una versión mejorada del editor de texto *Vi*, usado en el sistema Unix. Estos editores emplean operaciones simples de cursor para dar un formato de pantalla completa. Estos editores se inician desde la línea de comandos shell, sin ningún tipo de soporte de X Window System. En este modo, las operaciones de cursor no tienen la misma facilidad de uso que suele encontrarse en los editores de ventana. No existen características de menús, barras de desplazamiento o clic con el ratón. Sin embargo, K Desktop y GNOME soportan editores de texto con GUI poderosos que incluyen estas características. Estos editores operan de la misma forma que los encontrados en sistemas Macintosh y Windows. Tienen soporte completo para ratón, barras de desplazamiento y menús. Tal vez resulten mucho más fáciles de utilizar que los editores *Vi* y Emacs. Estos editores operan desde sus escritorios respectivos, por ello es necesario que primero tenga KDE o GNOME instalado, aunque los editores pueden ejecutarse en cualquier escritorio. *Vi* y Emacs tienen características de edición poderosas refinadas con el paso de los años. Emacs, en particular, puede extenderse a un entorno de desarrollo completo para programar nuevas aplicaciones. Las nuevas versiones de Emacs, como GNU Emacs y XEmacs, sustentan al X Window System con operaciones de ratón, menús y ventana. Se ejecutan en cualquier administrador de ventana o escritorio. Además, la versión *gvim* del editor Vim, también respalda operaciones básicas de ventana. Puede acceder a éste desde ambos escritorios, GNOME y KDE. En la tabla 11-7 se muestra una lista con varios editores con interfaz GUI para Linux.

Editor de GNOME: Gedit

Gedit es un editor de texto básico para el escritorio GNOME. Sustenta soporte completo para ratón, implementando operaciones estándar de GUI, como cortar y pegar para mover texto, hacer clic y arrastrar para seleccionarlo. Respalda operaciones de edición de texto estándar, como Buscar y Reemplazar. También puede utilizar Gedit para crear y modificar sus archivos de texto, incluida la configuración de archivos. Gedit también ofrece características más avanzadas, como vista previa de impresión y niveles de operaciones de deshacer y rehacer, además de leer datos de canalizaciones. Presenta un menú que proporciona funcionalidad agregada e incluye plug-ins para revisor ortográfico, codificación, correo electrónico y despliegue de páginas Web de texto.

K Desktop	Description
KEdit	Editor de texto
Kate	Editor de texto y programas
KJots	Editor de bloc de notas
KWord	Editor de publicaciones de escritorio, parte de KOffice
GNOME	
Gedit	Editor de texto
AbiWord	Procesador de palabras
X Window System	
GNU Emacs	Editor Emacs con soporte de X Window System
XEmacs	Versión para X Window System del editor Emacs
gvim	Versión de Vim con soporte de X Window System (vim-x11)
OpenWriter	Procesador de palabras de OpenOffice.org que permite la edición de archivos de texto

TABLA 11-7 Editores de escritorio Linux

Editores de K Desktop: Kate, KEdit y KJots

Todos los editores de K Desktop proporcionan soporte completo a ratón, implementando operaciones estándar de GUI, como cortar y pegar para mover texto, así como hacer clic y arrastrar para seleccionarlo. Kate es un editor avanzado, con características como revisor ortográfico, selección de fuentes y resaltado. Casi todos los comandos pueden seleccionarse mediante menús. Una barra de herramientas de iconos para operaciones comunes se despliega a lo largo de la parte superior de la ventana de Kate. Una barra lateral muestra paneles para un selector de archivos y una lista de archivos. Con el selector de archivos, puede recorrer el sistema de archivos para seleccionar aquellos con los que desea trabajar. Kate también soporta vistas múltiples de un documento, permitiéndole desplegar segmentos de sus propias ventanas, de manera vertical u horizontal. También puede abrir más de un documento simultáneamente, desplazándose entre ellos con la lista de archivos. Kate está diseñado para ser un editor para archivos de código fuente relacionados con desarrollo/programación de software. Aunque Kate carece de todas las características de Emacs o Vi, puede manejar la mayor parte de tareas importantes. Kate respeta la sintaxis para diferentes lenguajes de programación como C, Perl, Java y XML. Además, tiene capacidad para acceder y editar archivos en un sitio FTP o Web.

KEdit es un editor de texto más antiguo y simple, hecho para editar archivos de texto simples como los de configuración. Una barra de herramientas de botones, en la parte superior de la ventana de KEdit, permite ejecutar comandos de edición comunes de manera sencilla, con un solo clic de ratón. Con KEdit, también puede enviar por correo electrónico archivos en edición. La entrada para KEdit en el menú K se muestra sólo como Editor de texto.

El editor KJots está diseñado para tomar apuntes en un bloc de notas. Organiza escritos en los blocs, denominándolos simplemente *libros*. Seleccione el que quiera ver o agregar desde el menú Libros. Para iniciar KJots, seleccione su entrada en el menú Utilidades | Pim o inserte el comando **kjots** en una ventana de terminal.

El editor Emacs

Emacs puede describirse mejor como un entorno de trabajo presentando un editor, un programa de correo electrónico y así como lector de noticias, además de intérprete Lisp. El editor está hecho a la medida para desarrollo de programas, lo que permite formar código fuente de acuerdo con el lenguaje de programación usado. Hay muchas versiones de Emacs disponibles para usarse en sistemas Unix y Linux. Las versiones actuales, incluidas con distribuciones de Linux son GNU Emacs o XEmacs. La versión actual para GNU Emacs es 20.x; compatible con X Window System, permite características GUI como menús, barras de desplazamiento y operaciones de edición con el ratón. Revise los sitios FTP de actualización para su distribución en busca de nuevas versiones, a medida que se presenten, además del sitio Web de GNU en gnu.org y el sitio de Emacs en emacs.org. Encontrará más información acerca de XEmacs en su sitio Web, xemacs.org.

Emacs deriva gran parte de sus opciones y flexibilidad de la capacidad para manipular búferes. Emacs podría describirse como un editor orientado a búfer. Cuando edita un archivo en cualquier editor, el archivo se copia en un búfer (memoria intermedia) de trabajo. Emacs puede administrar muchos búferes de trabajo a la vez, permitiendo la edición de varios archivos al mismo tiempo. Puede editar los búferes almacenando texto copiado o eliminado. Incluso puede crear búferes propios, llenarlos con texto y después guardarlos en un archivo. Emacs extiende el concepto de búfer para cubrir muchas tareas. Cuando cree un correo, abra un búfer de correo; cuando lea noticias, abra un búfer de noticias. Para cambiar de una tarea a otra basta con ir de un búfer a otro.

El editor Emacs opera casi como un procesador de palabras estándar. Las teclas representan caracteres de entrada. Los comandos se implementan con teclas especiales, como las teclas de control (CTRL) y alternar (ALT). No existe modo de entrada especial, como en Vi o Ed. Se inserta el texto y si necesita ejecutar un comando de edición, como mover el cursor o guardar texto, utilice la tecla CTRL. Este tipo de organización hace que el editor Emacs sea fácil de usar. Sin embargo, Emacs es todo menos simple (es un editor sofisticado y flexible con varios cientos de comandos). Emacs también tiene características especiales, como ventanas múltiples. También puede desplegar dos ventanas para texto simultáneamente, además de abrir y trabajar en más de un archivo a la vez, desplegando cada uno en su propia ventana. El editor Emacs se invoca con el comando **emacs**. Puede insertar el nombre del archivo que quiera editar y, si el archivo no existe, se crea. En el siguiente ejemplo, el usuario se prepara para editar el archivo **misdatos** con Emacs:

```
$ emacs misdatos
```

El editor GNU Emacs da soporte ahora a una interfaz gráfica de usuario para X Window System. Para permitir el soporte de X, inicie Emacs con un entorno X Window System, como el escritorio de KDE, GNOME o XFce. Se permiten operaciones básicas de edición en GUI: selección de texto con las operaciones de hacer clic y arrastre; corte, copiado y pegado; y una barra de desplazamiento para desplazarse por el texto. Las áreas Mode line y Echo se despliegan en la parte inferior de la ventana, donde puede insertar comandos de teclado. La barra de desplazamiento se localiza en el extremo izquierdo. Para mover la barra de desplazamiento hacia abajo, haga clic en ésta con el botón izquierdo del ratón. Para mover la barra hacia arriba, haga clic en ésta con el botón derecho del ratón.

NOTA XEmacs es un editor Emacs completo con una interfaz gráfica de usuario y aplicaciones de Internet, incluidos explorador Web, utilería de correo y lector de noticias.

El editor Vi: Vim y Gvim

El editor Vim incluido con casi todas las distribuciones de Linux es una versión mejorada del editor Vi. Incluye todos los comandos y las características de éste. Vi, cuyo nombre surge de *visual*, sigue

siendo uno de los editores más utilizados en Linux. Los editores de teclado como Vim y Emacs dependen del teclado para dos diferentes operaciones: especificar comandos de edición y recibir entrada de caracteres. Utilizadas para editar comandos, ciertas teclas realizan eliminaciones, algunas ejecutan cambios y otras realizan movimientos de cursor. Empleadas para entrada de caracteres, las teclas indican tipos que pueden ingresarse al archivo en edición. Generalmente, estas funciones se dividen entre diferentes teclas. Las alfabéticas se reservan para entrada de caracteres, mientras las de función y control especifican comandos de edición, como eliminar texto o mover el cursor. Tales editores llegan a depender de la existencia de un teclado extendido incluyendo teclas de función y control. Sin embargo, los editores en Unix fueron diseñados para trabajar con un teclado mínimo, caracteres alfanuméricos y unos cuantos caracteres de control, además de teclas ESC y ENTER. En lugar de dividir las funciones de comando e inserción entre diferentes teclas, el editor Vi tiene tres modos separados de operación para el teclado: modos de comandos e inserción, además de edición de línea. En el modo *comando*, todas las teclas en el teclado se vuelven comandos de edición; en el modo *inserción*, las teclas se vuelven caracteres de entrada. Algunos de los comandos de edición como **a** o **i**, entran al modo de inserción. Al insertar **i**, deja el modo comando y entra al modo de inserción. Cada tecla representa un carácter a insertarse en el texto. Oprima ESC para regresar automáticamente al modo de comandos: las teclas se volverán de nuevo comandos del editor. A medida que inserta texto, se mueve constantemente de un modo de comando a uno de inserción y viceversa. Con Vim, puede utilizar el comando CTRL-O para saltar rápido al modo de comandos e insertar un comando y después regresar automáticamente al modo de inserción. En la tabla 11-8 se muestra una lista de un conjunto muy básico de comandos de Vi para empezar a trabajar en él.

Comando	Movimiento de cursor
h	Mueve el cursor a la izquierda un carácter.
l	Mueve el cursor a la derecha un carácter.
k	Mueve el cursor una línea arriba.
j	Mueve el cursor una línea abajo.
CTRL-F	Avanza una pantalla de texto; se despliega la siguiente pantalla de texto.
CTRL-B	Retrocede una pantalla de texto; se despliega la pantalla previa de texto.
Input	<i>Todos los comandos de inserción colocan al usuario en inserción; el usuario deja este modo con ESC.</i>
a	Inserta la entrada después del cursor.
i	Inserta la entrada antes del cursor.
o	Inserta la entrada bajo la línea donde está el cursor; inserta una nueva línea vacía bajo un cursor encendido.
Selección de texto (Vim)	Movimiento de cursor
v	Modo visual; mueve el cursor para expandir el texto seleccionado por carácter. Una vez seleccionado, oprima la tecla para ejecutar la acción: c cambia, d elimina, y copia, : comando de edición de línea, J líneas de unión, U mayúsculas, u minúsculas.

TABLA 11-8 Comandos del editor Vi

Selección de texto (Vim)	Movimiento de cursor
V	Modo visual: mueve el cursor para expandir el texto seleccionado por línea.
Delete	Efecto
x	Elimina el carácter donde está el cursor.
dd	Elimina la línea donde está el cursor.
Change	<i>(Excepto por el comando para remplazar, r, todos los comandos de cambio colocan al usuario en la entrada después del texto de eliminación.)</i>
cw	Elimina la palabra donde está el cursor y coloca al usuario en el modo de inserción.
r	Reemplaza el carácter donde está el cursor. Después de oprimir r, el usuario inserta el carácter de reemplazo. El cambio se hace sin insertar una entrada; el usuario permanece en el modo de comandos de Vi.
R	Primero se coloca en el modo de inserción, después sobrescribe carácter por carácter. Aparece como un modo de sobreescritura en la pantalla, pero realmente está en el modo de inserción.
Move	<i>Mueve el texto principal al eliminarlo primero, mover el cursor al lugar deseado de inserción y después oprimir el comando p. (Cuando el texto se elimina, se almacena automáticamente en un búfer especial.)</i>
p	Inserta texto eliminado o copiado después del carácter o línea donde está el cursor.
P	Inserta texto eliminado o copiado antes del carácter o línea donde está el cursor.
dw p	Elimina una palabra, después se mueve al lugar que indicó con el cursor (oprima p para insertar la palabra después de la palabra donde está el cursor).
yy o Y p	Copia la línea donde está el cursor.
Search	<i>Los dos comandos de búsqueda abren una línea en la parte inferior de la pantalla y permiten al usuario insertar un patrón de búsqueda; oprima enter después de escribir el patrón.</i>
/patrón	Busca un patrón hacia delante, en el texto.
?patrón	Busca un patrón hacia atrás, en el texto.
n	Repite búsquedas previas, ya sea hacia adelante o atrás.
Comandos de edición de línea	Efecto
w	Guarda el archivo.
q	Sale del editor; q! sale sin guardar.

TABLA 11-8 Comandos del editor Vi (continuación)

Aunque el modo de comandos Vi maneja casi todas las operaciones de edición, no puede realizar algunas, como guardar archivos o realizar sustituciones globales. Para tales operaciones, debe ejecutar comandos de edición de línea. Entre al modo de edición de línea con el comando de dos puntos de Vi. Los dos puntos son un comando especial que activa operaciones de edición de una

línea. Cuando escribe los dos puntos, una línea se abre en la parte inferior de la ventana, con el cursor colocado al principio de la línea. Ahora está en el modo de edición de línea. En este modo, debe insertar un comando de edición en una línea, oprimir **ENTER**, y entonces el comando se ejecutará. La entrada a este modo suele ser sólo temporal. Al oprimir **ENTER**, regresa automáticamente al modo de comandos de Vi y el cursor regresa a su posición anterior en la pantalla.

Aunque puede crear, guardar, cerrar y salir de archivos con el editor Vi, los comandos para cada uno no son muy similares. Para guardar y salir de un archivo se requiere el uso de comandos de edición de línea especiales, mientras para cerrar un archivo se requiere un comando de edición Vi. Para editar un archivo, escriba **vi** o **vim** y el nombre del archivo en la línea de comandos de shell. Si no existe un archivo con ese nombre, el sistema lo creará. En efecto, al señalar el nombre de un archivo inexistente, se indica al editor Vi que cree un archivo. El siguiente comando invoca al editor Vi, que trabaja con el archivo **listadelibros**. Si **listadelibros** no existe, el editor Vi lo crea.

```
$ vim listadelibros
```

Después de ejecutar el comando **vim**, entrará al modo de comandos de Vi. Cada clave se vuelve un comando de edición Vi, y la pantalla una ventana dentro del archivo de texto. El texto se despliega pantalla por pantalla. La primera pantalla del texto se muestra y el cursor se coloca en la esquina superior izquierda. Con un archivo nuevo, no existe texto para desplegar. Este hecho se indica con una columna de tildes en la sección izquierda de la pantalla. Las tildes representan la porción de una pantalla que no pertenece a un archivo.

Recuerde que cuando entra por primera vez al editor Vi, se encuentra en el modo de comandos. Para insertar texto, debe entrar al modo de inserción. En el modo de comandos, **a** es de edición para adjuntar texto. Oprimiendo esta tecla se le coloca en modo inserción. Ahora el teclado opera como máquina de escribir y puede ingresar texto en el archivo. Si oprime **ENTER**, simplemente inicia una nueva línea de texto. Con Vim, puede utilizar las teclas de flechas para moverse de una parte del texto insertado a otra y trabajar en diferentes partes del texto. Después de insertar texto, puede salir del modo de inserción y regresar al de comandos al oprimir **ESC**. Una vez que termine con la sesión de edición, salga de Vi escribiendo dos Z mayúsculas, **ZZ**. Mantenga oprimida la tecla **MAYÚS** y oprima **z** dos veces. Esta secuencia guarda primero el archivo y después lo saca del editor Vi, regresándolo a la shell de Linux. Para guardar un archivo mientras edita, use el comando de edición de línea **w**, que escribe un archivo en el disco; **w** es equivalente al comando Guardar encontrado en otros procesadores de palabras. Primero escriba dos puntos para acceder al modo de edición de línea y después escriba **w** y oprima **ENTER**.

Puede utilizar el comando **:q** para salir de una sesión de edición. A diferencia del comando **ZZ**, el comando **:q** no realiza operación de guardado antes de salir. En cuanto a esto, sólo tiene una restricción importante. Si se ha hecho cualquier modificación a su archivo desde la última operación de guardado, el comando **q** fallará y no saldrá del editor. Sin embargo, puede sobrescribir esta restricción si coloca un calificador **!** después del comando **:q**. El comando **:q!** Lo sacará del editor Vi sin guardar modificaciones hechas al archivo en esa sesión desde el último guardado (la combinación **:wq** es igual que **ZZ**).

Para obtener ayuda en línea, inserte el comando **:help**. Se trata de un comando de edición de línea. Escriba dos puntos, inserte la palabra **help** en la línea que se abre en la parte inferior de la pantalla y después oprima **ENTER**. Puede agregar el nombre de un comando específico tras la palabra **help**. La tecla **F1** también inicia la ayuda en línea.

Como una opción al uso de Vim en una interfaz de línea de comandos, utilice **gvim**, que proporciona menús basados en X Window System para operaciones de archivo, edición y ventana básicas. **Gvim** se instala como el paquete **vim-x11**, que incluye varios vínculos a **gvim** como **evim**,

gview y **gex** (línea de editor Ex abierto). Para utilizar **gvim**, puede seleccionarlo desde su menú principal de la distribución, o insertar el comando **gvim** en el indicador de comandos de terminal de X Window System.

Se despliega la interfaz Vi estándar, pero con varios botones de menú a lo largo de la parte superior, junto con una barra de herramientas con botones para comandos comunes como búsqueda y guardado de archivo. Todos los comandos de Vi estándar trabajan como se describió antes. Sin embargo, puede utilizar su ratón para seleccionar elementos en estos menús. Puede abrir y cerrar un archivo, o abrir varios archivos al utilizar ventanas divididas o ventanas diferentes. El menú de edición le permite cortar, copiar y pegar texto, al igual que operaciones para deshacer y rehacer. En el modo de edición puede seleccionar texto con su ratón en una operación haciendo clic y arrastrando, o usar el menú Edición para cortar o copiar y después pegar el texto seleccionado. Sin embargo, la inserción de texto todavía se realiza con los comandos **a**, **i** o **o** para entrar en el modo de inserción. Se permiten búsquedas y reemplazos a través de una ventana de diálogo. También hay botones en la barra de herramientas para encontrar instancias anteriores y siguientes. Gview también presenta soporte a programación, con codificación en color para sintaxis de programación, para secuencias de comandos de shell y programas de C++. Incluso existe un botón Make para ejecutar Makefiles.

Herramientas gráficas y multimedia

Linux soporta una amplia gama de aplicaciones y herramientas gráficas, además multimedia, algunas con la simplicidad del visor para imágenes KView; sofisticados programas de manipulación de imágenes como GIMP; reproductores de música y CD, entre los que figuran Rhythmbox; asimismo, visores de TV como Totem. En la tabla 12-2 se muestra una lista de herramientas gráficas disponibles para usarse en Linux. Además, existe abundante soporte para tareas multimedia, desde video y DVD, hasta edición de audio y música (véase la tabla 12-3, en páginas posteriores). Miles de proyectos multimedia y gráficos, junto con proyectos estándar, se encuentran bajo desarrollo o están disponibles actualmente en depósitos de distribución en línea, como sourceforge.net, freshmeat.net, o freshrpms.net de Fedora (véase la tabla 12-1). Asegúrese de consultar el sitio sourceforge.net para buscar cualquier tipo de aplicación que necesite.

NOTA El soporte para distintas operaciones multimedia populares, sobre todo MP3, DVD y DivX, no se incluyen con muchas distribuciones, incluidos Fedora y Red Hat, debido a restricciones de licencias y otros factores. Para reproducir archivos MP3, DVD y DivX, debe descargar e instalar manualmente paquetes de soporte. En el caso de Fedora, están disponibles los paquetes binarios Red Hat Package Manager (RPM), precompilados para muchas aplicaciones y bibliotecas populares, como MPlayer y Xvid, además de soporte a MP3 y video DVD. Se encuentran en rpm.livna.org y freshrpms.net.

Herramientas gráficas

GNOME, KDE y X Window System dan soporte a una cantidad impresionante de herramientas gráficas, incluidos visores de imágenes, capturadores de ventanas, editores de imágenes y herramientas de dibujo. Estas herramientas pueden encontrarse, en los escritorios de KDE y GNOME, bajo un submenú Gráficos o el menú Utilidades .

NOTA Linux se ha convertido en la plataforma preferida para muchas tareas multimedia a nivel profesional, como la creación de imágenes generadas por computadora (CGI) y animaciones para efectos especiales de películas, usando software como Maya y Softimage. Entre las bibliotecas gráficas de Linux incluyen para OpenGL, MESA y SGI.

Proyecto	Descripción y sitio
SourceForge.net	Este sitio almacena gran cantidad de software multimedia para Linux, la mayor parte todavía en desarrollo: sourceforge.net
Aplicaciones multimedia de KDE	KDE da soporte a un extenso conjunto de aplicaciones de software multimedia: kde-apps.org
Aplicaciones multimedia de GNOME	Muchas aplicaciones multimedia se han desarrollado para GNOME: gnomefiles.org
Software de audio y MIDI para Linux	Este sitio muestra una lista de la amplia variedad de software multimedia y audio: linux-sound.org
Advanced Linux Sound Architecture (ALSA)	El proyecto (ALSA) está bajo desarrollo en Linux, acorde con los lineamientos GPL: alsa-project.org
Open Sound System	Open Sound System soporta un amplio rango de aplicaciones multimedia: opensound.com

TABLA 12-1 Proyectos y sitios de multimedia para Linux

Herramientas de administración de fotografías: F-Spot y digiKam

F-Spot Photo Manager proporciona una manera simple y poderosa para administrar, desplegar e importar sus fotografías e imágenes (**f-spot.org**). Las fotografías pueden organizarse en diferentes categorías: eventos, personas y lugares. Puede realizar operaciones de despliegue estándar como rotación y vista en pantalla completa, junto con presentaciones. Se proporciona soporte para edición de imagen. Las fotos seleccionadas pueden grabarse directamente en un CD (recurriendo a las capacidades de grabación de Nautilus).

Entre las cualidades se incluye una interfaz fácil de usar. Un servicio de cronograma permite ver las fotografías en el orden que se tomaron. También puede desplegar fotografías en modo de pantalla completa o como presentación.

F-Spot incluye un editor de fotografías para ajustes básicos y cambios como rotación, corrección de ojos rojos y configuraciones de color estándar, incluidas temperatura y saturación.

Etiquete las fotografías y colóquelas en grupos, facilitando su acceso. Con una etiqueta, marque una colección de fotografías y después utilice la etiqueta para acceder a éstas de manera instantánea. La etiqueta en sí, puede ser un icono seleccionado por el usuario, incluido alguno creado por el usuario con el editor de iconos Tag.

F-Spot ofrece varias formas de subir fotos a un sitio Web. Proporciona acceso directo a la cuenta Flickr (**flickr.com**) o sitios soportados por Gallery (**gallery.menalto.com**). Las fotografías también pueden guardarse en una carpeta para después subirlas a un sitio Web, ya sea como archivos simples o HTML estáticos.

digiKam es un administrador de fotografías de KDE con muchas de las mismas características (**digiKam.org**). Un panel lateral permite acceder de manera sencilla por álbum, fecha, etiquetas o búsquedas previas. digiKam también cuenta con capacidades de edición de imágenes, con gran cantidad de efectos. La configuración de digiKam (el menú Preferencias) facilita muchas opciones, comprendiendo edición de imágenes, soporte para cámara digital y configuración de interfaz.

Herramientas gráficas de KDE

KView es un visor de imágenes simple para archivos de imagen GIF y JPEG. El programa KSnapshot es un capturador de pantallas simple, también para KDE; actualmente soporta sólo unos cuantos formatos de imágenes. KFourier es una herramienta para procesar imágenes que recurre a la transformada de Fourier para aplicar varios filtros a una imagen simultáneamente. KuickShow es un explorador y visor de imágenes, fácil y cómodo de usar que permite presentaciones, al igual que muchos formatos de imágenes basados en imlib. KolourPaint es un programa de dibujo simple con efectos de pinceles, formas y colores; da soporte a numerosos formatos. Krita es la aplicación de dibujo y edición de imágenes profesional de KOffice, con gran variedad de características, como creación de imágenes Web y modificación de fotografías (antes conocida como Krayon y KImageShop).

Herramientas gráficas de GNOME

GNOME presenta varias herramientas gráficas poderosas y fáciles de usar. Algunas vienen instaladas con Linux, aunque puede descargar otras, como GView y gtKam, de **gnomefiles.org**. Además, muchas herramientas de KDE trabajan muy bien en GNOME y pueden accederse desde el escritorio de GNOME.

La aplicación gThumb es un visor de imágenes en miniatura que le permite explorar y desplegar imágenes, mediante miniaturas y organizarlas en catálogos para una referencia sencilla. Consulte **sourceforge.net** para conocer más información.

GIMP es el GNU Image Manipulation Program, una aplicación de imágenes sofisticada muy parecida a Adobe Photoshop. Use GIMP para tareas como retoque de fotografías, composición y creación de imágenes. Entre las características que ofrece hay trabajo con capas, canales, mezclas y degradados. GIMP usa, en particular, el conjunto de widgets GTK+. Encontrará más información acerca de GIMP y podrá descargar las versiones más actuales del sitio Web, en **gimp.org**. GIMP se distribuye de manera gratuita bajo la licencia pública de GNU.

Inkscape es una aplicación gráfica de vectores, basada en GNOME, para imágenes SVG (gráficos vectoriales que pueden cambiar de tamaño sin deterioro de la calidad). Presenta capacidades similares a las aplicaciones gráficas vectoriales profesionales como Adobe Illustrator. El formato SVG permite la generación rápida de imágenes para uso en Web y trabajos artísticos complejos. Aunque su formato nativo es SVG, también exporta a formato PNG. Incluye capas, además de permitir la creación sencilla de objetos, incluidas estrellas y espirales. Una barra de color habilita el cambio rápido de rellenos de color.

El proyecto gPhoto ofrece software para acceder cámaras digitales (**gphoto.org**). Una biblioteca base, llamada **libgphoto2**, proporciona varias interfaces. Esta biblioteca consta de controladores y herramientas que pueden acceder a gran número de cámaras digitales.

Programas gráficos de X Window System

Las aplicaciones basadas en X Window System se ejecutan directamente en éste, soportando escritorios complejos como GNOME y KDE. Estas aplicaciones tienden a ser más sencillas, careciendo de la funcionalidad de escritorio encontrada en GNOME o KDE. Xpaint es un programa de dibujo muy similar a MacPaint, que permite cargar gráficos o imágenes y después crear formas, agregar texto y colores, además de usar herramientas de pincel con varios tamaños y colores. Xfig es un programa de dibujo y Xmorph permite modificar imágenes, cambiando sus formas. ImageMagick permite convertir imágenes de un formato a otro; por ejemplo, el cambio de una imagen TIFF a JPEG. En la tabla 12-2 se muestra una lista de las herramientas gráficas más populares para Linux.

Herramientas	Descripción
Administración de fotografías	
F-Spot	Aplicación de cámara digital y administrador de biblioteca de imágenes de GNOME (f-spot.org)
digiKam	Aplicación de cámara digital y administrador de biblioteca de imágenes de KDE (digikam.org)
KView	Visor de imágenes simple para archivos de imagen GIF y JPEG
ShowFoto	Visor de imágenes simple, trabaja con digiKam (digikam.org)
KSnapshott	Capturador de pantallas
KFourier	Herramienta de procesamiento de imágenes que utiliza el transformador Fourier
KuickShow	Explorador y visor de imágenes
KolourPaint	Programa de dibujo
Krita	Editor de imágenes (koffice.org/krita)
GNOME	
gThumb	Explorador, visor y catalogador de imágenes (gthumb.sourceforge.net)
GIMP	GNU Image Manipulation Program (gimp.org)
Inkscape	Aplicación de imágenes vectoriales de GNOME (inkscape.org)
X Window System	
Xpaint	Programa de dibujo
Xfig	Programa de dibujo
Xmorph	Herramienta que modifica imágenes
Xfractals	Generador de imagen de fractal
ImageMagick	Herramienta de edición y conversión de formato de imagen

TABLA 12-2 Herramientas gráficas para Linux

Multimedia

Están disponibles muchas aplicaciones multimedia para audio y video, incluidos editores de audio, reproductores MP3 y reproductores de video (véase la tabla 12-3). Entre las aplicaciones de audio se incluyen mezcladoras, herramientas de audio digital, escritores para CD de audio, reproductores MP3 y soporte de audio en red. Existen, literalmente, miles de proyectos bajo desarrollo en **sourceforge.net**. Si busca una aplicación específica, las posibilidades apuntan a que la encontrará ahí. Los proyectos actuales incluyen un reproductor de video completamente integrado, una grabadora de video digital y un mezclador de audio digital. Muchas aplicaciones diseñadas específicamente para la interfaz GNOME o KDE pueden encontrarse en sus sitios de software correspondiente (**gnomefiles.org** y **kde-apps.org**). Los paquetes binarios RPM o del administrador de paquetes Debian (DEB) precompilados, pueden descargarse e instalarse de manera sencilla desde los depósitos de distribución.

Aplicación	Descripción
Xine	Reproductor multimedia para video, DVD y audio
Rhythmbox	Administrador de música (GStreamer)
Sound Juicer	Extractor de audio de CD de GNOME (GStreamer)
Grip	Extractor de audio de CD
aKtione	Reproductor de video de KDE
Kscd	Reproductor de CD de música
Krec	Grabadora de sonidos de KDE
Kaboodle	Reproductor multimedia
GNOME CD Player	Reproductor de CD
GNOME Sound Recorder	Grabadora de sonidos
Pulse	Servidor de audio Pulse
XMMS	Reproductor de CD
Xplaycd	Reproductor de CD de música
Noatun	Reproductor multimedia de KDE
Xanim	Reproductor de animación y video
RealPlayer	Multimedia de flujo de RealMedia y RealAudio (real.com)
HelixPlayer	Versión de fuente abierta de RealPlayer (real.com)
K3b	Interfaz de escritura de CD para cdrecord, mkisofs y cdda2wav de KDE
KAudiocreator	Extractor y quemador de CD de KDE
dvdauthor	Herramientas para creación de DVD (dvdauthor.sourceforge.net)
Qauthor	Interfaz de KDE para dvdauthor (kde-apps.org)
DVDStyler	Aplicación de creación de DVD para GNOME (dvdstyler.sourceforge.net)
Fluendo	Codecs multimedia comerciales para Linux (fluendo.com)
Codec Buddy	Herramienta Codec Buddy

TABLA 12-3 Multimedia y aplicaciones de audio

Las aplicaciones multimedia manejan varios codecs para ejecutar diferentes tipos de medios, como MP3 para archivos de música. La herramienta Codec Buddy detectará el codec necesario y descargará, en caso de no estar instalado. Puede comprar codecs comerciales de terceros como Windows Media o Dolby en Fluendo (fluendo.com).

GStreamer

Muchas de las aplicaciones basadas en GNOME usan GStreamer, una plataforma basada en gráficas y filtros para difusión multimedia en tiempo real. Al utilizar una estructura de plug-in, las aplicaciones de GStreamer integran amplia variedad de tipos de medios. Puede descargar módulos y plug-ins de gstreamer.freedesktop.org. GNOME en Linux incluye varias aplicaciones de GStreamer:

- El reproductor de video Totem utiliza GStreamer para reproducir DVD, VCD y MPEG.
- Rhythmbox proporciona administración integrada de música; similar al reproductor de música iTunes de Apple.

- Sound Juicer es un extractor de música de CD
- Un reproductor de CD, grabadora de sonidos y control de volumen se ofrecen como elementos predeterminados de GStreamer en GNOME Media.

Selector de sistemas multimedia

GStreamer puede configurarse para manejar diferentes entradas y salidas, de controladores y servidores de audio y video. Estas selecciones se hacen con la herramienta propiedades de GStreamer. Para abrir esta herramienta desde el menú Escritorio, primero seleccione Preferencias, después Más preferencias y después la entrada Selector de sistemas multimedia. También puede insertar **gststreamer-properties** en una ventana de terminal. La ventana Propiedades despliega dos paneles con fichas, una para audio y otra para video. Los controladores de salida y servidores están etiquetadas Salida predeterminada, mientras los controladores de entrada muestran Entrada predeterminada. Existen menús emergentes para cada uno, presentando una lista de controladores o servidores de sonido y video. Por ejemplo, el servidor de audio usado es ALSA, pero puede cambiarlo a OSS.

Plug-ins de GStreamer: el bueno, el malo y el feo

Muchas aplicaciones multimedia de GNOME, como Totem, utilizan GStreamer para brindar soporte multimedia. Para usar características como DVD Video y MP3, debe instalar plug-ins adicionales de GStreamer. Encontrará más información acerca de GStreamer y sus paquetes de soporte en gststreamer.freedesktop.org.

Los paquetes de soporte pueden resultar confusos. Para la versión 1 y posterior, GStreamer establece cuatro paquetes de soporte diferentes llamados the base, the good, the bad y the ugly. El paquete base es un conjunto de plug-ins útiles, además de confiables. El paquete the good es un conjunto de plug-ins con soporte y probados, cumpliendo los requisitos de licencia. El paquete the bad es un conjunto de plug-ins sin soporte, cuyo desempeño no está garantizado y puede bloquearse, pero cumple los requisitos de licencia. El paquete the ugly contiene plug-ins funcionando bien, pero probablemente no cumplen los requisitos de licencia, como soporte a DVD.

- **The base** Plug-ins confiables y de uso común.
- **The good** Plug-ins confiables adicionales y útiles.
- **The ugly** Plug-ins confiables, pero sin licencia completa (soporte para DVD y MP3).
- **The bad** Plug-ins que tal vez no sean confiables pero que resultan útiles (es probable que cause bloqueos).

Compatibilidad con MP3 de GStreamer: iPod

Para que su iPod y otros dispositivos MP3 trabajen con aplicaciones GNOME como Rhythmbox, necesitará instalar soporte para MP3 de GStreamer. El soporte para MP3 no se incluye con varias distribuciones, debido a problemas de licencia. Sin embargo, tiene la opción de descargar e instalar el paquete de soporte **gststreamer-plugins-ugly** de GStreamer ya visto, que mantiene casi todos los paquetes de soporte no incluidos en la mayoría de distribuciones Linux.

Para sincronizar e importar desde su iPod, use software de administración para iPod como la interfaz GUI para iPod (gtkpod). Varias secuencias de comandos y herramientas están disponibles para operaciones relacionadas con iPod. Incluyen SyncPOD, myPod, gtkpod (una GUI para iPod) e iPod para Linux. Revise sourceforge.net y busque con la entrada “iPod”.

Aplicaciones de audio

Los dispositivos de audio en Linux funcionan debido a controladores, formando un sistema de sonido. Con el kernel actual, el soporte para audio se implementa mediante el sistema Advanced Linux Sound Architecture (ALSA, arquitectura avanzada de sonido de Linux). ALSA reemplaza la versión gratuita de Open Sound System usada en versiones anteriores, además de los controladores de sonido integrados. Encontrará más información acerca de ALSA en alsa-project.org.

MP3 con LAME

LAME significaba “Lame Ain’t an Mp3 Encoder” (Lame no es un codificador Mp3) pero hace mucho evolucionó para convertirse en un codificador MP3 completo, cuyo software está disponible bajo la licencia LPGL. Se incluye con VideoLAN y FFmpeg y se descarga para soporte de MPlayer o Xine.

Debido a problemas de licencia y patentes, muchas distribuciones de Linux han quitado el soporte para archivos MP3. La capacidad de reproducción de MP3 se ha eliminado de reproductores multimedia como XMMS y Noatun. Como una opción a MP3, puede usar la compresión Ogg Vorbis para archivos de música (vorbis.com).

Aplicaciones de música

En la actualidad hay disponibles muchas aplicaciones de música para GNOME, incluidos editores y reproductores de audio y MP3. Puede usar GNOME CD Player para reproducir música de CD y GNOME Sound Recorder para grabar fuentes de sonido. Revise el mapa del software en gnomefiles.org, para conocer los lanzamientos más recientes. También hay diversas aplicaciones disponibles para KDE, incluidos dos reproductores de medios (Kaiman y Kaboodle), una mezcladora (KMix), así como un reproductor de CD (Kscd). Visite kde-apps.org para conocer adiciones recientes. Varias aplicaciones multimedia basadas en X Window System se encuentran instaladas con casi todas las distribuciones. Entre éstas se incluyen XMMS y Xplaycd, reproductores de música en CD y Xanim, un reproductor de video y animación.

GNOME incluye el reproductor multimedia XMMS, GNOME CD Player, GNOME Sound Recorder y el Control de volumen, en el menú Sonido y Video. Las aplicaciones KDE incluyen KMidí, Kaboodle y Noatun. Los sistemas Linux también soportan Helix Player, el proyecto de fuente abierta utilizado para RealPlayer. HelixPlayer sólo ejecuta medios de fuente abierta como archivos Ogg Vorbis (aunque puede obtener codecs de audio y video para los reproductores). Visite helixcommunity.org para conocer más información. También puede descargar una copia de RealPlayer, el reproductor de flujo multimedia de Internet, desde real.com. Asegúrese de elegir RealPlayer para Unix.

El sitio Sound & Midi Software for Linux (linux-sound.org) almacena vínculos con sitios Web y FTP para muchas de sus aplicaciones de audio.

El servidor de sonido Pulse permite dirigir y administrar flujos de sonido desde dispositivos, permitiéndole dirigir y modificar sonidos para diferentes clientes.

Quemadores y extractores de CD

Diferentes programas de grabación de CD para escribir CDs de música y MP3 (quemadores y extractores), se encuentran disponibles en kde-apps.org. Entre éstos se incluyen K3b, CD-Rchive y KAudioCreator (extractor de música). Para GNOME, use CD-REC y el quemador de CD de Nautilus, integrado en el administrador de archivos Nautilus, el administrador de archivos predeterminado del escritorio de GNOME. Todos usan los programas de grabación de CD mkisofs, cdbrecord y cdda2wav, instalados como parte de su distribución. GNOME también presenta dos extractores para audio de CD, Grip y Sound Juicer.

SUGERENCIA Si su aplicación de CD o DVD tiene dificultades para encontrar el reproductor o quemador de CD/DVD, tal vez necesite revisar si HAL está creando un vínculo apropiado con su dispositivo de CD o DVD, mediante `/dev/cdrom` o `/dev/dvdrom`. Estos vínculos deben generarse automáticamente.

Aplicaciones de video

Hay varios proyectos en camino, para proporcionar soporte de TV, video, DivX, DVD y DTV para Linux (véase la tabla 12-4). En casi todos los casos, las versiones más recientes estarán en formato de código fuente en el sitio original. Para éstos deberá descargar el código fuente, que después necesitará compilar e instalar. En efecto, Firefox proporciona operaciones de descarga y extracción casi perfectas con las descargas. Para paquetes RPM, Firefox dará la opción de instalar automáticamente RPM con paquetes de instalación de sistema. En el caso de archivos comprimidos como **.tar.bz**, Firefox invocará automáticamente File Roller, permitiéndole descomprimir y extraer inmediatamente los archivos de código fuente en un directorio seleccionado.

Reproductores de video y DVD

El acceso a reproductores DVD y multimedia se proporciona en **dvd.sourceforge.net**. Aquí encontrará vínculos con reproductores como VideoLan, MPlayer y Xine.

Proyectos y reproductores	Descripciones y sitios
LinuxTV.org	Vínculos a sitios de video, TV y DVD: linuxtv.org
Lista de reproductores de DVD	dvd.sourceforge.net
xine	Reproductor de video Xine: xinehq.de
Totem	Reproductor de video y DVD Totem para GNOME, está basado en Xine y utiliza GStreamer: xinehq.de
VideoLAN	Flujo multimedia de red, incluye soporte de alta definición x264: videolan.org
MPlayer	Reproductor de DVD y multimedia MPlayer: mplayerhq.hu
PowerDVD	Cyberlink PowerDVD para Linux gocyberlink.com
DVD::rip	Conversión DVD y software DivX: exit1.org/dvdrip
kdetv	Visor de TV de KDE
tvtime	Visor de TV: tvtime.sourceforge.net
DivX para Linux	labs.divx.com/DivXLinuxCodec
XviD	DivX de fuente abierta, tal vez se incluya en ciertas distribuciones: xvid.org

TABLA 12-4 Proyectos y aplicaciones de video y DVD

- El proyecto VideoLan (videolan.org) ofrece soporte para flujo de red en casi todos los formatos multimedia, incluidos MPEG-4 y MPEG-2. Incluye un reproductor multimedia, VLC, que puede funcionar con cualquier tipo de sistema.
- Mplayer es uno de los reproductores más populares y con mayores opciones para multimedia y DVD en uso. Es una opción de fuente abierta y plataforma cruzada para RealPlayer y Windows Media, e incluye soporte para DivX. Puede descargar MPlayer de mplayerhq.hu. Mplayer utiliza un conjunto extenso de bibliotecas de soporte y aplicaciones como **lirc**, **lame**, **lzo** y **aalib**, también en el sitio. Si tiene problemas para desplegar video, asegúrese de revisar las preferencias para diferentes dispositivos de video y seleccione la que funcione mejor.
- Xine es un reproductor de video multipropósito para sistemas Linux y Unix que puede reproducir video, DVD y discos de audio. Consulte xinehq.de para conocer más información.
- Totem es un reproductor de películas de GNOME basado en Xine que usa GStreamer. Para expandir las capacidades de Totem, necesita instalar plug-ins GStreamer complementarios, como el plug-in de DivX para desplegar archivos DivX.
- Para la transcodificación, conversión de DVD y soporte a DivX, revise el proyecto DVD::rip (exit1.org/dvdrip).
- VideoLAN es otro reproductor popular que requiere de una lista de paquetes de soporte.
- El soporte adicional a codecs lo proporcionan ffmpeg y x264. El codec x264 es una versión de fuente abierta del codec para alta definición H.264, desarrollado por VideoLAN.

Ningún software de fuente abierta almacenado en SourceForge.net descifra el candado CSS de los DVD comerciales. Sin embargo, descargue e instale la biblioteca **libdvdcss**, que compensa la decodificación de CSS tratando el DVD como un dispositivo de bloque, permitiéndole utilizar cualquiera de los reproductores de DVD para ejecutar DVD comerciales. También proporciona acceso libre de región. Tenga en cuenta que tal vez esto no sea legal en países requiriendo licencias CSS para reproductores de DVD.

Originalmente, muchos de estos reproductores no soportaban menús de DVD. Con la biblioteca **libdvdnav**, estos reproductores ahora presentan soporte total a menús de DVD. La biblioteca **libdvdread** ofrece soporte básico a la interfaz de DVD, como lectura de archivos IFO.

Reproductores de TV

El sitio linuxtv.org proporciona vínculos detallados a DVD, transmisión de video digital (DVB, Digital Video Broadcasting) y multidifusión. El sitio también provee descargas de muchas aplicaciones de video para Linux.

tvtime es un reproductor de TV trabajando con muchas tarjetas de captura de video comunes, dependiente de controladores desarrollados por chips de sintonizador de TV en esas tarjetas, como los chips Conexant. Sólo puede desplegar una señal de TV. No tiene capacidades de grabación o reproducción de archivos. Consulte tvtime.sourceforge.net para conocer información más detallada.

En el caso de KDE, hay varias aplicaciones de video en desarrollo, incluidos kdetv. Consulte kde-apps.org para descargas. Para reproductores GNOME, consulte gnomefiles.org.

Soporte a DVB y HDTV

Para recepción de DVB y HDTV, puede usar casi todas las tarjetas DVB, además de muchas tarjetas de HDTV, como la tarjeta de video PCHDTV (pdhdtv.org). Por ejemplo, las tarjetas PCHDTV más recientes manejan controladores **cx88-dvb**, incluido en el kernel de Linux más reciente (para

versiones de kernel más antiguas debe descargar, compilar e instalar un controlador separado). Es probable que el controlador de kernel DVB no esté instalado como opción predeterminada. En tal caso tendrá que utilizar `modprobe` para instalarlo de la manera manual (en Debian, coloque el nombre del módulo en `/etc/modules` para que se descargue automáticamente). Utilice el comando `lsmod` para saber si su módulo DVB está cargado.

Muchas aplicaciones con opciones de DVB, como Kaffeine, ya tienen accesibilidad DVB instalada. También puede utilizar las herramientas `dvb-tools` para administrar el acceso; entre éstas se incluyen **scan** para explorar sus canales y las herramientas **zap** para acceder directamente a la señal. Primero necesitará crear un archivo de configuración de canales para utilizar la herramienta de exploración. Para la tarjeta PC-HDTV, puede utilizar `dvb-atsc-tools`, que se descarga e instala desde el sitio Web www.pchdtv.com. Las herramientas son de código fuente y puede usar los comandos `make` y `make install` simples para crearlas e instalarlas (asegúrese de que está instalado el soporte para cabeceras del kernel y software de desarrollo). Estas herramientas podrían funcionar con cualquier tarjeta HDTV compatible con Conexant, como Fusion HDTV.

Las herramientas DVB también se emplean para grabar HDTV y transmitir archivos DVB a TS (flujo de transporte).

Luego, el archivo **transport stream (.ts o .tp)** puede verse con un visor HDTV, como la versión HDTV de Xine o el reproductor de medios VLZ de Videolan. Puede emplear MythTV o Xine para ver y grabar. Consulte el sitio de MythTV para conocer más detalles (mythtv.org). Asegúrese de que los decodificadores apropiados están instalados, como `mpeg2`, `FFmpeg` y `A52 (ac3)`. Para transmisión DVB, muchos reproductores DVB y herramientas como Kaffeine y Klear, además de `vdr`, sintonizarán y grabarán transmisiones DVB en formatos `t`, `s` y `c`. El paquete `dvb-tools` almacena configuraciones de ejemplo.

DivX y Xvid en Linux

DivX es una tecnología de compresión de video comercial (gratis para uso personal) para proporcionar video con calidad DVD en tamaños de archivo relativamente pequeños. Puede comprimir 60 minutos de video DVD en cerca de 400MB, manteniendo muy buena calidad. DivX está basado en el formato de compresión MPEG-4, mientras para un DVD es MPEG-2. Puede descargar la versión para Linux de DivX gratis de labs.divx.com/DivXLinuxCodec. Tendrá que instalar manualmente el paquete. Si lo descarga con Firefox, puede seleccionar que se extraiga el archivo directamente.

En lugar de tratar de hacer que DivX trabaje, puede usar la versión de fuente abierta de DivX conocida como Xvid. Casi todos los archivos DivX pueden ejecutarse con XviD. XviD es un proyecto de fuente abierta totalmente independiente, pero compatible con archivos DivX. Casi todas las distribuciones proporcionan paquetes de software que pueden instalarse fácilmente para Xvid. También es posible descargar el código fuente de XviD desde xvid.org.

Clientes de correo y noticias

Su sistema Linux soporta un amplio rango de clientes de correo electrónico y noticias. Los clientes de correo posibilitan enviar mensajes a otros usuarios de su sistema o accesibles desde su red, así como recibirlos de ellos. Los clientes de noticias permiten leer artículos y mensajes publicados en grupos de noticias, con acceso abierto para todos los usuarios. En este capítulo se revisan clientes de correo y noticias instalados en Linux.

Clientes de correo

Puede enviar y recibir correos electrónicos en diversas formas, dependiendo del tipo de cliente de correo que utilice. Aunque todas las utilerías de correo electrónico realizan las mismas tareas básicas de recepción y envío de mensajes, suelen tener diferentes interfaces. Algunos clientes de correo operan en un escritorio, como KDE o GNOME. Otros se ejecutan en cualquier administrador de X Window System. Varios clientes de correo populares fueron diseñados para emplear una interfaz basada en pantalla y pueden iniciarse sólo desde la línea de comandos. Otros clientes de correo tradicionales se desarrollaron sólo para la interfaz de línea de comandos, la que requiere escriba sus comandos en una sola línea de comandos. Casi todos los clientes de correo descritos aquí se incluyen en distribuciones de Linux y vienen en paquetes estándar para una instalación sencilla. En el caso de servicios de correo de Internet basados en Web, como Hotmail, Google y Yahoo, utilice un explorador Web en vez de un cliente de correo para acceder a sus cuentas de correo proporcionadas por esos servicios. En la tabla 13-1 se muestra una lista de varios clientes de correo populares de Linux. El correo se transporta a los destinos y, desde éstos, al utilizar agentes de transporte de correo. Sendmail, Exim y Smail envían y reciben correo desde diferentes destinos en Internet u otros sitios de una red; para envío de correo por Internet, manejan un protocolo simple de transporte de correo (SMTP, Simple Mail Transport Protocol). Casi todas las distribuciones de Linux se instalan de manera automática y configuran localmente Sendmail (enviar correo). Al iniciar su sistema, una vez configuradas sus conexiones de red, puede enviar y recibir mensajes a través de Internet.

Puede firmar su mensaje de correo electrónico con la misma información de firma estándar, como su nombre, dirección o direcciones de Internet o frase de despedida. Es útil agregar automáticamente su información a mensajes. Para ello, necesita crear un archivo de firma en su directorio home e insertar su información de firma en éste. Un *archivo de firma* es un archivo de texto estándar que puede modificar con cualquier editor de texto. Los clientes de correo como KMail permiten especificar un archivo para funcionar como archivo de firma. Otros, como Mail, esperan que el archivo de firma se llame **.signature**.

Cliente de correo	Descripción
Kontakt (KMail, KAddressbook, KOrganizer)	Incluye el cliente de correo K de Desktop, KMail; correo integrado, libreta de direcciones y agenda
Evolution	Cliente de correo electrónico
Balsa	Cliente de correo de GNOME
Thunderbird	Cliente de correo y lector de noticias independientes del grupo Mozilla
Netscape	Cliente de correo basado en explorador Web
GNUEmacs and XEmacs	Clientes de correo Emacs
Mutt	Cliente de correo basado en pantalla
Sylpheed	Cliente de correo y noticias GTK
Mail	Cliente de correo de línea de comandos original basado en Unix
SquirrelMail	Cliente de correo basado en Web

TABLA 13-1 Clientes de correo de Linux

MIME

Las extensiones de correo de Internet para propósitos múltiples (*MIME*, Multipurpose Internet Mail Extensions) se usan para permitir a los clientes de correo enviar y recibir archivos multimedia y otros que manipulan diferentes conjuntos de caracteres como los empleados para otros idiomas. Los archivos multimedia pueden ser imágenes, archivos de sonido e incluso video. Los clientes de correo soportando MIME utilizan archivos binarios automáticamente como datos adjuntos en los mensajes. Los clientes de correo instrumentados con MIME mantienen un archivo llamado **mailcap**, para correlacionar diferentes tipos de mensajes MIME con aplicaciones que su sistema pueda ver o desplegar. Por ejemplo, un archivo de imagen se asigna a una aplicación capaz de desplegar imágenes. Entonces, sus clientes de correo pueden ejecutar dicho programa para mostrar una imagen dentro del mensaje. Un archivo de sonido se asigna a una aplicación para reproducir audio en sus bocinas. Casi todos los clientes de correo disponen de MIME y utilizan sus propias versiones del archivo **mailcap**. Otros usan un programa denominado *metamail* para añadir soporte a MIME. MIME no sólo se utiliza en clientes de correo; los administradores de archivos de GNOME y KDE recurren a MIME para correlacionar un archivo con una aplicación en particular, para de ese modo lanzar la aplicación directamente desde el archivo.

El archivo `mime.types`

Las aplicaciones se asocian con archivos binarios por medio de **mailcap** y **mime.types**. El archivo **mime.types** define diferentes tipos de MIME, al asociar un archivo MIME con cierta aplicación. Luego, el archivo **mailcap** asocia de vuelta cada tipo MIME con una aplicación específica. El sistema mantiene su propio archivo de tipos MIME, generalmente `/etc/mime.types`.

Las entradas en el archivo de tipos MIME asocia un tipo y posibles subtipos MIME de una aplicación con un conjunto de posibles extensiones de archivo utilizadas para archivos ejecutándose con un tipo de aplicación dado. El tipo MIME suele calificarse con mayor detalle por un subtipo, con una diagonal separando al tipo principal del subtipo. Por ejemplo, un tipo de imagen MIME tiene varios subtipos: JPEG, GIF o TIFF. Aquí se muestra un ejemplo de una entrada definiendo un tipo MIME para archivos JPEG. El tipo MIME es **image/jpeg**, y la lista de posibles extensiones de archivo es **"jpeg jpg jpe"**.

```
image/jpeg jpeg jpg jpe
```

Las aplicaciones especificadas dependerán de las disponibles en su sistema específico. El tipo MIME está separado de su aplicación con punto y coma. En muchos casos, establecen programas basados en X Window System. Los comentarios se indican con #. Un * utilizado en un subtipo MIME hace referencia a todos los subtipos. La entrada **image/*** se utilizaría para una aplicación que puede ejecutar todos los tipos de archivos de imagen. Un código de formación, %s, se utiliza para referir los archivos adjuntos a ejecutarse en esta aplicación. Aquí se muestran ejemplos de entradas **mailcap**. La primera entrada asocia todos los archivos **image** con el visor de imagen xv. Los dos siguientes asocian archivos de video en general y video MPEG con la aplicación XAnim.

```
image/*; xv %s
video/*; xanim %s
video/mpeg; xanim %s
```

Asociaciones MIME en GNOME y KDE

También es posible crear y editar tipos MIME en los escritorios de GNOME y KDE. En el caso de GNOME, dirijase a la capplet de tipos MIME en el Centro de control GNOME. Esta capplet mostrará una lista de tipos MIME definidos por su sistema, junto con extensiones asociadas a nombres de archivo. Edite una entrada para cambiar aplicación e icono asociados con el tipo MIME, el tipo de archivo. En KDE, use la entrada Asociaciones de archivo, del Centro de control de KDE, bajo Componentes de KDE. Éste mostrará una lista de tipos MIME y extensiones de nombre de archivo asociadas. Seleccione una entrada para editarla y cambiar la aplicación asociada con ésta. KDE g la información de tipo MIME en un archivo separado llamado **mimelnk**, en el directorio de configuración KDE.

Asociaciones estándar MIME

Aunque puede crear sus propios tipos MIME, un conjunto estándar ya está en uso. Los tipos texto, imagen, audio, video, aplicación, multiparte y mensaje, además de sus subtipos, ya se encuentran definidos en el sistema. Encontrará que extensiones de archivo de uso común como **.tif** y **.jpg**, para archivos de imagen TIFF y JPEG, ya están asociadas con una aplicación y tipo MIME. Aunque puede cambiar de manera sencilla la aplicación asociada, es mejor mantener los tipos MIME ya instalados. Los tipos MIME oficiales actuales se muestran en una lista en el sitio Web de IANA (**iana.org**) bajo el nombre Media Types, provistos como parte de los Assignment Services. Puede acceder a los archivos de tipos de medios directamente en su sitio.

Protocolos de codificación y autenticación OpenPGP/MIME y S/MIME

S/MIME y OpenPGP/MIME son protocolos de autenticación para iniciar sesión y codificar mensajes de correo. S/MIME fue desarrollado originalmente por RSA Data Security. OpenPGP es un estándar abierto basado en el protocolo PGP/MIME desarrollado por el grupo PGP (Pretty Good Privacy, muy buena privacidad). Los clientes KMail y Evolution pueden utilizar OpenPGP/MIME para autenticar mensajes. Consulte Internet Mail Consortium (Consortio de correo de Internet) para adquirir más información, **imac.org**.

Evolution

Evolution es el principal cliente de correo para el escritorio GNOME. Se instala como opción predeterminada junto con OpenOffice. Aunque está diseñado para GNOME, también funcionará en KDE. Evolution es un cliente de correo, calendario y libreta de direcciones integrado, actualmente desarrollado por Novell y ahora conocido como Novell Evolution. El correo Evolution es una herramienta poderosa con soporte para muchos protocolos (SMTP, POP e IMAP), varias cuentas de correo y codificación. Con Evolution, puede crear más de una cuenta de correo en diferentes

servidores, incluidos los que manejan diferentes protocolos, como POP o IMAP. También puede descifrar mensajes cifrados con PGP o GPG.

El correo Evolution brinda una GUI sencilla, con barra de herramientas para comandos de uso común y barra lateral para métodos abreviados. Un menú de comandos de Evolution le permite acceder a otras operaciones. El panel principal se divide en dos, uno para mostrar una lista de encabezados de correo y otro para desplegar el mensaje seleccionado. Haga clic en cualquier título de encabezado para ordenarlos por categoría. Evolution también permite carpetas virtuales. Se trata de carpetas creadas por el usuario para almacenar correos cumpliendo ciertos criterios. El correo entrante se puede distribuir automáticamente a una carpeta virtual particular. Para notificación automática de correo, use el plug-in de notificación de correo para Evolution.

Thunderbird

Thunderbird es un cliente de correo independiente lleno de opciones, ofrecido por el proyecto Mozilla (mozilla.org). Está diseñado para ser de fácil uso, incluyendo numerosas opciones de personalización y realmente seguro. Presenta un filtro avanzado e inteligente para correo basura, además de características de seguridad como cifrado, firmas digitales y S/MIME. Para proteger contra virus, es posible examinar los archivos adjuntos al correo sin ejecutarlos. Thunderbird habilita IMAP y POP, además del uso de libretas de direcciones LDAP. Funciona como un lector de noticias y presenta un lector RSS integrado. Además, Thunderbird es una aplicación expansible, que permite agregar módulos personalizados para mejorar sus funciones. Entre las extensiones que puede descargar se encuentran búsqueda de diccionario y barras laterales para agenda de contactos, desde el sitio Web del cliente. La codificación GPG puede soportarse con la extensión **enigmail**.

La interfaz maneja un formato de tres paneles estándar; en un panel lateral muestra listas de cuentas de correo y sus cuadros. El panel superior presenta una lista con las entradas principales y el panel inferior texto. Los comandos pueden ejecutarse a través de la barra de herramientas, menús o comandos rápidos de teclado. Incluso puede cambiar la apariencia usando diferentes temas. Thunderbird también soporta correo en formato HTML, para desplegar componentes Web como URL en mensajes de correo.

El panel lista de mensajes mostrará varios campos para ordenar sus mensajes. Algunos usan símbolos, como los iconos Threads (Mensajes agrupados por conversaciones), Adjuntos o Leídos. Al hacer clic en Conversaciones se recolectarán los mensajes en sus respectivas conversaciones, con las respuestas agrupadas juntas. El último icono en los campos de lista del mensaje es un menú emergente que permite seleccionar cuáles archivos desplegar. Thunderbird proporciona varios filtros de despliegue personalizables, como Personas que conozco, para sólo presentar mensajes de quienes están en su libreta de direcciones y Adjuntos, que extiende mensajes con archivos adjuntos. Incluso puede crear filtros propios de despliegue. Las capacidades de búsqueda y ordenamiento también incluyen filtros de coincidencia con patrones seleccionados en un campo, incluidos asunto, fecha o cuerpo del mensaje.

Cuando inicia por primera vez Thunderbird, se le pedirá crear una cuenta de correo. Agregue más cuentas de correo o modifique su cuenta actual seleccionando Configuración de las cuentas, del menú Editar. Después haga clic en Añadir cuenta, para abrir un cuadro de diálogo con cuatro opciones, una de las cuales es una cuenta de correo electrónico. Después de seleccionar la opción Cuenta de correo electrónico, se le pide inserte su dirección de correo electrónico y nombre. En el siguiente panel debe especificar el protocolo POP o IMAP e ingresar el nombre del servidor de correo electrónico entrante, como **smtp.miservidordecorreoelectronico.com**. Después especifique el nombre de usuario de su servicio de correo electrónico. Luego, inserte una etiqueta de nombre de entrada para identificar la cuenta en Thunderbird. Una pantalla de verificación final le confirma sus entradas. En la ventana Configuración de cuenta, verá una entrada para su servidor de noticias, con paneles

para Configuración del servidor, Copias y carpetas, Redacción y direcciones, Sin conexión y espacio en disco, Acuses de recibo y Seguridad. El panel Configuración del servidor tiene entradas para su configuración de nombre, puerto, nombre de usuario y conexión, además tareas de servidor, tal y como descargar automáticamente nuevos mensajes. El panel Seguridad abre el Administrador de certificado, donde selecciona certificados de seguridad para firmar o codificar mensajes digitalmente.

Thunderbird cuenta con una libreta de direcciones dónde introducir información de contacto completa, incluidos direcciones de correo electrónico, domicilios, números telefónicos y notas. Seleccione Libreta de direcciones desde el menú Herramientas, para abrir la ventana Libreta de direcciones. Existen tres paneles, uno para las libretas de direcciones disponibles, otro para mostrar una lista de entradas de direcciones con campos de entradas como nombre, correo electrónico y organización, y uno más para desplegar información de dirección. Puede ordenar las entradas por estos campos. Al hacer clic en una entrada se desplegará la información de dirección, incluidas de correo electrónico, postal y teléfono. Sólo se despliegan campos con valores. Para crear una nueva entrada en una libreta de direcciones, haga clic en Nueva tarjeta, para extender una ventana con paneles señalando Contacto y Dirección. A fin de Crear listas de correo desde entradas de libreta de direcciones, haga clic en el botón Nueva lista, especifique el nombre de una e inserte las direcciones de correo electrónico.

Una vez tenga su libreta de direcciones configurada, podrá usar sus direcciones cuando escriba mensajes de correo de manera sencilla. En la ventana Redactar, haga clic en el botón Contactos, para abrir el panel Contactos. Las entradas de su libreta de direcciones se mostrarán en una lista, partiendo del nombre del contacto. Sólo haga clic en el nombre para agregarlo al cuadro de direcciones de su mensaje de correo electrónico. Como opción, puede abrir la libreta de direcciones, arrastrar y colocar direcciones en un cuadro de direcciones en su ventana de mensaje.

Los mensajes de correo electrónico, direcciones e información de configuración del usuario se mantienen en archivos localizados en el directorio **.thunderbird**, en el directorio home del usuario. Para respaldar esta información basta crear una copia de ese directorio. Los mensajes para diferentes cuadros de mensaje se almacenan en el subdirectorio **Mail**. Si está migrando a un nuevo sistema, simplemente copie el directorio del sistema antiguo. Para respaldar el correo de cualquier cuenta de correo dada, copie el subdirectorio **Mail** de esa cuenta. Aunque las libretas de direcciones predeterminadas, **abook.mab** e **history.mab**, pueden copiarse de manera intercambiable, las libretas de direcciones no predeterminadas necesitan exportarse a un formato LDIF y después importarse a una nueva aplicación Thunderbird. Es recomendable exportar regularmente sus libretas de direcciones a archivos LDIF como copias de seguridad.

Clientes de correo de GNOME: Evolution, Balsa y otros

Hay varios clientes de correo basados en GNOME disponibles (véase la tabla 13-2). Entre éstos se incluyen Evolution, Balsa y Sylpheed (Evolution se incluye con GNOME). Consulte gnomefiles.org

Aplicación	Descripción
Balsa	Cliente de correo electrónico para GNOME que soporta POP3, IMAP, carpetas locales y multiprocesamiento
Evolution	Cliente de correo, calendario y administrador de contactos integrados
Sylpheed	Cliente de correo y noticias similar a clientes de Windows
gnubiff	Herramienta de revisión y notificación de correo electrónico
Mail Notification	Revisor y notificador de correo electrónico; opera con diferentes clientes de correo, incluidos MH, Sylpheed, Gmail, Evolution y Mail

TABLA 13-2 Clientes de correo de GNOME

para conocer más información acerca de clientes de correo, conforme aparezcan. Muchos se basan en bibliotecas para cliente de correo de GNOME (camel), que ofrece soporte a operaciones de correo estándar.

Balsa implementa una GUI llena de características para redactar, enviar y recibir mensajes de correo. La ventana Balsa despliega tres paneles para carpetas, encabezados y mensajes. El panel de la izquierda despliega carpetas de correo. Inicialmente tiene tres carpetas: bandeja de entrada para correos recibidos, bandeja de salida para correos redactados pero sin enviar y una carpeta de papelería para mensajes eliminados. También puede crear carpetas propias donde almacenar mensajes particulares. Para colocar un mensaje en una carpeta creada, haga clic y arrastre el encabezado del mensaje a la carpeta.

El cliente de correo de K Desktop: KMail

El cliente de correo de K Desktop incluye una GUI con características completas para redactar, enviar y recibir mensajes de correo. KMail ahora es parte del conjunto KDE Personal Information Management, KDE-PIM, que incorpora el conjunto libreta de direcciones (KAddressbook), organizador y programador (KOrganizer), así como escritor de notas (KNotes). Todos estos componentes también se encuentran directamente integrados en el escritorio en Kontact. Para iniciar KMail, inicie la aplicación Kontact. La ventana de KMail despliega tres paneles para carpetas, encabezados y mensajes. El panel de la sección superior izquierda muestra carpetas de correo. Tiene un folder de entrada para correo recibido, otro de salida para correo redactado pero sin enviar y otra más de correo enviado para mensajes ya entregados. Si así lo desea, cree carpetas propias de correo y guarde los mensajes seleccionados en éstas. El panel de la sección superior derecha despliega encabezados de correo de la carpeta seleccionada. Para extender un mensaje, haga clic en su encabezado. El mensaje entonces se despliega en el gran panel bajo la lista de encabezados. También puede enviar y recibir archivos adjuntos, incluidos archivos binarios. Imágenes y películas recibidas se despliegan mediante la utilería de K Desktop apropiada. Si hace clic con el botón derecho en el mensaje, un menú desplegable indicará opciones para las acciones a tomar. Mueva o copie un mensaje a otra carpeta o simplemente bórralo. También, responda o reenvíe el mensaje. Se accede a KMail, además de Kontact, KOrganizer y KAddressbook, desde los menús Escritorio, Oficina e Internet de KDE.

Con el fin de configurar KMail para ser usado en sus cuentas de correo, debe ingresar la información de cuenta. Seleccione la entrada Configurar KMail, en el menú Preferencias. Hay disponible varios paneles desde la ventana Preferencias, desplegables al hacer clic en los iconos de la columna izquierda. Para cuentas, seleccione el panel Cuentas. Quizás tenga más de una cuenta en servidores de correo mantenidas por su ISP o LAN. Se despliega una ventana de configuración dónde ingresar información de inicio de sesión, contraseña y host. Para un acceso seguro, ahora KMail soporta SSL, siempre y cuando OpenSSL esté instalado. Los usuarios pueden ahora cifrar y descifrar mensajes. También soporta IMAP, además de los protocolos POP y SMTP.

Cliente de correo Web SquirrelMail

Puede utilizar la herramienta de correo Web SquirrelMail para acceder correo desde un sistema Linux con su explorador Web. Desplegará una pantalla de inicio de sesión para los usuarios de correo. Presenta una lista de bandeja de entrada y un lector de mensajes; soporta edición y envío de nuevos mensajes, además de una estructura de plug-in para agregar nuevas características. Encontrará más información acerca de SquirrelMail en squirrelmail.org. El archivo de configuración de Apache es `/etc/httpd/conf.d/squirrelmail.conf` y SquirrelMail está instalado en `/usr/share/squirrelmail`. Asegúrese de que el servidor IMAP también está instalado.

Para configurar SquirrelMail, siga la secuencia de comandos `config.pl` en el directorio `/usr/share/squirrelmail/config`. Esto despliega un menú de texto simple dónde configurar opciones

como qué servidor usar, opciones predeterminadas de carpeta, opciones generales y preferencias de organización.

```
./config.pl
```

Para acceder a SquirrelMail, emplee la dirección del servidor Web con la extensión **/squirrelmail**, como en **localhost/squirrelmail** para usuarios del sistema local o **mytrek.com/squirrelmail** para usuarios remotos.

Emacs

Los clientes de correo Emacs están integrados en el entorno Emacs, del que el editor Emacs es la aplicación primaria. Sin embargo, son clientes de correo totalmente funcionales. La versión GNU de Emacs incluye un cliente de correo así como otros componentes, un lector de noticias y editor, entre otros. GNU Emacs se incluye en distribuciones de Linux. Revise el sitio Web de **gnu.org/software/emacs** para conocer más información. Cuando inicia GNU Emacs, los botones de menú se despliegan a lo largo de la parte superior de la pantalla. Si ejecuta Emacs en un entorno X Window System, tiene capacidades GUI completas y puede seleccionar menús con su ratón. Para acceder al cliente de correo Emacs, selecciónelo desde las entradas de correo del menú Herramientas. Para redactar y enviar mensajes, seleccione Enviar correo, en el menú Herramientas. Se abre una pantalla con peticiones de encabezados Para y Asunto. Entonces escriba el mensaje en el recuadro inferior, empleando cualquiera de las capacidades de edición de Emacs. GNU Emacs es un entorno de trabajo dentro del cual se realizan varias tareas (cada una con su propio búfer). Cuando lee correo, se abre un búfer para almacenar la lista de encabezados, mientras la lectura de un mensaje, otro búfer maneja los abiertos para almacenar el contenido. Cuando crea un mensaje, un búfer más almacena el texto escrito. Los búferes abiertos para correo, noticias o notas de edición o archivos, se muestran en una lista en el menú Buffers. Puede utilizar este menú para cambiar entre éstos.

XEmacs es otra versión de Emacs, diseñada para operar sólo con una GUI. Las aplicaciones de Internet, a las que accede de manera sencilla desde la barra de botones principal de XEmacs, incluyen explorador Web, utilería de correo y lector de noticias. Cuando crea un mensaje, tiene el uso completo del editor Emacs con todas sus características, incluidos revisor ortográfico, opciones de búsqueda y reemplazo.

Clientes de correo de línea de comandos

Varios clientes de correo emplean una interfaz de línea de comandos simple. Pueden ejecutarse sin soporte, como X Window System, escritorios ni soporte de cursor. Son simples y fáciles de usar pero incluyen un extenso conjunto de características y opciones. Dos de los clientes de correo más utilizados de este tipo son Mail y Mutt. Mail es el cliente de correo mailx desarrollado para el sistema Unix. Se considera un tipo de cliente de correo predeterminado que puede encontrarse en todos los sistemas Unix y Linux. Mutt es un cliente basado en cursor capaz de ejecutarse desde la línea de comandos.

NOTA También puede usar el cliente de correo Emacs desde la línea de comandos, como se describió en la sección anterior.

Mutt

Mutt tiene una interfaz basada en pantalla, fácil de usar y un conjunto extenso de características, como soporte a MIME. Encontrará más información acerca de Mutt en el sitio Web de Mutt, **mutt.org**. Allí puede descargar versiones recientes de Mutt, acceder a manuales en línea y recursos de

ayuda. En casi todas las distribuciones, el manual de Mutt está localizado en el directorio `/usr/doc`, bajo Mutt. El grupo de noticias de Mutt es **comp.mail.mutt**, donde puede publicar consultas y revisar desarrollos recientes de Mutt.

Mail

Lo que ahora se conoce como la utilidad Mail fue creada originalmente para BSD Unix y sólo se llama mail. Versiones posteriores del sistema Unix System V adoptaron la utilidad de correo BSD y la renombraron mailx; ahora se conoce como Mail. Las funciones de Mail son un cliente de correo predeterminados de facto en sistemas Linux y Unix. Todos los sistemas tienen el cliente de correo Mail, en el caso de que no haya otros clientes de correo. Consulte la página Man de **mail** para adquirir información más detallada y comandos.

Para enviar un mensaje con Mail, escriba **mail** junto a la dirección del destinatario del mensaje. Tras oprimir ENTER se le pedirá un asunto. Ingrese el asunto del mensaje y oprima ENTER de nuevo. En este punto, se le coloca en el modo de inserción. Cualquier cosa escrita se toma como contenido del mensaje. Oprima ENTER para agregar una nueva línea al texto. Cuando termine de escribir su mensaje, oprima CTRL-D en una línea para terminar el mensaje. Entonces se pedirá que inserte un usuario a quien enviará una copia del mensaje (Cc). Si no quiere una copia, oprima ENTER. Entonces verá EOT (*End of Transmission, final de la transmisión*) desplegado tras pulsar oprima CTRL-D para terminar su mensaje.

Para enviar un mensaje a más de un usuario simultáneamente, elabore una lista de direcciones de usuarios como argumentos en la línea de comandos tras el comando **mail**. En el siguiente ejemplo, el usuario enviará el mismo mensaje a **carlos** y **alicia**.

```
$ mail carlos alicia
```

Para recibir un correo, introduzca sólo el comando **mail** y oprima ENTER. Esto invocará una shell de Mail con su propio indicador de comandos y comandos de correo. Se despliega una lista de encabezados de mensaje. La información de encabezados se ordena en archivos, comenzando por el estado del mensaje y su número. El estado de un mensaje se indica con una sola letra mayúscula, usualmente **N** para *nuevo* o **U** para *no leído*. Un número de mensaje, usado para referir de manera sencilla sus mensajes, sigue al campo estado. El siguiente campo es dirección del remitente, seguido por fecha y hora en que se recibió el mensaje, después el número de líneas y caracteres del mensaje. El último campo contiene asunto dado por el remitente al mensaje. Después de los encabezados, la shell de Mail despliega su indicador de comandos, un signo **&**. En el indicador de comandos de Mail, puede insertar comandos que operan en el mensaje. Aquí se muestra un ejemplo de encabezado de Mail e indicador de comandos:

```
$ mail
Mail version 8.1 6/6/93. Type ? for help.
"/var/spool/mail/larisa": 3 messages 2 unread
 1 carlos@tortuga.mipista. Thu Jun 7 14:17 22/554 "viaje"
>U 2 alicia@tortuga.mipista Thu Jun 7 14:18 22/525 "fiesta"
>U 3 daniel@tortuga.mipista Thu Jun 7 14:18 22/528 "boletín informativo"
& q
```

Mail refiere mensajes a través de una lista o el marcador de mensaje actual (>). El signo de mayor que (>) se coloca antes de un mensaje que considerado el actual. Se hace referencia al mensaje actual cuando no se incluye número de mensaje con un comando de Mail. También puede hacer referencia a mensajes, usando una lista de mensajes que consta de varios números de mensaje. Con los mensajes dados en el ejemplo anterior, puede hacer referencia a tres mensajes con **1-3**.

Los comandos **R** y **r** se usan para responder a un mensaje recibido. El comando **R**, insertado con un número de mensaje, genera un encabezado para enviar un mensaje que después cambia a estado de inserción para escribir el mensaje. El comando **q** sale de Mail. Cuando sale, los mensajes ya leídos se colocarán en un archivo llamado **mbox** en su directorio home. En vez de guardar mensajes en el archivo **mbox**, puede usar el comando **s** para guardar un mensaje explícitamente en el archivo elegido. Mail tiene su propio archivo de inicialización, llamado **.mailrc**, en ejecución cada vez que Mail se invoca, ya sea para enviar o recibir mensajes. Dentro de éste, defina las opciones de Mail y cree alias de Mail. Configure opciones que agregan características diferentes a Mail, como cambiar el indicador de comandos o guardar copias de mensajes enviados. Para definir un alias, inserte la palabra clave **alias**, seguida por el alias seleccionado y después la lista de direcciones representadas. En el siguiente ejemplo, el alias **miclase** se define en el archivo **.mailrc**.

```
.mailrc
alias miclase carlos daniel alicia julia clarisa
```

En el siguiente ejemplo, el contenido del archivo **tarea** se envía a todos los usuarios, cuyas direcciones están en el alias **miclase**.

```
$ mail miclase < tarea
```

Notificaciones de correo recibido

Conforme reciba mensajes de correo, se colocarán automáticamente en su archivo de bandeja de entrada, pero no se le notifica cuando recibe un mensaje. Utilice un cliente de correo para recuperar cualquier mensaje nuevo, o use una herramienta para monitorear correo, indicándole si tiene correo en espera. Hay varias herramientas de notificación de correo disponibles, como gnumbiff y Mail Notification. Mail Notification soportará Gmail, al igual que Evolution (para Evolution, instale el paquete de plug-in separado). Cuando inicia sesión por primera vez después de haber instalado Mail Notification, se despliega la ventana de configuración Mail Notification. Aquí se agregan nuevas cuentas de correo para revisar, como Gmail, además de establecer otras características, por ejemplo resúmenes desplegables. Cuando recibe un correo, un icono de Mail aparecerá en la applet de notificación de su panel. Mueva el cursor sobre el applet para revisar si hay algún mensaje nuevo. Al hacer clic en el applet se desplegará la ventana de configuración de Mail Notification, aunque puede configurar esto directamente en su aplicación de correo electrónico. gnumbiff notificará si cualquier correo POP3 o IMAP se recibe.

El escritorio de KDE tiene una utilería para monitoreo de correo llamada Korn, trabaja casi de la misma forma. Korn muestra una bandeja de entrada vacía cuando no hay mensajes y una con cartas inclinadas al recibir correo. Si existe correo antiguo en su bandeja de entrada, las cartas se desplegarán en un cuadro limpio. Configure estos iconos con cualquier imagen que desee. También especifique el cliente de correo que se utilizará y el intervalo de consultas para revisar el correo nuevo. Si tiene varias cuentas de correo, configure un perfil de Korn para cada una. Aparecen diferentes iconos para cada cuenta, indicándole cuando se recibe un correo en una de éstas.

En el caso de interfaces de línea de comandos, utilice la utilería biff. Ésta notifica de inmediato cuando se recibe un mensaje. Esto es útil cuando espera un mensaje y quiere saberlo de inmediato tras recibirlo. Para activar biff, escriba **biff y** en la línea de comandos. Para desactivar biff, inserte **biff n**. Para saber si biff está activado, inserte **biff** solo.

Bloquee biff temporalmente con el comando **mesg n** para evitar que cualquier mensaje se despliegue en su pantalla. El comando **mesg n** no sólo detiene cualquier mensaje de Write and Talk, también biff y los mensajes Notify. Después, puede desbloquear biff con un comando **mesg y**. Un comando **mesg n** es muy útil si no quiere ser molestado mientras trabaja en un proyecto.

Acceso a Mail desde servidores de correo POP remotos

La mayoría de clientes de correo más recientes están equipados para acceder a cuentas de correo en servidores remotos. Para tales clientes de correo, puede especificar una cuenta de correo separada con su propia bandeja de entrada. Por ejemplo, si utiliza un ISP, lo más probable es que deba usar un servidor de correo ISP para recibir su correo. Su dirección de correo electrónico es una cuenta de correo con usuario y contraseña para acceder a su correo. Su dirección de correo electrónico suele ser su nombre de usuario y el nombre de dominio del ISP. Por ejemplo, el nombre de usuario **juan** para un dominio ISP denominado **mired.com**, tendrá la dirección **juan@mired.com**. La dirección de servidor de correo real podría ser algo como **mail.mired.com**. El usuario **juan** iniciaría sesión en el servidor **mail.mired.com** usando el nombre de usuario **juan** y la contraseña para acceder a su correo enviado a la dirección **juan@mired.com**. Los clientes de correo, como Evolution, KMail, Balsa y Thunderbird, permiten configurar una bandeja de entrada para dicha cuenta y acceder desde el servidor de correo de su ISP, para revisar y descargar correo recibido. Debe especificar el protocolo usado por el servidor de correo. Éste suele ser POP o IMAP. Dicho procedimiento se utiliza para cualquier servidor remoto. Al utilizar una dirección de servidor de correo, puede acceder a su cuenta con el nombre de usuario y contraseña.

SUGERENCIA Muchos clientes de correo, como Mutt y Thunderbird, dan soporte directo a IMAP y POP.

Si tiene más de una cuenta de correo electrónico remota, en vez de crear bandejas de entrada separadas para cada cliente de correo, puede hacer que el correo de esas cuentas se envíe directamente a la bandeja de entrada mantenida por su sistema Linux, para su cuenta de usuario. Todo su correo, ya sea de otros usuarios en su sistema Linux o cuentas de correo remotas, aparecerá en su bandeja de entrada local. Dicha característica es útil usando un cliente de correo, como Mail, sin la capacidad para acceder a correo en su servidor ISP. Implemente dicha característica con Fetchmail, para revisar servidores de correo remotos y descargarlos a su bandeja de entrada local, mismo que aparece como correo recibido recientemente (deberá estar conectado a Internet o la red del servidor de correo).

Para utilizar Fetchmail, tiene que conocer la dirección de Internet y el protocolo del servidor de correo remoto. La mayor parte de los servidores de correo manejan el protocolo POP3, pero otros pueden usar IMAP o POP2. Escriba **fetchmail** en la línea de comandos con la dirección de servidor de correo y cualquier opción necesaria. El protocolo de correo se indica con la opción **-p** y el tipo de servidor de correo, generalmente POP3. Si su nombre de usuario del correo electrónico difiere de su nombre de inicio de sesión de Linux, utilice la opción **-u** y el nombre del correo electrónico. Una vez ejecutado el comando **fetchmail**, se le pedirá contraseña. La sintaxis del comando **fetchmail** para un servidor de correo POP3 se ve así:

```
fetchmail -p POP3 -u nombredeusuario mail-servidor
```

Para utilizar Fetchmail, conéctese a su ISP y después inserte el comando **fetchmail** con opciones y nombre del servidor POP en la línea de comandos. Verá mensajes indicando si el correo está ahí y, de estarlo, cuantos mensajes se encuentran en descarga. Puede usar un cliente de correo para leer el mensaje desde su bandeja de entrada. Puede ejecutar Fetchmail en el modo de daemon para revisar automáticamente el correo. Debe incluir una opción especificando el intervalo en segundos para revisar correo.

```
fetchmail -d 1200
```

Puede especificar opciones como tipo de servidor, nombre de usuario y contraseña en un archivo **.fetchmailrc** dentro del directorio home. También tiene entradas para otros servidores de correo y cuentas que pudiera tener. Una vez configurado, puede insertar **fetchmail** sin argumentos; leerá las entradas para su archivo **.fetchmailrc**. También haga entradas directamente en el archivo **.fetchmailrc**. Una entrada en el archivo **.fetchmailrc** para una cuenta de correo particular consta de varios campos y valores: servidor, protocolo, nombre de usuario y contraseña. *Servidor* se utiliza para especificar el nombre del servidor de correo, mientras *protocolo* señala el tipo de protocolo utilizado. Observe que también puede especificar contraseña, en vez de ingresarla cada vez que Fetchmail acceda al servidor de correo.

Listas de correos

Como opción a los grupos de noticias, puede suscribirse a listas de correo. Los usuarios de las listas de correos reciben automáticamente mensajes y artículos enviados a la lista. Las listas de correo trabajan de manera muy similar a los alias de correo, transmitiendo mensajes a todos los usuarios de la lista. Las listas de correo fueron diseñadas para servir a pequeños grupos especializados de gente. En vez de publicar artículos para lectura de cualquier usuario, sólo son recibidos por quienes se suscriben. Gran cantidad de listas de correo, con amplia variedad de temas, están disponibles para Linux. Por ejemplo, en el sitio **gnome.org**, puede suscribirse a cualquiera de las listas sobre temas de GNOME, como **gnome-themes-list@gnome.org**, envíe una petición a **gnome-themes-list-request@gnome.org**. En **linux.org**, puede vincularse con sitios de soporte de listas de correo orientadas a Linux, como el sitio Web Linux Mailing Lists. Existen listas para temas como el kernel de Linux, administración, seguridad y diferentes distribuciones. Por ejemplo, **linux-admin** cubre temas de administración y **linux-apps** discute aplicaciones de software; **vger.kernel.org** ofrece servicios de listas de correo para desarrolladores de kernel de Linux.

NOTA Puede utilizar los programas Mailman y Majordomo para administrar automáticamente sus listas de correo. Mailman es el administrador de listas de correo de GNU (list.org). Puede encontrar más información acerca de Majordomo en greatcircle.com/majordomo y acerca de Mailman en sourceforge.net.

Noticias Usenet

Usenet es un sistema de correo abierto con el que los usuarios publican mensajes que pueden contener noticias, discusiones y opiniones. Opera como una bandeja de entrada en que cualquier usuario de su sistema Linux puede leer o enviar mensajes. Los mensajes del usuario se incorporan en archivos Usenet, distribuidos a cualquier sistema suscrito para recibirlos. A cada sistema que recibe archivos Usenet se le conoce como *sitio*. Ciertos sitios realizan operaciones de organización y distribución para Usenet, recibiendo mensajes desde otros sitios y organizados en archivos Usenet, para luego ser transmitidos a muchos otros sitios. Tales sitios se conocen como *sitios de espina dorsal*, y operan como publicadores, recibiendo artículos y organizándolos en grupos diferentes.

Para acceder a noticias de Usenet, necesita acceder un servidor de noticias. Un servidor de noticias recibe alimentaciones diarias de noticias de Usenet y los pone a disposición de otros sistemas. Su red puede tener un sistema operando como servidor de noticias. Si está usa ISP, es probable mantenga un servidor de noticias en su ISP para ser usado. Para leer artículos Usenet, requiere un *lector de noticias* (programa cliente que conecta a un servidor de noticias y accede artículos). En Internet y redes TCP/IP, los servidores de noticias se comunican con lectores de noticias mediante el protocolo de red de transferencia de noticias (NNTP, Network News Transfer Protocol), a menudo conocidos como servidores de noticias NNTP. O quizás también tenga que crear un servidor de noticias propio en su sistema Linux, para ejecutar un servicio de noticias Usenet local o descargar y mantener un conjunto completo de artículos Usenet. Varios programas de Linux, denominados *agentes de transporte de noticias*, pueden usarse para crear dicho servidor. Este capítulo se concentra en los diversos lectores de noticias disponibles para su plataforma Linux.

Los archivos Usenet fueron diseñados originalmente para funcionar como periódicos. A los mensajes contenidos en archivos se les conoce como *artículos*. Un usuario puede escribir un artículo, publicarlo en Usenet y hacer que se distribuya de inmediato a otros sistemas de todo el mundo. Alguien podría leer el artículo en Usenet, en vez de esperar una publicación del periódico. Los archivos Usenet, por sí solos, se organizaron como publicaciones de periódico. Debido a que los periódicos están diseñados para dirigirse a grupos específicos, los Usenet se organizaron de acuerdo con grupos llamados *grupos de noticias*. Cuando un usuario publica un artículo, está diseñado para un grupo de noticias específico. Si otro usuario quiere leer ese artículo, leerá los pertenecientes a ese grupo de noticias. Puede considerar que cada grupo de noticias es una revista actualizada constantemente. Por ejemplo, para leer artículos en el sistema operativo Linux, puede acceder al grupo de noticias Usenet en Linux. Los archivos Usenet también se utilizan como periódicos murales en que la gente lleva a cabo debates. Nuevamente, dichos archivos se clasifican en grupos de noticias, aunque sus artículos se leen más como conversaciones que artículos de periódico. También puede crear artículos propios, para después agregarlos a un grupo de noticias que otros lean. A la adición de un artículo a un grupo de noticias se le llama *publicación* de un artículo.

NOTA El sitio Web de Google mantiene acceso en línea a grupos de noticias Usenet. Tiene la capacidad adicional de permitirle buscar archivos de grupos de noticias extensos. También puede localizar artículos de manera fácil en temas similares residentes en grupos de noticias distintos. Otros sitios como Yahoo mantienen sus propios grupos de noticias operando de la misma manera que grupos de noticias Usenet, pero con más supervisión.

Linux tiene grupos de noticias sobre varios temas. Algunos para discusión y otros como fuentes de información acerca de desarrollos recientes. En algunos, puede pedir ayuda para problemas específicos. Aquí se proporciona una selección de grupos de noticias populares de Linux:

Grupo de noticias	Tema
comp.os.linux.announce	Anuncios de desarrollos de Linux
comp.os.linux.admin	Preguntas sobre administración de sistemas
comp.os.linux.misc	Temas y preguntas especiales
comp.os.linux.setup	Problemas de instalación
comp.os.linux.help	Preguntas y respuestas a problemas particulares
linux.help	Ayuda para problemas de Linux

Lectores de noticias

Los artículos Usenet se leen con un lector de noticias, como KNode, Pan, Mozilla, trn o tin, que le permiten seleccionar primero un grupo de noticias específico y después leer artículos en éste. Un lector de noticias opera con interfaz de usuario, permitiéndole explorar y seleccionar artículos específicos para leer, guardar o imprimir. Casi todos los lectores de noticias emplean una característica de recuperación sofisticada llamada *cadena* reuniendo artículos en una misma discusión o tema. Los lectores de noticias están diseñados para operar usando ciertos tipos de interfaces. Por ejemplo, KNode es un lector de noticias de KDE con interfaz KDE, diseñada para el escritorio KDE. Pan tiene una interfaz GNOME, diseñada para operar en el escritorio de GNOME. Pine es un lector de noticias basado en cursor, lo que significa que proporciona una interfaz de pantalla completa con puede trabajar mediante un cursor basado en pantalla para mover con teclas de flechas. No permite el uso del ratón ni cualquier otra característica de GUI. El programa tin usa una interfaz de línea de comandos simple, con soporte limitado para cursor. Casi todos los comandos deben ingresarse, pulsando después ENTER para ejecutarlos. Varios lectores de noticias populares se muestran en la lista de la tabla 13-3.

NOTA Actualmente, hay varios lectores de noticias en desarrollo para GNOME y KDE. Puede revisar los lectores de noticias de KDE en la lista de software del sitio Web de K Desktop, en kde-apps.org. En cuanto a lectores de noticias de GNOME, revise las herramientas de Internet, en el mapa de software del sitio Web de GNOME en gnome-files.org.

Casi todos los lectores de noticias pueden leer Usenet proporcionado por servidores de noticias recurriendo a NNTP. Muchos servidores de noticias remotos están disponibles en Internet. Los lectores de noticias de escritorio, como KNode y Pan, funcionan bajo una dirección específica de Internet de un servidor de noticias remoto en sus propias opciones de configuración. Sin embargo, varios lectores de noticias basados en shell, como trn y tin, obtienen la dirección de Internet a partir del servidor de noticias de la variable de shell **NNTPSERVER**. Antes de conectarse a un servidor de noticias remoto con estos lectores de noticias, primero debe asignar la dirección Internet del servidor de noticias a la variable de shell **NNTPSERVER** y luego exportar esa variable. Puede colocar la asignación y exportar **NNTPSERVER** a un archivo de inicialización, como **.bash_profile**, para realizarse automáticamente siempre que inicie sesión. Los administradores pueden colocar esta

Lector de noticias	Descripción
Pan	Lector de noticias del escritorio de GNOME
KNode	Lector de noticias del escritorio de KDE
Thunderbird	Cliente de correo con capacidades de lector de noticias (basado en X)
Sylpheed	Lector de noticias parecido a Windows de GNOME
slrn	Lector de noticias (basado en cursor)
Emacs	Editor Emacs, cliente de correo y lector de noticias (basado en cursor)
trn	Lector de noticias (interfaz de línea de comandos)
NewsBin	Lector de noticias (versión de Windows que trabaja bajo Wine)

TABLA 13-3 Lectores de noticias de Linux

entrada en el archivo `/etc/profile`, en el caso de un servidor de noticias disponible para todos los usuarios del sistema.

```
$ NNTPSERVER=news.dominio.com
$ export NNTPSERVER
```

El lector de noticias **slrn** está basado en pantalla. Los comandos se despliegan a lo largo de la parte superior de la pantalla y puede ejecutarse mediante teclas mostradas en la lista. Existen diferentes tipos de pantallas para la lista de grupos de noticias, lista de artículos y contenido de artículo, cada una con su propio conjunto de comandos. Una pantalla inicial muestra una lista de grupos de noticias a los que está suscrito, con comandos para publicar, mostrar listas y darse de baja de sus grupos de noticias. Cuando inicia **slrn** por vez primera, deberá crear un archivo `.jnewsrsrc` en su directorio home. Use el siguiente comando: **slrn -f .jnewsrsrc -create**. También, deberá configurar la variable **NNTPSERVER** y asegurarse de que se exporte.

Las características de lector de noticias de **slrn** presentan una nueva utilidad llamada **slrnpull**, que puede usarse para descargar artículos automáticamente en grupos de noticias específicos. Esto permite ver sus grupos de noticias seleccionados sin estar en línea. La utilidad **slrnpull** fue diseñada como versión simple de un solo usuario de Leafnode; accederá a su servidor de noticias y descargará grupos de noticias designados, haciéndolos disponibles mediante **slrn**, siempre que el usuario elija examinarlos. Los artículos de grupos de noticias se descargan al directorio **SLRNPULL_ROOT**, por lo general `/var/spool/slrnpull`. Los grupos de noticias seleccionados para descargarse se insertan en el archivo de configuración **slrnpull.conf**, colocado en el directorio **SLRNPULL_ROOT**. En este archivo, puede especificar cuántos artículos descargar de cada grupo y cuándo deben expirar. Para utilizar **slrn** con **slrnpull**, tendrá que configurar aún más el archivo `.slrnrc`, para hacer referencia a los directorios de **slrnpull**, donde se almacenan los archivos de grupos de noticias.

***NOTA** Varios lectores de noticias basados en Windows, como el popular Newsbin, se ejecutarán en Linux, a través del emulador de Windows, Wine. Para que funcione el lector de noticias, debe seguir las instrucciones de configuración, que a menudo requieren DLL específicos de Windows. En el caso de Newsbin, revise el foro de Newsbin para Linux. Asegúrese de agregar los DLL de Windows a su directorio `.win/drive_c/Windows/System32`.*

Agentes de transporte de noticias

Las noticias Usenet se proporcionan a través de Internet como alimentación de noticias diaria de artículos y publicaciones para sus miles de grupos de noticias. Esta alimentación de noticias se envía a sitios que ofrecen acceso a las noticias para otros sistemas, a través de lectores de noticias. Estos sitios operan como servidores de noticias; los lectores de noticias usados para acceder a éstos son sus clientes. Los software de servidores de noticias, llamados *agentes de transporte de noticias*, proporcionan las noticias a los lectores, permitiendo leer grupos de noticias y publicar artículos. Para Linux, tres de los agentes de transportes de noticias populares son INN, Leafnode y Cnews. Cnews y Leafnode son pequeños, simples y útiles para redes pequeñas. INN es más poderoso y complejo, se diseñó teniendo en mente sistemas más grandes (visite isc.org para conocer más detalles).

Las alimentaciones diarias de noticias en Usenet a menudo son grandes y consumen muchos recursos del servidor de noticias, tanto en tiempo como memoria. Por esta razón, tal vez no desee configurar su propio sistema Linux para recibir dichas alimentaciones. Si está operando en una red de sistemas Linux, designe uno de éstos como servidor de noticias e instale el agente de transporte

de noticias en éste para recibir y administrar alimentaciones de noticias de Usenet. Luego, los usuarios de otros sistemas en su red podrán acceder al servidor de noticias con sus propios lectores de noticias.

Si su red ya tiene un servidor de noticias, no necesita instalar un agente de transporte de noticias. Sólo debe usar su lector de noticias para acceder de manera remota a ese servidor (consulte **NNTPSERVER** en la sección anterior). En el caso de un ISP, dichos proveedores a menudo operan servidores de noticias propios, como KNode y Pan. Aunque, recuerde los lectores de noticias deben tomar tiempo para descargar artículos de los grupos de noticias seleccionados, además de actualizar información de todos los grupos de noticias.

También puede usar los agentes de transporte de noticias para ejecutar versiones locales de noticias, sólo para los usuarios de su sistema o red local. Para esto, instale INN, Leafnode, **slrnpull** o Cnews y configúrelos sólo para que administren grupos de noticias locales. Entonces los usuarios de su sistema podrán publicar artículos y leer noticias locales.

Clientes Web, FTP y Java

Prácticamente todas las distribuciones proporcionarán clientes Web y FTP poderosos para acceder a Internet. Muchos se instalan automáticamente y están listos para utilizarse apenas inicia por vez primera su sistema Linux. También incluye soporte para desarrollo Java completo, permitiéndole ejecutar y construir applets de Java. En este capítulo se cubrirán clientes Web, Java y FTP populares, disponibles para Linux. Los clientes Web y FTP se conectan a sitios que ejecutan servidores, usando páginas Web y archivos FTP para proporcionar servicios a usuarios.

Clientes Web

World Wide Web (WWW o Web) es una base de datos de hipertexto para diferentes tipos de información, distribuida a través de diferentes sitios en Internet. Una *base de datos de hipertexto* consta de elementos vinculados con otros que, a su vez, pueden estar vinculados con otros elementos, etc. Una vez recupera un elemento, puede utilizarlo para descubrir otros relacionados. Por ejemplo, puede encontrar un artículo de la selva tropical del Amazonas y después utilizarlo para recuperar un mapa o imagen de la selva tropical. En este sentido, una base de datos de hipertexto es como una telaraña de datos interconectados, que puede seguir de un elemento de datos a otro. La información se despliega en páginas conocidas como *páginas Web*. En una página Web, ciertas palabras clave o imágenes se resaltan para formar vínculos con otras páginas Web o elementos, como imágenes, artículos o archivos.

En su sistema Linux, puede seleccionar entre varios exploradores Web, incluidos Firefox, Konqueror, Epiphany y Lynx. Firefox, Konqueror y Epiphany son exploradores basados en X Window System, con capacidad para desplegar imágenes, sonido y video completas. Casi todas las distribuciones incluyen también el explorador Web Lynx, un explorador en modo de línea que sólo funciona con líneas de texto. K Desktop incorpora capacidades de explorador Web en su administrador de archivos, dejando que una ventana de directorio opere como explorador Web. Los exploradores de GNOME, como Express y Mnemonic, también están diseñados para ser habilitados con facilidad.

Los exploradores Web y clientes FTP suelen utilizarse para conducir transacciones seguras como iniciar sesión en sitios remotos, ordenar artículos o transferir archivos. Tales operaciones se aseguran actualmente con métodos de cifrado proporcionados por Secure Sockets Layer (SSL). Si usa un explorador para transacciones seguras, debe tener SSL permitida. Casi todos los exploradores, como Mozilla y ELinks incluyen soporte SSL. En el caso de operaciones FTP, puede usar la versión SSH de ftp, **sftp**, o Kerberos 5. Las distribuciones de Linux incluyen SSL como parte de una instalación estándar.

Direcciones URL

Un recurso de Internet se accede al usar un localizador universal de recursos (URL, Universal Resource Locator), integrado por tres elementos: protocolo de transferencia, nombre de host y nombre de ruta. El protocolo de transferencia y nombre del host se separan por dos puntos y dos diagonales, *://*. El *nombre de ruta* siempre comienza con una sola diagonal:

protocolo-transferencia://nombre-host/nombre-ruta

El *protocolo de transferencia* suele ser HTTP (Hypertext Transfer Protocol, protocolo de transferencia de hipertexto), indicando una página Web. Otros valores posibles para protocolos de transferencia son **ftp** y **file**. Como su nombre sugiere, **ftp** inicia sesiones FTP y **file** despliega un archivo local en su propio sistema, como archivos de texto o HTML. En la tabla 14-1 se muestra una lista de varios protocolos de transferencia.

El *nombre de host* es el equipo en que se localiza un sitio Web particular. Puede considerarlo como la dirección del sitio Web. Por convención, casi todos los nombres de host comienzan con **www**. En el siguiente ejemplo, el URL localiza una página Web llamada **guias.html** en el sitio Web **tldp.org**:

`http://tldp.org/guias.html`

Si no quiere acceder a una página Web en particular, puede dejar fuera la referencia de archivo, con lo que accederá automáticamente a la página de inicio del sitio Web. Para acceder a un sitio Web directamente, utilice su nombre de host. Si no se especifica una página para un sitio Web, el archivo **index.html** en el directorio más alto suele usarse como la de inicio. En el siguiente ejemplo, el usuario entra a la página de inicio de GNOME:

`http://www.gnome.org`

El nombre de ruta especifica el directorio donde el recurso puede encontrarse en el sistema host, además del nombre de archivo del recurso. Por ejemplo, **/pub/Linux/nuevosdatos.html** hace referencia a un documento HTML llamado **nuevosdatos**, localizado en el directorio **/pub/Linux**.

La extensión de archivo del recurso indica el tipo de acción que se tomará. Una imagen tiene una extensión **.gif** o **.jpeg** y se convierte para ser desplegada. Un archivo de sonido tiene extensión **.au** o **.wav** y se reproduce. El siguiente URL refiere un archivo **.gif**. En vez de desplegar una página Web, el explorador invoca un visor de imágenes para desplegar la imagen. En la tabla 14-2 se muestra una lista de extensiones de archivo más comunes.

`http://www.tren.com/motor/motor1.gif`

Exploradores Web

La mayor parte de exploradores Web están diseñados para acceder varios tipos diferentes de información. Los exploradores Web acceden a páginas en un sitio Web remoto o un archivo en su

Protocolo	Descripción
http	Usa el protocolo de transferencia de hipertexto
ftp	Utiliza el protocolo de transferencia de archivos (FTP, File Transfer Protocol) para conexiones FTP anónimas.
telnet	Crea una conexión Telnet
news	Lee noticias de Usenet; usa el protocolo de transferencia de noticias en red (NNTP, Network News Transfer Protocol).

TABLA 14-1 Protocolos Web

Tipo de archivo	Descripción
.html	Documento de página Web formado con HTML (Hypertext Markup Language, lenguaje de marcado de hipertexto)
Archivos gráficos	
.gif	Imágenes, que utilizan compresión GIF
.jpeg	Imágenes, que utilizan compresión JPEG
.png	Imágenes, que utilizan compresión PNG (Portable Network Graphics)
Archivos de audio	
.au	Archivo de audio de Sun (Unix)
.wav	Archivo de audio de Microsoft Windows
.aiff	Archivo de audio de Macintosh
Archivos de video	
.QT	Archivo de video QuickTime, multiplataforma
.mpeg	Archivo de video
.avi	Archivo de video de Microsoft Windows

TABLA 14-2 Tipos de archivo Web

propio sistema. Algunos exploradores también pueden acceder servidores de noticias o un sitio FTP. El tipo de información de un sitio se especifica con la palabra clave **http** para sitios Web, **nntp** para servidores de noticias, **ftp** para sitios FTP o **file** para archivos de su sistema. Como ya se observó, hay varios exploradores populares disponibles para Linux. Aquí se describen tres especiales: Mozilla, Konqueror y Lynx. Mozilla es un explorador Web basado en X Window System, capaz de desplegar imágenes, video y sonido, además de operar como lector de noticias y programa de correo. Konqueror es el administrador de archivos de K Desktop. KDE tiene capacidades integradas de exploración Web, completas en el administrador de archivos Konqueror, permitiéndole acceder sin restricciones la Web o sus archivos del sistema mediante la misma aplicación. Lynx y ELinks son exploradores de línea de comandos, sin capacidades gráficas, pero en cualquier otro aspecto son exploradores Web totalmente funcionales.

Para buscar archivos en sitios FTP, use los motores de búsqueda proporcionados por sitios Web como Yahoo!, Google o Lycos. Estos suelen buscar páginas Web y archivos FTP. Para encontrar una página Web particular en Internet, utilice cualquiera de estos motores de búsqueda o realice búsquedas desde cualquier portal Web. Las búsquedas se han convertido en un servicio estándar para casi todos los sitios Web. Estas se llevan a cabo en documentos dentro del sitio Web, mediante indexadores de búsqueda configurados y mantenidos por programas como ht://Dig. Los sitios que utilizan ht://Dig recurren a una interfaz de búsqueda de página Web estándar. Las bases de datos de hipertexto están diseñadas para acceder cualquier tipo de datos, ya sea texto, imágenes, sonido o incluso video. Si puede o no acceder realmente a estos datos, dependerá en gran medida del tipo de explorador que emplee.

El marco estructural de Mozilla

Mozilla es un proyecto de fuente abierta basado en el código del explorador Netscape original, que ofrece un marco de desarrollo para aplicaciones basadas en Web, sobre todo el explorador y cliente de correo electrónico. Originalmente, el objetivo del proyecto Mozilla fue proporcionar un explorador Web de usuario final, llamado Mozilla. El propósito ha cambiado y ahora busca brindar

Sitio Web	Descripción
mozilla.org	El proyecto Mozilla
mozdev.org	Extensiones y plug-ins de Mozilla
oreillynet.com/mozilla	Documentación y noticias de Mozilla
mozillazine.org	Noticias y artículos de Mozilla
mozillanews.org	Noticias y artículos de Mozilla
bugzilla.org	Reporte de errores y seguimiento del sistema Mozilla

TABLA 14-3 Recursos de Mozilla

un marco estructural de desarrollo que cualquiera pueda utilizar para crear aplicaciones Web, aunque el proyecto también presenta aplicaciones propias. En la tabla 14-3 se muestra una lista de recursos Mozilla.

Actualmente el marco estructural se utiliza para productos Mozilla como el explorador Web Firefox y el cliente de correo Thunderbird, además de productos que no son de Mozilla, como los exploradores Web Netscape, Euphoria y Galeon. Además, la estructura puede extenderse fácilmente para dar soporte a gran cantidad de complementos en forma de plug-ins y extensiones. El sitio del proyecto Mozilla es **mozilla.org** y el sitio que suele utilizarse para plug-ins y extensiones de desarrollo es **mozdev.org**.

El producto de primera generación del proyecto Mozilla fue el explorador Web de Mozilla, aún disponible. Como el Netscape original, incluía cliente de correo y lector de noticias, todo en una interfaz integrada. La segunda generación de productos dividió el paquete integrado en dos aplicaciones independientes y separadas, el explorador Firefox, así como el cliente de correo electrónico y lector de noticias Thunderbird. También se encuentran bajo desarrollo el explorador Web Camino, para Mac OS X y la aplicación de calendario Sunbird.

En 1998, Netscape hizo su código fuente disponible de manera libre bajo la licencia pública de Netscape (NPL, Netscape Public License). Mozilla se desarrolla con un modelo de fuente abierta, de manera parecida a Linux, KDE y GNOME. Los desarrolladores pueden subir modificaciones y adiciones en Internet al sitio Web de Mozilla. Los lanzamientos de Mozilla se conocen como Milestones y los productos de Mozilla se lanzan actualmente bajo la licencia NPL, para modificaciones del código mozilla y la licencia MPL (Mozilla Public License, licencia pública de Mozilla) para nuevas adiciones.

El explorador Web Firefox

Firefox es la nueva generación de exploradores basados en código fuente base de Netscape, conocido como mozilla. En versiones actuales, casi todas las distribuciones usan Firefox como explorador primario, en lugar de Netscape. Firefox es un explorador afinado que ofrece acceso Web rápido y protección segura ante spyware invasor.

Firefox es una aplicación de X Window System operando desde cualquier escritorio, incluidos GNOME, KDE y XFce. Firefox se instala como opción predeterminada, con una entrada en el menú Internet del menú principal y un icono en los diferentes paneles. Cuando se abre, Firefox despliega un área en la parte superior de la ventana para insertar una dirección URL y series de botones, para varias operaciones en páginas Web, como navegación por página. Los menús desplegables en la barra de menús de la parte superior, proporcionan acceso a características de Firefox como Herramientas, Ver y Marcadores.

A la derecha del cuadro de URL se encuentra un cuadro de búsqueda, dónde utilizar diferentes motores de búsqueda para buscar en Web, sitios seleccionados o elementos en particular. Un menú

emergente permite seleccionar el motor de búsqueda. Los motores de búsqueda incluidos actualmente son Google, Yahoo, Amazon y eBay, junto con Dictionary.es, para buscar definiciones de palabras. Firefox también presenta vínculos de botón y páginas con pestañas. Puede arrastrar el URL desde el cuadro URL a la barra de botones de vínculos, para crear un botón con el que puede acceder al sitio. Utilice esto para sitios a los que accede frecuentemente.

Para explorar de manera sencilla, Firefox presenta paneles con pestañas para desplegar páginas Web. Para abrir un panel de pestaña vacío, oprima CTRL-T o seleccione

Nueva pestaña, en el menú Archivo. Para desplegar una página en ese panel, arrastre su URL del cuadro URL o la lista de marcadores al panel. Se pueden tener varios paneles abiertos al mismo tiempo y se va de una página a otra haciendo clic en sus fichas. Para que todos sus botones de vínculos se abran como paneles de pestaña, haga clic con el botón derecho en la barra de vínculo y seleccione Abrir todo en pestañas.

Firefox denomina *marcadores* a los URL de páginas Web que quiere conservar, marcando aquellas páginas que quiere acceder directamente. El menú Marcadores le permite agregar sus páginas Web favoritas. Luego puede ver una lista de marcadores y seleccionar uno para verlo. También puede editar la lista, agregando marcadores nuevos o quitando viejos. El Historial es una lista de URL a la que ha accedido. El cuadro URL también presenta un menú emergente mostrando una lista de sitios previos del historial. Marcadores e Historial pueden verse como barras laterales, para seleccionarlas en el menú Ver.

Cuando descarga un archivo con Firefox, el Administrador de descargas se encarga de administrarlo. Puede descargar varios archivos en forma paralela. Es posible desplegar las descargas en la barra de herramientas Administrador de descargas. También puede cancelar una descarga en cualquier momento o pausarla y reanudarla después. Al hacer clic con el botón derecho en una entrada de descarga, se desplegará el sitio desde el que hizo la descarga, al igual que el directorio donde se guardó. Para quitar una entrada, oprima Eliminar, en la barra de herramientas.

El menú Preferencias (Editar | Preferencias) de Firefox permite configurar varias opciones. Firefox también da soporte a características avanzadas como cookies, formularios, imágenes y administración de contraseña. Puede seleccionar que se eliminen las cookies de los sitios, que se rellenen automáticamente los formularios, que no se desplieguen imágenes de sitios y que se configure información de inicio de sesión, como usuarios y contraseñas para sitios seleccionados. Hay acceso a la configuración de opciones para características generales, privacidad, Web y Administración de descarga, además de características avanzadas. En la sección Principal, puede determinar página de inicio, las fuentes de las páginas y sus colores, además de las opciones de conexión como información de proxy. En Privacidad puede controlar la información guardada (como el número de sitios de historial a ser recordados y el historial de descargas), configurar las directivas para guardar cookies y configurar el tamaño de su caché. Toda esta información puede borrarse manualmente. En Contenido se controlan los menús desplegables, si se permite la instalación de software y se habilitación de JavaScript. El panel Administración de descarga permite configurar sus operaciones de descarga, permitiéndole especificar un directorio predeterminado para descargas, si se le pedirá uno automáticamente y los plug-ins que tal vez quiera ejecutar automáticamente, en ciertos tipos de archivos, como Adobe Acrobat para archivos de Adobe PDF. Desde el panel Avanzado controla características más complejas de exploración: desplazamiento, niveles de seguridad y administración de certificados.

Si está en una red que se conecta a Internet a través de firewall, debe usar la pantalla Proxies para introducir la dirección del equipo de puerta de enlace de su firewall de red. Un *firewall* es un equipo operando como puerta de enlace controlado de Internet para su red. Existen varios tipos de firewalls. Los más restrictivos utilizan programas llamados *proxies*, recibiendo peticiones de Internet de usuarios y después hacen esas peticiones en su representación. No hay conexión directa a Internet.

El administrador de archivos de K Desktop: Konqueror

Si está utilizando K Desktop, puede emplear una ventana del administrador de archivos como explorador Web. El administrador de archivos de K Desktop se configura automáticamente para actuar como explorador Web. Puede desplegar páginas Web, con imágenes y vínculos incluidos. El administrador de archivos de K Desktop soporta operaciones de página Web estándar, entre ellas, avance y retroceso a través de páginas a las que ha accedido. Haga clic en un vínculo para acceder y desplegar la página Web de referencia. En este sentido, Web se integra perfectamente en K Desktop.

Exploradores Web de GNOME: Nautilus, Galeon y Epiphany

También están disponibles varios otros exploradores Web basados en GNOME. Epiphany, Galeon y Kazehakase soportan operaciones de Web estándar. Epiphany es un explorador Web de GNOME, diseñado para ser una interfaz rápida y simple, trabaja bien como explorador simple con una interfaz limpia. También está integrada en el escritorio, presentando un applet de descarga que continuará, incluso después de cerrar Epiphany. Además, soporta paneles con pestañas para acceso a varios sitios Web. Encontrará más información acerca de Epiphany en epiphany.mozdev.org. Galeon es un explorador rápido y ligero, también basado en el motor de explorador Mozilla (Gecko). Kazehakase pone énfasis en una interfaz personalizable, con cuadros de descarga y marcas RSS.

Para GNOME, puede descargar herramientas de soporte, como RSSOwl, para desplegar alimentaciones de noticias y GNOME Download Manager (Gwget), para controlar descargas basadas en Web. Downloader, para el cliente X, es útil con descargas de archivos FTP y Web. Tiene numerosas características que le permiten controlar velocidades de descarga, además de subdirectorios para ello.

Lynx y ELinks: exploradores de modo de línea

Lynx es un explorador en modo de línea que se usa sin X Window System. Una página Web se despliega sólo como texto. Una página de texto puede contener vínculos a otros recursos de Internet, pero sin desplegar imagen, video o sonido. Excepto por limitaciones de despliegue, Lynx es un explorador Web totalmente funcional. Puede utilizar Lynx para descargar archivos o establecer conexiones Telnet. Toda la información en Web continúa accesible. Debido a que no requiere muchos de los recursos demandados por los exploradores gráficos, Lynx puede operar a mayor velocidad, desplegando rápidamente el texto de una página Web. Para iniciar el explorador Lynx, inserte el comando **lynx** en la línea de comandos y oprima ENTER.

Otro explorador útil basado en texto, incluido en casi todas las distribuciones es ELinks. Se trata de un explorador poderoso basado en pantalla que incluye características como marcos, formularios y soporte a tablas. También permite la codificación segura SSL. Para iniciar ELinks, inserte el comando **elinks** en una ventana de terminal.

Creación de su propio sitio Web

Para crear su propio sitio Web, necesita acceder un servidor Web. Casi todas las distribuciones de Linux instalan automáticamente el servidor Web Apache en sus sistemas. También es posible rentar espacio para una página Web en un servidor remoto (un servicio que muchos ISP proporcionan, algunos gratis). El directorio para configurar su servidor Web Apache para páginas de sitio Web suele ser **/var/httpd/html**. Otros servidores proporcionan un directorio para su página de inicio en el que puede colocar páginas Web creadas. Coloque la página de inicio en ese directorio, luego cree otros subdirectorios a los que puede vincularse con sus propias páginas Web. No es tan difícil crear páginas Web. Los vínculos llevan al usuario de una página a otra a través del sitio Web. Incluso puede crear vínculos con páginas Web o recursos en otros sitios. Hay muchos textos excelentes disponibles sobre creación y administración de páginas Web.

Las páginas Web se crean al utilizar HTML o la nueva versión extendida, XML (eXtended Markup Language, longitud de marcado extendido). Ambos utilizan un subconjunto del lenguaje estándar de marcado generalizado (SGML, Standard Generalized Markup Language). Para crear un documento HTML o XML, basta insertar etiquetas de HTML o XML en un archivo de texto. En este sentido, la creación de una página Web es tan simple como emplear procesadores de palabra basados en etiquetas. Use etiquetas de HTML para formar texto de modo que se desplieguen como página Web. Las etiquetas XML incluyen información detallada acerca de una conexión particular, como datos de objeto o características de transacción. La página Web por sí sola es un archivo de texto que se puede crear con cualquier editor, como Vi. Si está familiarizado con el procesamiento de palabras basado en etiquetas en sistemas Unix, lo encontrará muy similar, en cuanto a concepto, a nroff. Algunas etiquetas HTML indican encabezados, listas y párrafos, además de vínculos para hacer referencia a recursos Web.

En vez de insertar manualmente el código HTML o XML, puede usar compositores de páginas Web, instrumentados como interfaz gráfica para construir páginas Web. Los programas de creación de páginas Web pueden ayudar a crear páginas más complejas de manera sencilla, sin escribir una etiqueta HTML de manera explícita. Pero recuerde que no importa la herramienta utilice para crear una página Web, ésta por sí sola será un documento HTML. Como parte del proyecto KDE, KDE web Dev (kdewebdev.org) ofrece varias aplicaciones de desarrollo Web, como el editor Web Quanta Plus y el generador de diálogo Kommander.

NOTA Muchos editores estándar para K Desktop y GNOME incluyen características de construcción de páginas Web. Le permiten insertar vínculos y formar encabezados. Por ejemplo, el programa KEdit soporta componentes básicos de página Web basados en texto. Puede agregar encabezados, vínculos o líneas, pero no imágenes.

Java para Linux

Para el desarrollo de aplicaciones Java, use las herramientas de Java y ejecutar numerosos productos de en tal lenguaje, debe instalar Java 2 Software Development Kit (SDK, kit de desarrollo de software) y Java 2 Runtime Environment (JRE, entorno de motor en tiempo de ejecución) en su sistema. Juntos, crean Java 2 Platform, Standard Edition (J2SE). Actualmente Sun soporta y distribuye versiones de Linux de estos productos. Puede descargarlos de Sun en java.sun.com/j2se e instalarlos en su sistema. Los paquetes y aplicaciones de Java se muestran en la lista de la tabla 14-4.

Sun, Java-like, JPackage y Blackdown

Muchas distribuciones de Linux incluyen gran número de aplicaciones y soporte gratuitos de Java, como Yakarta; la mayoría fueron desarrollados originalmente por JPackage Project (jpackage.org) para usarse en Linux. Debe emplear las versiones de estos paquetes para su distribución, ya que es probable se hayan modificado especialmente para usarlos en ésta. Sin embargo, los principales, Java Runtime Environment y SDK no se incluyen. En cambio, debe recurrir a un conjunto compatible de paquetes GNU (Java-like), para ejecutar applets de Java o instalar puertos Linux JRE y SDK desde JPackage o Blackdown o descargar e instalar JRE y SDK originales de Sun. Ninguna de estas opciones es exclusiva. El servicio JPackage se encuentra disponible para distribuciones basadas en RPM como Red Hat, SUSE y Fedora, y el paquete Blackdown es popular con distribuciones como Debian y Ubuntu.

Casi todas las distribuciones como Ubuntu, Fedora y SUSE incluyen una colección parecida a Java de paquetes de soporte que permiten ejecutar operaciones de Java Runtime. No existe nombre oficial para esta colección, aunque suele conocerse como java-gci-compat, además de Java-like.

Aplicación	Descripción
Java 2 Software Development Kit (SDK)	Entorno de desarrollo de Java con un compilador, intérpretes, depurador y más: java.sun.com/j2se . Componente de la plataforma Java 2. Descargue el puerto para Linux desde JPackage o Blackdown, o directamente desde Sun.
Java 2 Runtime Environment 1.4 (J2RE)	Java Runtime Environment se utiliza para ejecutar applets Java. Descargue el puerto para Linux desde java.com , blackdown.org , jpackage.org .
Java 3D para Linux	Interfaz de programa de aplicación 3-D para programas 3-D de Java.
Entorno Java-like	Java-like Free and open Environment, consta del motor en tiempo de ejecución GNU Java (libgcj), compilador Eclipse Java (ecj) y envolturas y vínculos de soporte (java-gcj-compat).
Java Advanced Imaging (JAI) para Linux	API de Java Advanced Imaging.
Java 1.1 Development Kit (JDK) y Java 1.1 Runtime Environment (JRE) para Linux	El antiguo entorno de desarrollo Java 1.1 con un compilador, intérpretes, depurador y más. Descargue el puerto Linux para actualizar su distribución a través de blackdown.org o jpackage.org .
Servidor Web Java System	Servidor Web implementado con Java. Disponible en el sitio Web de Java: java.sun.com (comercial).
GNU Java Compiler	Java Compiler (GJC) con licencia pública GNU para compilar programas Java: gcc.gnu.org/java , libgcj.
Jakarta Project	Proyecto Apache Software Foundation para aplicaciones de Java de fuente abierta: jakarta.apache.org .

TABLA 14-4 Paquetes de Java y aplicaciones Web de Java

Esta colección proporciona un entorno libre y de fuente abierta, constando de tres paquetes: motor en tiempo de ejecución GNU Java (libgcj), compilador Eclipse Java (ecj) y, en Fedora, un conjunto de envolturas y vínculos (java-gcj-compat).

Aunque Sun soporta versiones para Linux de Java, pueden obtenerse más puertos de Linux completos y efectivos de Java del proyecto Blackdown en **www.blackdown.org**. Las distribuciones como Debian y Ubuntu son compatibles con puertos Java de Blackdown y proporcionan acceso a éstos. El proyecto Blackdown ha transferido tanto Java JRE como SDK, además de versiones previas de Java.

Las distribuciones Linux basadas en RPM, como Fedora y SUSE, recomiendan descargar Java y aplicaciones de Java desde JPackage, diseñado para que sus paquetes de Java no tengan conflicto y sean compatibles con la versión específica de Linux. Con las versiones de JPackage de Java, puede instalar y desinstalar de manera segura gran número de aplicaciones y herramientas de soporte de Java. La mejor manera de utilizar JPackage es descargar su archivo de configuración de depósito Yellowdog Updater Modified (Yum). Asegúrese de elegir el correcto para la distribución apropiada (cuando está disponible). Por desgracia, debido a restricciones de licencia, JPackage debe cobrar por el JRE principal, al igual que la aplicación SDK de Java. Éstos se incluyen en su paquete de Java.

NOTA Consulte **java.sun.com/products** para conocer una lista extensa de aplicaciones de Java.

Instalación de Java Runtime Environment: JRE

Muchos sitios Web ejecutarán aplicaciones requiriendo entorno de motor en tiempo de ejecución de Java (JRE, Java Runtime Environment). Muchas distribuciones Linux no vienen con éste, pero Java-like actualmente soporta aplicaciones compatibles como Office.org y Eclipse. Para Java de Sun, deberá descargar e instalar JRE en su sistema Linux. Obtenga una copia del sitio Web de Java de Sun (java.com). SDK y JRE están disponibles en forma de archivos comprimidos que se extraen solos, **.bin**. Estos archivos son, en realidad, secuencias de comandos shell con un archivo comprimido incrustado. (Existen instrucciones de instalación por separado.) Debido a los conflictos de nombre de archivo, no debe utilizar paquetes RPM de Sun (**.bin.rpm**). En cambio, puede descargar el paquete **.bin** y extraerlo en el directorio **/opt**. Tendrá que hacer ejecutable archivo bin de extracción automática con el comando **chmod**. El siguiente comando cambiará el archivo JRE, **jre-1_5_0_02-linux-i586-rpm.bin**, a ejecutable. Primero se le pedirá acepte los acuerdos de licencia.

```
chmod a+x jre-1_5_0_02-linux-i586-rpm.bin
./ jre-1_5_0_02-linux-i586-rpm.bin
```

El JRE se instalará en el directorio **/opt**, en este caso bajo **/opt/jre-1_5_0_02**.

Habilitación de Java Runtime Environment para Mozilla/Firefox

Para que los exploradores Web Mozilla o Firefox usen JRE, necesita crear un vínculo desde el directorio de plug-ins de Mozilla para utilizar bibliotecas de plug-ins de Java. Asegúrese de instalar primero JRE. En el directorio **/usr/lib/mozilla/plugins**, deberá crear un vínculo con la biblioteca **libjavaplugin_oji.so** en el subdirectorio **/plugin/i386/ns7** de JRE, donde “ns7” indica Netscape 7.

```
# cd /usr/lib/mozilla/plugins
# ln -s /opt/jre1.5.0_06/plugin/i386/ns7/libjavaplugin_oji.so libjavaplugin_oji.so
```

NOTA En Firefox y Mozilla, asegúrese que el soporte de Java está habilitado.

Las aplicaciones de Java

Actualmente se pueden adaptar a Linux gran número de productos y herramientas adicionales basadas en Java. Entre las herramientas se incluyen Java 3D, Java Media Framework (JMF) y JAI. Muchos productos, como el servidor Web de Java, se ejecutan directamente como los proporcionados por Sun. Puede descargar varios directamente del sitio Web de Java de Suscripción, en java.sun.com. El proyecto Jakarta (jakarta.apache.org), parte de Apache Software Foundation, proporciona herramientas y aplicaciones de fuente abierta de Java, incluidas bibliotecas, aplicaciones de servidor y motores. Jakarta, junto con otros paquetes, se incluye en casi todas las distribuciones.

Java 2 Software Development Kit

Java 2 SDK ofrece herramientas para crear y depurar sus propias applets de Java, proporcionando soporte para aplicaciones Java. El conjunto incluye applets de demostración con código fuente. Obtenga documentación detallada acerca de SDK del sitio Web de Sun en java.sun.com. Cuatro versiones principales de SDK están disponibles actualmente (1.2, 1.3, 1.4.x y 1.5 [también conocida

como 5.0]), con sus respectivas versiones de Java 2 Runtime Environment (J2RE), para 1.2, 1.3, 1.4 y 1.5 (5.0). Java SDK agrega capacidades para seguridad, soporte a interfaz gráfica de usuario (GUI) con JFC (también conocido como Swing) y mejoras de ejecución de Java, como Java 3D y Java Sound.

El SDK incluye características estándar encontradas en JDK para internacionalización, applets firmadas, formato de archivo JAR, mejoras en AWT (ventana de conjunto de herramientas), el modelo de componente de Java Beans, mejoras de red, paquete de matemáticas para números grandes, bases de datos de conectividad (JDBC), serialización de objetos y clases internas. Las aplicaciones de Java incluyen compilador (javac), depurador (jdb) y visor de applets (appletviewer). Las descripciones detalladas de estas características se encuentran en la documentación de SDK, en java.sun.com/docs.

Un applet de Java se crea de la misma forma que un programa, mediante el uso de un lenguaje de programación estándar. Primero use un editor de texto para crear el código fuente, para ser guardado con la extensión **.java**. Después use el compilador **javac**, para compilar el código fuente, generando un applet de Java. Gran número de aplicaciones del entorno de desarrollo integrado (IDE, Integrated Development Environment) están disponibles para componer applets y aplicaciones de Java. Aunque casi todos son comerciales, algunas proporcionan versiones shareware gratuitas. Un IDE brinda una GUI para construir applets de Java. Eclipse opera como plataforma de desarrollo para applets de Java.

Cientes FTP

Con los clientes FTP, se conecta a un sitio FTP correspondiente y descarga archivos desde éste. Los clientes FTP suelen usarse para descargar software de sitios FTP operando como depósitos de software. Casi todas las aplicaciones de software de Linux pueden descargarse a su sistema Linux desde tales sitios, presentando inicios de sesión anónimos, permitiéndole a cualquier usuario acceder a sus archivos. Un sitio de distribución como **ftp.redhat.com** es un ejemplo de sitio FTP, almacenando un extenso conjunto de paquetes de aplicaciones para Linux, que puede descargarse con un cliente FTP y luego instalarse de manera sencilla en su sistema. En los administradores de archivo Konqueror (KDE) y Nautilus (GNOME), se incorporan capacidades básicas de un cliente FTP. Puede utilizar una ventana del administrador de archivos para acceder a un sitio FTP y arrastrar los archivos a sus directorios locales para descargarlos. En efecto, los clientes FTP ahora también se incorporan en casi todos los exploradores Web, volviéndose la herramienta de descarga principal. Firefox en particular, tiene importantes opciones para descarga desde FTP.

Aunque los administradores de archivo y exploradores Web proporcionan acceso efectivo a sitios públicos (inicios de sesión anónimos), para acceder a sitios privados es probable necesite un cliente FTP independiente, como curl, wget, gFTP o ftp. Estos clientes permiten insertar nombres de usuarios y contraseñas con las que puede acceder a sitios FTP privados. Los clientes independientes también son útiles para descargas grandes de sitios FTP públicos, sobre todo los que tienen poco o nulo soporte para despliegue. Los clientes FTP populares se muestran en la lista de la tabla 14-5.

Transferencia de archivos en red: FTP

Con los clientes FTP, se transfieren archivos extremadamente largos de un sitio a otro. FTP puede manejar archivos de texto y binarios. Éste es uno de los protocolos de TCP/IP, operando en sistemas conectados a redes que pueden utilizar protocolos TCP/IP, como Internet. FTP realiza un inicio de sesión remoto a otra cuenta en otro sistema conectado a través de su red. Una vez inicia sesión en otro sistema, se transfieren archivos a éste, o desde éste. Para iniciar sesión, necesita conocer nombre y contraseña de inicio de sesión de la cuenta en el sistema remoto. Por ejemplo, si tiene cuentas en sitios diferentes de Internet, puede usar FTP para transferir archivos de uno a otro.

Cliente FTP	Descripción
Firefox	Explorador Web y FTP de Mozilla
Konqueror	Administrador de archivos de K Desktop
Nautilus	Administrador de archivos de GNOME
gFTP	Cliente FTP de GNOME
ftp	Cliente FTP de línea de comandos
lftp	Cliente FTP de línea de comandos con capacidad de varias conexiones
NcFTP	Cliente FTP basado en pantalla
curl	Cliente de transferencia de Internet (FTP y HTTP)

TABLA 14-5 Clientes FTP de Linux

Sin embargo, muchos sitios en Internet permiten acceso público con FTP. Tales sitios sirven como depósitos para archivos grandes a los que cualquiera accede para descargas. A estos sitios suele conocerseles como *sitios FTP* y, en muchas casos, sus direcciones de Internet comienzan generalmente con la palabra *ftp*, como **ftp.gnome.org** o **ftp.redhat.com**. Estos sitios públicos permiten inicios de sesión FTP anónimos. Como nombre de inicio de sesión, use la palabra “anonymous” y para la contraseña, escriba su dirección de correo electrónico. Entonces podrá transferir archivos de ese sitio a su sistema.

Tiene opción de realizar operaciones con FTP mediante un programa de cliente FTP; en el caso de sistemas Linux, seleccione entre varios clientes FTP. Muchos ahora operan con una GUI, como GNOME. Algunos, como Firefox, tienen capacidades limitadas, mientras otros, como NcFTP, incluyen un extenso conjunto de mejoras. El cliente FTP original, *ftp*, es igual de efectivo, aunque no resulta tan fácil de usar. Opera mediante una interfaz de línea de comandos simple sin requerir GUI o soporte de cursor, como otros clientes.

Internet tiene gran cantidad de sitios abiertos al acceso público conteniendo archivos que cualquiera puede obtener, usando programas de transferencia de archivos. Sin embargo, localizar un archivo puede ser difícil, a menos que ya sepa dónde se encuentra. Para buscar archivos en sitios FTP, use los motores de búsqueda proporcionados por sitios Web, como Yahoo!, Google o Lycos. Para software de Linux, revise sitios como **freshmeat.net**, **sourceforge.net**, **rpmfind.net**, **freshrpms.net**, **apps.kde.com** y **gnome.org**. Estos sitios generalmente buscan en páginas Web y archivos FTP.

FTP basado en exploradores Web: Firefox

Puede acceder un sitio FTP y descargar archivos de éste con cualquier explorador. Un explorador Web es efectivo para revisar un sitio FTP y ver qué archivos se encuentran allí. Cuando accede a un sitio FTP con un explorador Web, la lista entera de archivos de un directorio se muestra en una lista con aspecto de página Web. Puede ir a un subdirectorio haciendo clic en su entrada. Con Firefox, puede explorar de manera sencilla un sitio FTP para descargar archivos: sólo haga clic en el vínculo para descarga. Esto iniciará la operación de transferencia, al abrir un cuadro para seleccionar su directorio local y nombre del archivo. El nombre predeterminado es el mismo del sistema remoto. Puede manejar sus descargas con el Administrador de descargas, que permite cancelar una operación de descarga en progreso o eliminar otras solicitadas. El administrador mostrará tiempo restante, velocidad y cantidad transferida de su descarga actual. Los exploradores son útiles para

localizar archivos individuales, aunque no para descargar gran conjunto de archivos, como suele necesitarse para una actualización del sistema.

El administrador de archivos de K Desktop: Konqueror

En K Desktop, el administrador de archivos de escritorio (Konqueror) tiene capacidad FTP incluida. La operación FTP ha sido integrada perfectamente en operaciones de archivo de escritorio estándar. Descargar archivos de un sitio FTP es tan simple como copiar archivos y arrastrarlos de la ventana de un directorio a otra, sólo que uno de los directorios está localizado en un sitio FTP remoto. En K Desktop, use una ventana del administrador de archivos para acceder a un sitio FTP remoto. Los archivos del directorio remoto se muestran en una lista, igual que sus archivos locales. Para descargar archivos de un sitio FTP, abra una ventana para acceder a ese sitio, insertando el URL del sitio FTP en el cuadro Ubicación de la ventana. Abra el directorio deseado y después otra ventana de su directorio local donde prefiera se copien los archivos remotos. En la ventana mostrando los archivos FTP, seleccione aquellos quiera descargar. Después sólo haga clic y arrastre dichos archivos a la ventana para el directorio local. Aparece un menú emergente con opciones para Copiar, Vincular o Mover. Seleccione Copiar. Entonces se descargarán los archivos seleccionados. Se abre otra ventana, mostrando progreso de la descarga y desplegando el nombre de cada archivo, junto con una barra indicando porcentaje de descarga hasta el momento.

FTP en el escritorio de GNOME: Nautilus

La mejor forma de descargar archivos consiste en utilizar las opciones de FTP incluidas en el administrador de archivos de GNOME, Nautilus. La operación FTP ha sido integrada perfectamente en operaciones de escritorio estándar. Para descargar archivos desde un sitio FTP, basta arrastrar los archivos de una ventana de directorio a otra, donde uno de los directorios está localizado en un sitio FTP remoto. Use el administrador de archivos GNOME para acceder a un sitio FTP remoto, presentando una lista de archivos del directorio remoto, como si fueran archivos locales. Sólo inserte el URL del FTP tras el prefijo **ftp://** y oprima ENTER. Se desplegará el directorio principal del sitio FTP remoto. Use el administrador de archivos para recorrer el árbol de directorios del sitio FTP remoto hasta encontrar el archivo deseado. Después, abra otra ventana para el directorio local a donde quiere se copien los archivos remotos. En la ventana mostrando los archivos FTP, seleccione los que quiera descargar. Después oprima CTRL, haga clic y arrastre esos archivos a la ventana del directorio local. Al oprimir CTRL y hacer clic, se realiza una operación de copiado, no movimiento. Mientras los archivos se descargan, un cuadro de diálogo aparece, mostrando el progreso.

gFTP

El programa gFTP es un cliente FTP simple de GNOME, diseñado para permitir transferencias de archivos FTP estándar. La ventana gFTP contiene varios paneles. El panel de la sección superior izquierda muestra listas de archivos en su directorio local, mientras el panel superior derecho muestra una lista de su directorio remoto. Los subdirectorios tienen iconos de carpeta antes de sus nombres. Se puede distinguir el directorio principal por la entrada con dos puntos seguidos (..) y una flecha hacia arriba, en la parte superior de cada lista. Haga doble clic en la entrada del directorio para acceder a éste. Los nombres de ruta de todos sus directorios se despliegan en cuadros, arriba de cada panel. Si lo desea, puede insertar un nuevo nombre de ruta para ir a un directorio diferente.

Dos botones entre los paneles se utilizan para transferir archivos. El botón de la flecha hacia la izquierda (<) descarga archivos seleccionados en el directorio remoto y, el botón de la flecha hacia la derecha (>), sube archivos desde el directorio local. Para descargar un archivo, haga clic en éste,

en el panel del extremo derecho y después en el botón de la flecha hacia la izquierda. Cuando el archivo se descarga, su nombre aparece en el panel de la izquierda, que es su directorio local. Los menús a lo largo de la parte superior de la ventana pueden utilizarse para administrar transferencias. Un administrador de conexiones permite ingresar información de inicio de sesión acerca de un sitio determinado. Especifique si quiere realizar un inicio de sesión anónimo o si proporcionará nombre de usuario y contraseña. Haga clic en Conectar para conectarse al sitio. Un menú desplegable permite seleccionar el sitio que quiere. Las descargas interrumpidas pueden reiniciarse de manera sencilla.

wget

Con **wget** puede acceder de manera sencilla a sitios Web y FTP para directorios particulares y archivos. Los directorios pueden descargarse de manera recursiva, permitiendo copiar todo un sitio Web. **wget** toma como opción el URL del archivo o directorio deseado. Entre las operaciones útiles se incluyen **-q** para modo silencioso, **-r** para repetitivo (directorios), **-b** para descargar en segundo plano y **-c** para continuar la descarga interrumpida de un archivo. Uno de los inconvenientes es que la referencia del URL puede ser muy compleja. Debe conocer el URL; no puede localizar de manera interactiva un elemento, como haría con un cliente FTP. En el siguiente ejemplo se descarga el DVD de Fedora en segundo plano:

```
wget -b ftp://download.fedora.redhat.com/pub/fedora/linux/core/7/i386/iso/
FC-7-i386-DVD.iso
```

SUGERENCIA Con la herramienta **wget** de GNOME, puede ejecutar descargas **wget** mediante GUI.

curl

El cliente de Internet **curl** opera casi de la misma forma que **wget** pero con más flexibilidad. Puede especificar varios URL en la línea de comandos de **curl** y usar llaves para especificar varios URL coincidentes, como el caso de sitios Web diferentes con el mismo nombre de dominio. Puede hacer una lista de nombres de host de sitios Web diferentes entre llaves, seguidos por su nombre de dominio (o viceversa). También puede usar corchetes para especificar un rango de varios elementos. Esto es muy útil para descargar archivos almacenados con el mismo nombre raíz de extensiones variables, como diferentes temas de la misma revista. Para sus descargas, **curl** usa cualquier protocolo e intentará adivinar de manera inteligente qué protocolo utilizar, si no se da uno. Revise las páginas Man de **curl** para adquirir más información.

ftp

El nombre **ftp** designa al cliente FTP original, usado en sistemas Unix y Linux. El cliente **ftp** recurre a una interfaz de línea de comandos con un extenso conjunto de comandos y opciones usados para administrar sus transferencias FTP. Puede iniciar el cliente **ftp** ingresando el comando **ftp** en el indicador de comandos de shell. Si tiene un sitio específico al que quiere conectarse, incluya el nombre de ese sitio en la línea de comandos tras la palabra clave **ftp**. De otra forma, necesita conectarse al sistema remoto con el comando de **ftp open**. Después se le pedirá nombre del sistema remoto con la petición "(to)". Cuando ingrese el nombre del sistema remoto, **ftp** lo conectará al sistema y luego le pedirá nombre de inicio de sesión. La petición para el nombre de inicio de sesión consiste de la palabra "Name" y, entre paréntesis, nombre del sistema e inicio de sesión local. En ocasiones, el nombre de inicio de sesión en el sistema remoto es idéntico al del propio sistema. Si los nombres son iguales, oprima ENTER en la petición. Si son diferentes, inserte nombre

de inicio de sesión del sistema remoto. Después de escribir el nombre, se le pide la contraseña. En el siguiente ejemplo, el usuario se conecta al sistema remoto **garnet** e inicia sesión en la cuenta **roberto**:

```
$ ftp
ftp> open
(to) garnet
Conected to garnet.berkeley.edu.
220 garnet.berkeley.edu FTP server ready.
Name (garnet.berkeley.edu:root): roberto
password required
Password:
user roberto logged in
ftp>
```

Una vez que inicie sesión, puede ejecutar comandos Linux en el sistema remoto o local. Se ejecuta un comando en su sistema local en ftp escribiendo un signo de admiración antes del comando. Cualquier comando de Linux sin este signo se ejecuta en el sistema remoto. Existe una excepción a esta regla: aunque puede cambiar el comando en el sistema remoto con **cd**, para cambiar de directorio en su sistema local, necesita utilizar un comando ftp especial llamado **lcd** (local **cd**). En el siguiente ejemplo, el primer comando muestra una lista de archivos en el sistema remoto, mientras el segundo comando presenta una lista de archivos en el sistema local:

```
ftp> ls
ftp> !ls
```

El programa ftp proporciona un conjunto básico de comandos para administrar archivos y directorios en su sitio remoto, siempre y cuando tenga permiso para hacerlo (véase la tabla 14-6). Utilice **mkdir** para crear un directorio remoto y **rmdir** para eliminar uno. Utilice el comando **delete** para eliminar un archivo remoto. Con el comando **rename**, puede cambiar nombres de archivo. Cierre la conexión con un sistema con el comando **close**. Abra otra conexión si así lo desea. Para terminar la sesión ftp, use el comando **quit** o **bye**.

```
ftp> close
ftp> bye
Good-bye
$
```

Para transferir archivos al sistema remoto, y desde éste, use los comandos **get** y **put**. El comando **get** recibe archivos del sistema remoto a su sistema local y el comando **put** envía archivos de su sistema local al remoto. En un sentido, su sistema local recibe archivos *desde* el remoto y coloca archivo *en* el remoto. En el siguiente ejemplo, el archivo **clima** se envía de un sistema local al remoto mediante el comando **put**:

```
ftp> put clima
PORT command succesful.
ASCII data connection
ASCII Transfer complete.
ftp>
```

Comando	Efecto
ftp	Invoca al programa ftp.
open <i>dirección-sitio</i>	Abre una conexión con otro sistema.
close	Cierra la conexión con un sitio.
quit o bye	Termina la sesión ftp.
ls	Muestra una lista con el contenido de un directorio.
dir	Muestra una lista con el contenido de un directorio de forma larga.
get <i>nombredearchivo</i>	Envía un archivo del sistema remoto al local.
put <i>nombredearchivo</i>	Envía un archivo del sistema local al remoto.
mget <i>expresión-regular</i>	Permite descargar varios archivos al mismo tiempo desde un sistema remoto. Puede usar caracteres especiales para especificar archivos; se le pide transfiera archivo por archivo.
mput <i>expresión-regular</i>	Puede enviar varios archivos simultáneamente a un sistema remoto. Puede utilizar caracteres especiales para especificar archivos; se le pide confirmación para cada archivo transferido.
runique	Activa y desactiva almacenamiento de archivos con nombres únicos. Si ya existe un archivo con el mismo nombre en el sistema local, se genera un nuevo nombre de archivo.
reget <i>nombredearchivo</i>	Reanuda la transferencia interrumpida de un archivo, desde donde se quedó.
binary	Transfiere archivos en modo binario.
ascii	Transfiere archivos en modo ASCII.
cd <i>directorio</i>	Cambia directorios en el sistema remoto.
lcd <i>directorio</i>	Cambia directorios en el sistema local.
help o ?	Muestra una lista de comandos ftp.
mkdir <i>directorio</i>	Crea un directorio en el sistema remoto.
rmdir	Elimina un directorio remoto.
delete <i>nombredearchivo</i>	Elimina un archivo en el sistema remoto.
mdelete <i>listadearchivo</i>	Elimina varios archivos remotos al mismo tiempo.
rename	Cambia el nombre a un archivo en un sistema remoto.
hash	Despliega signos hash progresivos durante la descarga.
status	Despliega el estado actual del ftp.

TABLA 14-6 Comandos del cliente ftp

Si una descarga se interrumpe, reanúdela con el comando **reget**. Es útil para un archivo muy largo. La descarga se reanuda donde quedó, así no necesita descargar nuevamente el archivo completo. Además, asegúrese de descargar los archivos binarios en modo binario. En casi todos los sitios FTP, el modo binario es el predeterminado, pero algunos sitios pueden tener ASCII (texto)

como predeterminado. El comando **ascii** configura el modo de carácter, y el comando **binary** configura el modo binario. Casi todos los paquetes de software disponibles en sitios de Internet son archivos archivados y comprimidos, binarios. En el siguiente ejemplo, el modo de transferencia se configura en binario, y el paquete de software archivado **misdatos.tar.gz** se envía del sistema remoto al local mediante el comando **get**:

```
ftp> binary
ftp> get misdatos.tar.gz
PORT command successful.
Binary data connection
Binary Transfer complete.
ftp>
```

Es probable que a menudo quiera enviar varios archivos, al especificar sus nombres con caracteres comodines. Sin embargo, los comandos **put** y **get** sólo operan en un archivo único y no funcionan con caracteres especiales. Para transferir varios archivos simultáneamente, debe usar dos comandos más, **mput** y **mget**. Cuando utiliza **mput** y **mget**, se le pide una lista de archivos. Entonces puede ingresar la lista o especificación de lista de archivo mediante caracteres especiales. Por ejemplo, ***.c** especifica todos los archivos con extensión **.c**, y ***** especifica todos los archivos en el directorio actual. En el caso de **mget**, los archivos se envían uno por uno del sistema remoto a su sistema local. Cada vez, se le pide confirme el nombre del archivo que se está enviando. Puede escribir **y** para enviar el archivo o **n** para cancelar la transmisión. Después se le pedirá lo mismo para el siguiente archivo. El comando **mput** trabaja de la misma forma, pero envía archivos de su sistema local al remoto. En el siguiente ejemplo, todos los archivos con extensión **.c** se envían a su sistema local mediante **mget**:

```
ftp> mget
(archivos-remotos) *.c
mget calc.c? y
PORT command successful
ASCII data connection
ASCII transfer complete
mget principal.c? y
PORT command successful
ASCII data connection
ASCII transfer complete
ftp>
```

Contestar las confirmaciones de cada archivo puede ser una tarea tediosa si planea descargar gran número de archivos, como los de un sistema de actualización. En este caso, puede desactivar la confirmación con el comando **prompt**, que activa y desactiva el modo interactivo. Después, la operación **mget** descargará todos los archivos que encuentre, uno tras otro.

```
ftp> prompt
Interactive mode off.
ftp> mget
(remote-files) *.c
PORT command successful
ASCII data connection
ASCII transfer complete
```



```
PORT command successful
ASCII data connection
ASCII transfer complete
ftp>
```

NOTA Para acceder a un sitio FTP público, iniciar sesión en forma anónima. En vez de un nombre de inicio de sesión, inserte la palabra clave *anonymous* (o *ftp*). Después, para la contraseña, inserte su dirección de correo electrónico. Una vez se despliegue el indicador de comandos *ftp*, está listo para transferir archivos. Tal vez necesite cambiar al directorio apropiado primero o configurar el modo de transferencia a binario.

Inicio de sesión automático y macros: *.netrc*

El cliente *ftp* tiene capacidad de inicio de sesión automática y soporte para macros. Ambas se insertan en un archivo de configuración *ftp* del usuario llamado *.netrc*. Cada vez que se conecta a un sitio, se revisa el archivo *.netrc* en busca de información de conexión, como un nombre de inicio de sesión y contraseña. De esta forma, no necesita insertar un nombre de inicio de sesión ni contraseña cada vez que se conecta a un sitio. Esta característica es muy útil en el caso de inicios de sesión anónimos. En vez de insertar el nombre de usuario *anonymous* y su dirección de correo electrónico como contraseña, esta información se lee automáticamente del archivo *.netrc*. Incluso puede crear información de inicio de sesión anónima como predeterminada para que, en caso de indicar lo contrario, se intente un inicio de sesión anónimo con cualquier sitio FTP al que trate de conectarse. Si tiene sitios a los que puede conectarse, especifíquelos en el archivo *.netrc* y, cuando se conecte, iniciará con su nombre de usuario y contraseña para ese sitio, o se le pedirá automáticamente su nombre de usuario y contraseña.

Las entradas del archivo *.netrc* tienen la siguiente sintaxis. Una entrada para un sitio comienza con el término “*machine*”, seguido por la red o dirección de Internet, y después la información de inicio de sesión y contraseña.

```
machine dirección-sistema login nombre-iniciodesesión-remoto password contraseña
```

En el siguiente ejemplo se muestra una entrada para iniciar sesión en la cuenta **daniel** en el sistema **tortuga.pista.com**:

```
machine tortuga.pista.com login daniel password legogolf
```

Para un sitio en que normalmente iniciaría sesión de manera anónima, inserte la palabra “*anonymous*”, como nombre de inicio de sesión y dirección de correo electrónico para la contraseña.

```
machine ftp.redhat.com login anonymous password daniel@tortuga.pista.com
```

En casi todos los casos, utilizará *ftp* para acceder sitios FTP anónimos. En vez de crear una entrada para cada uno, puede crear una entrada anónima predeterminada para inicios de sesión FTP. Cuando se conecte a un sitio, *ftp* buscará una entrada *machine* en el archivo *.netrc*. Si no existe una, *ftp* busca una entrada predeterminada y la usará. Una entrada predeterminada comienza con la palabra “*default*” sin dirección de red. Para inicios sesión anónimos predeterminados, inserte *anonymous* y su dirección de correo electrónico como inicio de sesión y contraseña.

```
default login anonymous password daniel@tortuga.pista.com
```


Aquí se muestra un ejemplo de archivo **.netrc** con una definición machine y una entrada predeterminada:

```
.netrc
machine golf.misjuegos.com login daniel password legogolf
default login anonymous password daniel@tortuga.pista.com
```

También puede definir macros en su archivo **.netrc**. Con una macro, puede ejecutar varias operaciones ftp al mismo tiempo empleando un nombre de macro. Las macros permanecen activas durante la conexión. Cuando cierra una conexión, las macros quedan sin definición. Aunque una macro puede definirse en su línea de comandos ftp, definir las en las entradas **.netrc** tiene más sentido. De esta forma, no necesita definir las de nuevo; se leen automáticamente desde el archivo **.netrc** y se definen solas. Coloque definiciones de macro en una entrada machine particular del archivo **.netrc** o la entrada predeterminada. Las macros definidas en entradas machine sólo permanecen definidas cuando se conecta a un sitio. Las macros en la entrada predeterminada se definen cuando se conecta con un sitio.

A continuación se presenta la sintaxis para una definición de macro. Comienza con la palabra clave **macdef**, seguida por el nombre que quiere asignar a la macro, y termina con una línea vacía. Una macro ftp puede tomar argumentos, a los que se refiere en la macro con **\$n**, donde **\$1** hace referencia al primer argumento, **\$2** al segundo, etc. Si necesita usar un carácter **\$** en una macro, debe citarlo usando una diagonal invertida, **\\$**.

```
macdef nombre-macro
comandos ftp
línea-vacía
```

lftp

El programa **lftp** es un cliente FTP mejorado con características avanzadas como la capacidad para descargar sitios espejo y ejecutar varias operaciones FTP en segundo plano, simultáneamente. **lftp** usa un conjunto de comandos similares a los del cliente ftp: se utilizan los comandos **get** y **mget** para descargar archivos, con la opción **-o** para especificar ubicaciones locales. Utilice **lcd** y **cd** para cambiar directorios locales y remotos.

Para administrar comandos en segundo plano, se usan muchos de los mismos comandos que para la shell. Un **&** colocado al final de un comando lo lleva a segundo plano y **CTRL-Z** coloca un trabajo en ejecución en segundo plano. Los comandos pueden estar agrupados entre paréntesis y unidos en segundo plano. Utilice el comando **jobs** para mostrar una lista de trabajos en segundo plano y **wait** o **fg** para mover los trabajos a primero o segundo plano. Cuando sale de **lftp**, el programa seguirá ejecutando cualquier trabajo en segundo plano. En efecto, **lftp** se convierte en un trabajo en segundo plano.

Cuando se conecta a un sitio, puede formar una cola de comandos con **queue**, configurando una lista de operaciones FTP que habrán de realizarse. Esta característica permite colocar en cola varias operaciones de descarga de un sitio. La cola puede reordenarse y eliminar entradas, si así lo desea. También puede conectar varios sitios y configurar una cita para cada uno. El comando **mirror** permite mantener una versión local de un sitio de espejo. Puede descargar todo un sitio o sólo actualizar nuevos archivos, además de eliminar archivos que ya no están presentes en el espejo.

Puede ajustar **lftp** a la medida, con opciones que se configuran en el archivo **.lftprc**. Las configuraciones para todo el sistema se colocan en el archivo **/etc/lftp.conf**. Aquí, puede configurar características como el indicador de comandos que habrá de usarse y su contraseña anónima. El directorio **.lftp** almacena archivos de soporte para historial de comandos, inicios de sesión,

marcadores y comandos de arranque. El programa `lftp` también soporta el archivo `.netrc`, y lo revisa en busca de información de inicio de sesión.

NcFTP

El programa NcFTP tiene una interfaz basada en pantalla que puede ejecutarse desde cualquier línea de comandos shell. No usa interfaz de escritorio. Para iniciar NcFTP, inserte el comando `ncftp` en la línea de comandos. Si está trabajando con un administrador de ventana, como KDE, GNOME o XFce, abra una terminal de shell y escriba el comando en el indicador de comandos. La pantalla principal de NcFTP consta de una línea de entrada en la parte inferior de la pantalla, con una línea de estado arriba de ésta. En lo que sobra de la pantalla se despliegan comandos y respuestas de sistemas remotos. Por ejemplo, cuando descarga archivos, se despliega en la línea de estado un mensaje especificando qué archivos se descargarán. NcFTP permite configurar preferencias para diferentes características, como inicio de sesión anónimo, medidores de progreso o un directorio de descarga. Inserte el comando `pref` para abrir la pantalla de preferencias. Desde ahí, seleccione y modifique las preferencias en la lista.

Para conectarse a un sitio FTP, inserte el comando `open` en la línea de entrada, seguido por dirección del sitio. La dirección puede ser IP o un nombre de dominio, como `ftp.gnome.org`. Si no proporciona una dirección, se despliega una lista de sus sitios marcados, para elegir uno de ahí. Como opción predeterminada, NcFTP intenta un inicio de sesión anónimo, usando el término “anonymous” como nombre de usuario y dirección de correo electrónico como contraseña. Cuando se conecta, la barra de estado despliega los nombre del sitio y directorio remotos a la izquierda.

Si quiere iniciar sesión en una cuenta específica de un sitio remoto, haga que se le pida nombre de usuario y contraseña usando la opción `-u` con el comando `open`. El comando `open` recuerda el último inicio de sesión realizado para un sitio específico y lo repite. Si quiere cambiar nuevamente a un inicio de sesión anónimo para un inicio de sesión de usuario, utilice la opción `-a` con el comando `open`.

Una vez conectado, inserte los comandos en la línea de entrada para realizar operaciones FTP como desplegar una lista de archivo, cambiar directorios o descargar archivos. Con el comando `ls`, se muestra una lista del contenido del directorio remoto actual. Utilice el comando `cd` para cambiar a otro directorio remoto. El comando `dir` despliega una lista detallada de archivos. Con el comando `page`, se ve el contenido de un archivo remoto, de pantalla en pantalla. Para descargar archivos, use el comando `get` y para subir archivos, utilice el comando `put`. Durante una descarga, un medidor de progreso, arriba de la barra de estado despliega cuánto se ha descargado del archivo hasta el momento. El comando `get` tiene varias características descritas con más detalle en la siguiente sección. Cuando termine, use el comando `close` para desconectarse del sitio. Utilice `open` para conectarse a otro sitio o salga del programa NcFTP con el comando `quit`. El comando `help` muestra una lista de todos los comandos NcFTP. Emplee el comando `help` seguido por el nombre de un comando, para desplegar información acerca de éste.

El comando `get` de NcFTP difiere significativamente del comando `get` del cliente FTP original. Mientras `ftp` usa dos comandos, `get` y `mget`, para realizar operaciones de descarga, NcFTP sólo requiere el comando `get`. Sin embargo, el comando `get` de NcFTP combina capacidades de `mget` y `get`, también agrega varias características nuevas. Como opción predeterminada, el comando `get` de NcFTP realiza búsqueda de nombres de archivo con comodines. Si inserta sólo parte de un nombre de archivo, el comando `get` intenta descargar todos los archivos comenzando con ese nombre. Puede desactivar el uso de comodines con la opción `-G`, en cuyo caso debe escribir nombres completos de archivos que desee.

Herramientas de red

Existe gran variedad de herramientas para realizar tareas, como obtener información acerca de otros sistemas de su red, acceder a dichos sistemas y comunicarse directamente con otros usuarios. La información de red se obtiene con utilerías como **ping**, **finger**, **traceroute** y **host**. Los clientes Talk, ICQ y RC posibilitan comunicarse directamente con otros usuarios de su red. Telnet realiza un inicio de sesión remota en una cuenta de otro sistema conectado a su red. Algunas herramientas tienen versiones correspondientes de K Desktop o GNOME. Además, su red puede hacer uso de comandos de acceso remoto a red. Son útiles para redes pequeñas y permiten acceder a sistemas remotos directamente para copiar archivos o ejecutar programas.

Información de red: ping, finger, traceroute y host

Use los comandos **ping**, **finger**, **traceroute** y **host** para conocer información de estado de sistemas y usuarios de su red. El comando **ping** se emplea para revisar si un sistema remoto está activo y en ejecución. Utilice **finger** para encontrar información acerca de otros usuarios de su red, al ver si iniciaron sesión o han recibido correo; **host** despliega información sobre direcciones de un sistema en su red, al darle un IP del sistema y direcciones de nombre de dominio; mientras **traceroute** se emplea para trazar la secuencia de redes y sistemas de computadora por los que pasó su mensaje hasta llegar a usted. En la tabla 15-1 se muestra una lista de varias herramientas de información de red.

Herramientas de red de GNOME: gnome-nettool

Para el escritorio GNOME, la utilería **gnome-nettool** ofrece una interfaz GNOME para escribir los comandos **ping** y **traceroute**, entre otras características, incluidas Finger, Whois y Lookup para citar usuarios y hosts en la red. Whois proporcionará información de nombre de un dominio en particular, de la forma en que Lookup proporcionará el nombre de dominio y direcciones IP. Acceda a **gnome-nettool**, en la entrada Herramientas de red, del menú Herramientas del sistema. También incluye herramientas de estado de red como **netstat** y **portscan**. El primer panel, Dispositivos, describe sus dispositivos de red conectados, incluida información de configuración y transmisión acerca de cada dispositivo, como dirección de hardware y bytes transmitidos. Se mostrarán las listas de direcciones IP IPv4 y IPv6 de host.

Herramienta de información de red	Descripción
ping	Detecta si un sistema está conectado a la red.
finger	Obtiene información acerca de usuarios en la red.
who	Revisa qué usuarios están en línea.
whois	Obtiene información de dominio.
host	Obtiene direcciones de red e información acerca de un host remoto.
traceroute	Rastrea la secuencia de redes de computadora y hosts por los que pasó su mensaje.
wireshark	Analizador de protocolos, examina el tráfico de red.
gnome-nettool	Interfaz GNOME para varias herramientas de red incluidas ping, finger y traceroute.
mtr y xmtr	Combina operaciones de ping y traceroute (Traceroute en el menú Herramientas del sistema).

TABLA 15-1 Herramientas de información de red

ping

El comando **ping** detecta cuando un sistema está activo y en ejecución. **ping** toma como argumento el nombre del sistema que quiere revisar. Si el sistema de su interés no está activo, **ping** envía un mensaje de vencimiento indicando que la conexión no pudo establecerse. En el siguiente ejemplo se revisa si **redhat.com** está activo y conectado a su red:

```
# ping www.redhat.com
PING www.redhat.com (209.132.177.50) 56(84) bytes of data.
64 bytes from www.redhat.com (209.132.177.50): icmp_seq=1 ttl=118 time36.7 ms
64 bytes from www.redhat.com (209.132.177.50): icmp_seq=2 ttl=118 time36.9 ms
64 bytes from www.redhat.com (209.132.177.50): icmp_seq=3 ttl=118 time37.4 ms

--- www.redhat.com ping statistics ---
4 packets transmitted, 3 received, 25% packet loss, time 3000ms
rtt min/avg/max/mdev = 36.752/37.046/37.476/0.348 ms
```

También puede utilizar **ping** con una dirección IP en vez de un nombre de dominio. Con una dirección IP, **ping** intenta detectar el sistema remoto directamente, sin pasar por un servidor de nombre de dominio para traducir el nombre de dominio a una dirección IP. Esto es útil en situaciones en que el servidor de nombre de dominio de red está inactivo y quiere revisar si un host remoto particular en su red está conectado. En el siguiente ejemplo, el usuario revisa el sitio Red Hat mediante su dirección IP:

```
# ping 209.132.177.50
PING 209.132.177.50 (209.132.177.50) 56(84) bytes of data.
64 bytes from 209.132.177.50: icmp_seq=1 ttl=118 time37.4 ms
64 bytes from 209.132.177.50: icmp_seq=2 ttl=118 time37.0 ms
64 bytes from 209.132.177.50: icmp_seq=3 ttl=118 time36.3 ms
```

```
--- 209.132.177.50 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 36.385/36.969/37.436/0.436 ms
```

NOTA Una operación **ping** fallará si el acceso **ping** es denegado por un firewall de la red.

finger y who

Utilice el comando **finger** para obtener información acerca de otros usuarios en su red y el comando **who** para ver qué usuarios están en línea actualmente en su sistema. Los comandos **who** y **w** muestran una lista de usuarios conectados, además de cuándo, hace cuánto tiempo y dónde iniciaron sesión. El comando **w** ofrece información detallada y con opciones para especificar el nivel de detalle. El comando **who** está hecho para operar en un sistema o red local; **finger** puede operar en redes grandes, incluido Internet, aunque casi todos los sistemas los bloquean por razones de seguridad.

NOTA Wireshark es un analizador de protocolo para capturar paquetes de red y desplegar información detallada acerca de éstos. Puede detectar qué tipo de información se transmite en su red, así como fuente y destino. Wireshark se utiliza principalmente para administración de servidor y red.

host

Con el comando **host**, se encuentra información de dirección de red acerca de un sistema remoto conectado a su red. Esta información suele constar de la dirección IP de un sistema, dirección de nombre de dominio, apodos de nombre de dominio y servidor de correo. Tal información se obtiene desde su servidor de nombre de dominio de la red. Para Internet, incluye todos los sistemas conectados a Internet.

El comando **host** es una forma efectiva para determinar una dirección IP o URL del sitio. Si sólo tiene la dirección IP de un sitio, puede usar **host** para saber su nombre de dominio. Para administración de red, una dirección IP puede ser útil para crear sus propias entradas de nombre de dominio en su archivo **/etc/host**. De esa forma, tal vez no necesite depender de un servidor de nombre de dominio remoto (DNS; Domain Name Server) para localizar un sitio.

```
# host gnomefiles.org
gnomefiles.org has address 67.18.254.188
gnomefiles.org mail is handled by 10 mx.zayda.net.
```

```
# host 67.18.254.188
188.254.18.67.in-addr.arpa domain name pointer gnomefiles.org.
```

traceroute

Las conexiones de Internet se hacen a través de varias rutas, viajando por una serie de host de puerta de enlace interconectados. La ruta de un sistema a otro toma diferentes vías, algunas más rápidas que otras. En el caso de una conexión lenta, use **traceroute** para revisar la ruta por que se conecta a un host, al monitorear velocidad y número de conexiones de puerta de enlace interviniendo en una ruta. El comando **traceroute** toma como argumento nombre del host o direcciones IP del sistema cuya ruta quiere revisar. Hay opciones disponibles para especificar parámetros, como tipo de servicio (**-t**) o host de origen (**-s**). El comando **traceroute** regresará

una lista de hosts por los que viaja la ruta, junto con tiempos para tres “sondas” enviadas a cada puerta de enlace. Los tiempos mayores a 5 segundos se despliegan con un asterisco (*).

```
tracert conejo.mipista.com
```

Maneje también las herramientas mtr o xmtr para usar ping y traceroute (Traceroute en el menú Herramientas del sistema).

Cientes para charla en red y mensajería: VoIP, ICQ, IRC, AIM y Talk

Quizás, ocasionalmente, quiera comunicarse directamente con otros usuarios en su red. Puede hacerlo con utilerías de voz en protocolo de Internet (VoIP, Voice over Internet Protocol), Talk, ICQ, mensajería instantánea (IM, Instant Messenger) e IRC, siempre y cuando el otro usuario también haya iniciado sesión en un sistema conectado simultáneamente (véase tabla 15-2). Con aplicaciones de VoIP, puede hablar a través de conexiones de Internet como si fuera un teléfono ordinario. La utilería Talk opera como herramienta de mensajería de texto de dos vías, permitiéndole tener una conversación de dos vías, directamente con otro usuario. Talk está diseñado para usuarios en el mismo sistema o conectados a una red local. ICQ (representación de la frase “I Seek You”, te busco) es una herramienta de Internet notificándole cuando otros usuarios están en línea y permitiéndole comunicarse con ellos. ICQ trabaja muy parecido a IM. Con la utilería charla de retransmisión en Internet (IRC, Internet Relay Chat), puede conectarse a un servidor remoto donde otros usuarios también están conectados, para charlar con ellos. Los clientes IM operan casi de la misma forma, permitiéndoles a los usuarios del mismo sistema IM comunicarse en cualquier lugar a través de Internet. Actualmente los principales sistemas IM son AOL Instant Messenger (AIM), Microsoft Network (MSN), Yahoo, ICQ y Jabber. A diferencia de otros, Jabber es un servicio IM de fuente abierta (jabber.org).

Ekiga

Ekiga es una nueva aplicación VoIP de GNOME proporcionando soporte teléfono IP en Internet y conferencias de video. Inicialmente fue llamada GnomeMeeting y su sitio Web todavía es gnomemeeting.org. Ekiga soporta los protocolos H.323 y SIP (Session Initiation Protocol, protocolo de inicio de sesión). Es compatible con NetMeeting de Microsoft. H.323q3, es un protocolo muy extenso incluyendo transmisión digital, como DVB y H.261 para flujo de video, además de protocolos de soporte como la serie H.450 para administrar llamadas.

Cliente	Descripción
Ekiga	Aplicación de VoIP
GnomeICU	Cliente ICQ de GNOME
X-Chat	Cliente IRC
Konversation	Cliente IRC de KDE
Gabber	Cliente Jabber
Gaim	Cliente AIM de GNOME
psi	Cliente Jabber que utiliza QT (KDE)
nalm	Cliente de línea de comandos IRC, ICQ y AIM basado en cursor

TABLA 15-2 Clientes para charla y mensajería

Para usar Ekiga necesita una dirección SIP. Obtenga una gratis en **ekiga.net**, pero primero tendrá que suscribirse al servicio. Cuando inicie Ekiga, se le pedirá configurar su conexión. Debe proporcionar información como datos de contacto, método de conexión, controlador de sonido y dispositivo de video. Use la libreta de direcciones para conectarse con otro usuario de Ekiga. Un directorio de páginas blancas le permite buscar personas usando también Ekiga.

ICQ e IRC

El protocolo ICQ favorece la comunicación directa con otros usuarios en línea, como una utilería IM. Al utilizar un cliente ICQ, puede enviar mensajes a usuarios, platicar con ellos o enviar archivos. Configure una lista de contactos para usuarios con los que quiera ponerse en contacto cuando estén en línea. Se le notificará en tiempo real cuando se conecten y podrá comunicarse con ellos, si lo desea. Se da soporte a varios modos de comunicación. Entre estos se incluye charla, mensaje, correo electrónico, transferencia de archivos y juegos. Para utilizar ICQ, regístrese con un servidor ICQ ofreciéndole un número de ICQ, también conocido como número universal de Internet (UIN, Universal Internet Number). Aprenderá más acerca del protocolo ICQ en **icq.com**.

IRC opera como cuarto de charla, donde puede entrar a canales y hablar con otros usuarios que ya están ahí. Primero, elija un servidor IRC adonde conectarse. Están disponibles varios servidores disponibles para diferentes localidades y temas. Una vez conectado a un servidor, seleccione desde la lista de canales para entrar. La interfaz trabaja de forma muy parecida a un cuarto de charla. Cuando se conecta a un servidor, seleccione un apodo por el que se dará a conocer. Hay varios clientes IRC disponibles para ser usados en sus sistemas Linux. Casi todos operan en plataformas X Window System, KDE o GNOME.

Hay varios clientes ICQ e IRC basados en GNOME y KDE disponibles. Revise la lista de software de GNOME en **gnomefiles.org** para conocer nuevas versiones y actualizaciones recientes. En el caso de clientes ICQ basados en KDE, consulte **kde-apps.org** (Network | Chat).

Mensajería instantánea

AOL Instant Messenger (AIM) es un servicio gratuito proporcionado por AOL para quién se registre en éste, además de los que ya son miembros de AOL. Con AIM, puede enviar mensajes a miembros de manera instantánea, participar en juegos con ellos y recibir alertas de acciones. Incluso compartir imágenes, sonidos y fotografías. AOL ya ofrece clientes para Windows y Macintosh. Una nueva versión llamada AIM Express está diseñada para ejecutarse en cualquier explorador Web y en sistemas con JDK 1.1 o mayor. Conozca más acerca de AIM en **aim.com**.

Varios clientes de mensajería instantánea de GNOME están diseñados para trabajar con todos los sistemas de mensajería instalados, incluidos AIM, Yahoo, MSN e ICQ. Gaim tiene un plug-in que permite conectarse a ICQ, Yahoo, MSN, IRC, Jabber y Zephyr. Gabber, un cliente Jabber, es un sistema de mensajería instantánea de fuente abierta para comunicaciones con otros sistemas, incluidos AIM, Yahoo, MSN e ICQ.

NOTA *Talk es una utilería de plática original de Unix, para configurar una comunicación interactiva de dos vías, entre usted y otro usuario usando una interfaz de línea de comandos. Funciona de manera muy similar a la mensajería instantánea. Debido a cuestiones de seguridad, sólo debe utilizar Talk en un sistema seguro de manera local. Una versión de Talk para K Desktop, llamada KTalk, despliega paneles y pantallas en una ventana de K Desktop. GNU Talk es una versión de Talk de GNOME soportando varios clientes, transferencia de archivos, codificación, aplicaciones compartidas, respuestas automáticas y reenvío de llamadas.*

Telnet

El comando **telnet** se usa para iniciar sesión de manera remota en otro sistema de su red. El sistema puede estar en su red de área local o disponible a través de una conexión a Internet. Telnet opera como si iniciara sesión en otro sistema desde una terminal remota. Se pedirá un nombre de inicio de sesión y, en algunos casos, contraseña. En efecto, está iniciando sesión en otra cuenta de otro sistema. En realidad, si tiene cuenta en otro sistema, puede utilizar Telnet para iniciar sesión en ésta.

PRECAUCIÓN *Se sabe que la versión original de Telnet es muy insegura. Para conexiones seguras en una red o Internet, debe utilizar las versiones de Secure Shell (SSH, shell segura) o Kerberos de Telnet. Operan de la misma forma que el original pero mediante autenticación y cifrado para asegurar la conexión Telnet. Aún así, es aconsejable que nunca use Telnet para iniciar en su cuenta root.*

La utilidad Telnet se invoca con la palabra clave **telnet**. Si conoce el nombre del sitio al que quiere conectarse, escriba **telnet** y el nombre del sitio en línea de comandos de Linux. Como opción, emplee la utilidad KTelnet de K Desktop. Ésta ofrece una interfaz GUI para conectarse e iniciar sesión en un sistema remoto.

```
$ telnet purpucone.berkeley.edu
Connected to purpucone
login:
```

El programa Telnet también tiene modalidad de comandos, en la que utiliza una serie de comandos para configurar su conexión. Entre al modo de comandos de **telnet**, invocando a éste con la palabra clave **telnet** u oprimiendo CTRL-] durante una sesión. El comando **help** de Telnet muestra una lista de todos los comandos de Telnet disponibles. Una lista amplia está disponible en las páginas Man (**man telnet**). En el siguiente ejemplo, el usuario invoca primero la utilidad Telnet. Tras desplegar un indicador de comandos, mostrando el modo de comandos, **telnet>**. El comando **open** de Telnet después se conecta a otro sistema.

```
$ telnet
telnet> open purpucone.berkeley.edu
Connected to purpucone.berkeley.edu
login:
```

Una vez conectado, siga el procedimiento de inicio de sesión para ese sistema. Si inicia sesión en un sistema regular, debe proporcionar el nombre de inicio de sesión y contraseña. Una vez iniciada la sesión, se le proporciona un indicador de comandos del sistema operativo; en el caso de Linux o Unix, será **\$** o **%**. Entonces estará conectado directamente a una cuenta de ese sistema y podrá enviar cualquier comando que quiera. Cuando termine su trabajo, salga de su sesión. Esto rompe la conexión y devuelve al indicador de comandos Telnet en su propio sistema. Después puede salir de Telnet con el comando **quit**.

```
telnet> quit
```

Cuando use Telnet para conectarse a un sitio ofreciendo acceso público, no necesita proporcionar un nombre de usuario o contraseña. El acceso suele controlarse con una serie de menús restringiendo lo que puede hacer en el sistema. Si inició sesión en una cuenta específica de otro sistema, puede usar la opción **-l** para especificar el nombre de inicio de sesión de esa cuenta.

Comandos de acceso remoto RSH, Kerberos y SSH

Los comandos de acceso remoto fueron diseñados para redes pequeñas, como intranets. Permiten iniciar sesión de manera remota en otra cuenta de otro sistema y copiar archivos de un sistema a otro. También puede adquirir información de otros sistemas, como quién tiene una sesión iniciada actualmente (véase la tabla 15-3). Muchos de los comandos remotos tienen utilerías de comunicación de red equiparables, utilizados para Internet. Por ejemplo, **rlogin**, para iniciar sesión de manera remota en un sistema, similar a **telnet**. El comando **rcp**, que copia archivos de manera remota, realiza casi las mismas funciones que **ftp**.

Debido a riesgos de seguridad con las versiones originales de las operaciones remotas **rcp**, **rlogin** y **rsh** (paquete RSH), las implementaciones de seguridad ahora se instalan con casi todas las distribuciones de Linux. Kerberos y SSH ofrecen las versiones seguras de estos comandos. Las versiones de Kerberos se configuran como predeterminadas. En cualquier momento que inserte un comando **rcp** o **rsh**, realmente estará invocando la versión de Kerberos del comando. Kerberos brinda versiones para Telnet, **rlogin**, **rcp**, **rsh** y **ftp**, ofreciendo autenticación y cifrado. Estas versiones operan al usar los mismos comandos y opciones que los originales, haciendo su uso transparente para el usuario. En algunas distribuciones, como Fedora, cuando Kerberos se instala en su sistema, la variable **PATH** se configura para acceso a versiones de los comandos remotos de Kerberos, localizados en **/usr/kerberos/bin**, en vez de **/usr/bin**, haciendo que las versiones de Kerberos sean las predeterminadas.

Las versiones de SSH utilizan nombres ligeramente distintos, al emplear una **s** al principio de los comandos, como **ssh**, **slogin** o **scp**. Los comandos de SSH están cifrados, lo que proporciona un nivel elevado de seguridad.

Comandos remotos	Efecto
rwho	Muestra todos los usuarios que iniciaron sesión en el sistema en su red.
ruptime	Despliega información acerca de cada sistema de su red.
rlogin nombre-sistema	Permite iniciar sesión de manera remota en una cuenta de otro sistema. La versión Kerberos se utiliza de forma predeterminada. La opción -l permite especificar el nombre de inicio de sesión de la cuenta.
slogin nombre-sistema	Inicio de sesión seguro en una cuenta de otro sistema.
rcp nombre-sys::archivo1 nombre-sys::archivo2	Permite copiar un archivo de una cuenta en un sistema a otra en otro sistema. Con la opción -p , se preservan los tiempos y los modos de modificación de los archivos fuente. La versión de Kerberos se usa de forma predeterminada.
scp nombre-sys::archivo1 nombre-sys::archivo2	Copia un archivo de una cuenta en un sistema a otra en otro sistema.
rsh nombre-sys comando-Linux	Le permite ejecutar un comando de manera remota en otro sistema. La opción -l permite especificar el nombre de inicio de sesión; -n redirige la entrada desde el dispositivo especial nulo, /dev/null . La versión de Kerberos se usa de forma predeterminada.
scp nombre-sys comando-Linux	Ejecuta de manera segura y remota un comando en otro sistema.

TABLA 15-3 Comandos de acceso remoto

Incluso los comandos remotos originales incluyen ahora soporte a Kerberos, lo que permite usar configuraciones de acceso más seguras, como las proporcionadas por **.k5login**. Aún así, estos comandos permiten un acceso remoto fácil, sin cifrado, a un sistema Linux. Sólo deben utilizarse en una red segura local.

Información de acceso remoto

Se utilizan varios comandos para obtener información de diferentes sistemas en su red. Conozca quién ha iniciado sesión, obtenga información acerca de un usuario en otro sistema o encuentre si un sistema está activo y en ejecución. Por ejemplo, el comando **rwho** funciona de la misma forma que **who**. Despliega todos los usuarios que han iniciado sesión en cada sistema de su red

```
$ rwho
violeta roberto:tty1 Sept 10 10:34
gonzalo carlos:tty2 Sept 10 09:22
```

El comando **ruptime** despliega información de cada sistema de su red. La información muestra cómo se ha desempeñado cada sistema: **ruptime** muestra cuando un sistema está en funcionamiento o no; cuánto tiempo ha estado en funcionamiento o inactivo; número de usuarios en el sistema y la carga en promedio en el sistema durante los últimos cinco, diez y quince minutos.

```
$ ruptime
violeta up 11+04:10, 8 users, load 1.20 1.10 1.00
gonzalo up 11+04:10, 20 users, load 1.50 1.40 1.30
```

Permisos de acceso remoto: **.k5login**

En muchas distribuciones, los comandos remotos están habilitados para Kerberos, que permite usar la autenticación Kerberos para controlar el acceso. Para mayor facilidad, use el archivo **.k5login** para controlar el acceso a su cuenta por parte de usuarios empleando comandos remotos (no se utiliza **.rhosts**). Los usuarios crean este archivo en sus propias cuentas mediante un editor estándar. Deben estar ubicados en el directorio home del usuario.

El archivo **.k5login** es una forma sencilla de permitir a otras personas acceder a su cuenta sin dar su contraseña. Para denegar el acceso a un usuario, sólo elimine los nombres del sistema e inicio de sesión del usuario en su archivo **.k5login**. Si un nombre de inicio de sesión del usuario y de sistema están en un archivo **.k5login**, el usuario puede acceder directamente a esa cuenta sin conocer la contraseña (en lugar de utilizar **.k5login**, puede utilizar una contraseña). El archivo **.k5login** contendrá nombres para usuarios de Kerberos, incluidos nombres de usuario y dominio. Dicho usuario se someterá a la autenticación de Kerberos para acceder. Es necesario un archivo **.k5login** para otros comandos remotos, como copiar archivos o ejecutar comandos de Linux de manera remota.

El tipo de acceso que proporciona **.k5login** establece que pueda utilizar comandos remotos para acceder directamente a cuentas que tenga en otros sistemas. No necesita iniciar sesión en éstas, primero. En efecto, puede tratar a sus cuentas en otros sistemas como extensiones del sistema en que ha iniciado sesión. Con el comando **rcp**, puede copiar cualquier archivo de un directorio a otro, sin importar en qué cuenta estén. Con el comando **rsh**, puede ejecutar cualquier comando de Linux en cualquier otra cuenta.

rlogin, slogin, rcp, scp, rsh, y ssh

Tal vez tenga cuentas en diferentes sistemas de su red, o quizás tenga permitido el acceso a la cuenta de alguien más en otro sistema. Para acceder a una cuenta en otro sistema, primero debe iniciar sesión en su propio sistema y después, de manera remota, a través de su red, en la cuenta del otro sistema. Este inicio de sesión remoto se realiza usando el comando **rlogin**, tomando como argumento un nombre de sistema. El comando lo conecta al otro sistema y comienza los procedimientos de inicio de sesión. Considere que si utiliza una conexión de red con SSH habilitado, puede recurrir al comando **slogin** en vez de **rlogin**. Tanto **slogin** como **rlogin** de Kerberos, concederán acceso de inicio de sesión con cifrado seguro.

Utilice el comando **rcp** para copiar archivos entre sistemas remotos y locales. En el caso de conexiones de red con SSH habilitado, utilizaría **scp** en vez de **rcp**. Los comandos **scp** y **rcp** son herramientas de transferencia de archivos que operan como el comando **cp** pero a través de una conexión de red a un sistema remoto. El comando **rcp** comienza con la palabra clave **rcp**, cuyos argumentos son los nombres del archivo de origen y la copia. Para especificar el archivo en el sistema remoto, debe colocar el nombre del sistema remoto antes del nombre del archivo, separado por dos puntos. Cuando copia un archivo del sistema remoto a su propio sistema, el archivo de origen es remoto y requiere el nombre del sistema remoto. La copia es un archivo en su propio sistema y no requiere el nombre de éste:

```
$ rcp nombre-sistema-remoto:archivo-original archivo-copia
```

En el siguiente ejemplo, el usuario copia el archivo **martes** del sistema remoto **violeta** a su propio sistema y cambia su nombre por **hoy**:

```
$ rcp violeta:martes hoy
```

También puede usar los comandos **scp** o **rcp** para copiar todos los directorios a un sistema remoto o desde éste. El comando **scp** con la opción **-r** copia un directorio y todos los subdirectorios de un sistema a otro. Como el comando **cp**, estos comandos requieren directorios fuente y destino. El directorio en el sistema remoto requiere el nombre del sistema y dos puntos antes del nombre del directorio. Cuando copia un directorio de su propio sistema a uno remoto, la copia del directorio está en el sistema remoto y necesita el nombre del sistema remoto. En el siguiente ejemplo, el usuario emplea el comando **scp** para copiar el directorio **cartas**, al directorio **cartasviejas** en el sistema remoto **violeta**:

```
$ scp -r cartas violeta:cartasviejas
```

NOTA En el caso de copias de seguridad o de copias de una gran cantidad de archivos, utilice **rsync**.

En ocasiones, tal vez necesite ejecutar un solo comando en un sistema remoto. El comando **rsh** ejecuta un comando de Linux en otro sistema y despliega los resultados en el suyo. Su nombre de sistema y de inicio de sesión deben, por supuesto, estar en el archivo **.k5login** del sistema remoto. Para conexiones de red con SSH habilitado, puede usar **ssh** en vez de **rsh**. Los comandos **ssh** y **rsh** toman dos argumentos generales: un nombre de sistema y un comando Linux. La sintaxis es la siguiente:

```
$ rsh nombre-sistema-remoto comando-Linux
```

En el siguiente ejemplo, el comando **rsh** ejecuta un comando **ls** en el sistema remoto **violeta** para desplegar una lista de archivos en el directorio **/home/roberto de violeta**:

```
$ rsh violeta ls /home/roberto
```

El sistema local evalúa los caracteres especiales, a menos que estén entre comillas, en cuyo caso el carácter especial se vuelve parte del comando de Linux evaluado en el sistema remoto. La opción para incluir entre comillas operadores de redireccionamiento, permite realizar operaciones de redireccionamiento en el sistema remoto. En el siguiente ejemplo, el operador de redireccionamiento está entre comillas. Se vuelve parte de un comando Linux, incluido su argumento: el nombre de archivo **misarchivos**. Entonces, el comando **ls** genera una lista de nombres de archivo que se dirige en el sistema remoto a un archivo llamado **misarchivos**, también localizado en el sistema remoto.

```
$ ssh violet ls /home/roberto '>' misarchivos
```

Lo mismo es cierto para el caso de barras verticales. El primer comando (mostrado más adelante) imprime la lista de archivos en la impresora del sistema local. La salida estándar se canaliza a su propia impresora en línea. En el segundo comando, la lista de archivos se imprime en la impresora del sistema remoto. La barra vertical se incluye entre comillas y es evaluada por el sistema remoto, canalizando la salida a la impresora de éste.

```
$ ssh violeta ls /home/roberto | lpr
$ ssh violeta ls /home/roberto '|' lpr
```

NOTA La versión Kerberos de comandos remotos también permite especificar dominios y credenciales de Kerberos.

V

PARTE

Seguridad

CAPÍTULO 16

Cifrado, verificaciones de integridad y firmas

CAPÍTULO 17

Linux con seguridad mejorada

CAPÍTULO 18

IPsec y redes privadas virtuales

CAPÍTULO 19

Secure Shell y Kerberos

CAPÍTULO 20

Firewalls

Cifrado, verificaciones de integridad y firmas

Cifrado, verificación de integridad y firmas digitales se utilizan para proteger sus datos en una red. Por ejemplo, el paquete de cifrado GNU Privacy Guard permite cifrar sus mensajes de correo electrónico o archivos que quiere enviar, al igual que incluir una firma digital cifrada, autenticando que usted envió el mensaje. La firma digital también incluye información cifrada de compendio de modificaciones proporcionando una verificación de integridad, permitiendo así al destinatario verificar que el mensaje recibido es original y no uno que fue modificado o sustituido.

Este tipo de cifrado fue implementado originalmente con Pretty Good Privacy (PGP, muy buena privacidad). Originalmente, una metodología controlada de manera privada fue pasada a la Internet Engineering Task Force (IETF) para soportar un estándar abierto para PGP llamado OpenPGP (véase la tabla 16-1). Cualquier proyecto puede usar OpenPGP para crear aplicaciones de cifrado, como GnuPGP. PGP Corporation, que también utiliza el estándar OpenPGP, aún desarrolla productos comerciales para PGP.

Cifrado de clave pública, verificaciones de integridad y firmas digitales

Cifrar datos es la única forma segura de asegurar los que se transmiten por red. Los datos se cifran con una clave y el destinatario o destinatarios después pueden descifrarlos. Para proteger de manera completa datos transmitidos en red, no sólo debe cifrarlos, sino asegurarse de que no se han modificado, además de confirmar que realmente fueron creados por el autor. Es posible interceptar y modificar un mensaje cifrado y después cifrarlo de nuevo. Verificaciones de integridad como el compendio de modificaciones aseguran que los datos no se hayan alterado. Aunque cifrado y verificaciones de integridad protegen datos, no los autentican. También necesita saber que quien dice haber enviado el mensaje realmente es quien lo envió, en lugar de ser un impostor. Para autenticar un mensaje, el autor puede usar una firma digital. Esta también se cifra, permitiendo al destinatario validarla. Las firmas digitales aseguran que el mensaje recibido es auténtico.

Sitio Web	Descripción
gnupg.org	GnuPGP, GPG
openpgp.org	Estándar abierto IETF para PGP
pgp.com	PGP Corporation, productos comerciales de PGP

TABLA 16-1 Sitios PGP

Cifrado de clave pública

El cifrado usa una clave para cifrar datos de forma tal que una clave correspondiente permite descifrarlos. En el pasado, formas antiguas de cifrado usaban la misma clave para cifrar y descifrar un mensaje. Sin embargo, para esto era necesario proporcionar un destinatario con la clave, abriendo la posibilidad de que cualquiera con una clave pudiese descifrar los datos. El cifrado de clave pública utiliza dos claves para cifrar y descifrar un mensaje, una clave privada y una pública. La clave *privada* siempre se conserva y usa para descifrar los mensajes recibidos. La clave *pública* está disponible para todos los destinatarios de un mensaje. Ellos entonces utilizan esta clave pública para cifrar cualquier mensaje que quieran enviar. La clave privada descifra mensajes y la clave pública los cifra. Cada usuario tiene un conjunto de claves privadas y públicas. De manera recíproca, si quiere enviar mensajes a otro usuario, primero debe adquirir la clave pública del usuario y manejarla para cifrar un mensaje que quiere enviarle. Después, el usuario descifra el mensaje con su propia clave privada. En otras palabras, los demás utilizan su clave pública para cifrar mensajes recibidos por usted y, a su vez, usted emplea la clave pública del usuario para cifrar mensajes que envía. Todos los usuarios de su sistema Linux pueden tener sus claves propias públicas o privadas. Utilizarán el programa **gpg** para generarlas y mantener su clave privada en directorios propios.

Las firmas digitales

Una *firma digital* se emplea para autenticar un mensaje y verificar la integridad. La autenticación garantiza que el mensaje no se ha modificado (es el mensaje original enviado por usted) y las verificaciones de integridad comprueban que no ha cambiado. Aunque suele combinarse con mensajes cifrados para proporcionar mayor nivel de seguridad, las firmas digitales también se usan para mensajes que pueden enviarse al mundo exterior. Por ejemplo, tal vez quiera saber si una noticia pública de actualizaciones de Red Hat fue enviada por Red Hat y no alguien más tratando de crear confusión. Dicho mensajes todavía necesita autenticarse y revisarse para ver si realmente fue enviado por el emisor o, en caso de que lo haya enviado el emisor original, no fue cambiado de alguna forma en la ruta. Verificaciones como estas protegen contra modificaciones o sustituciones del mensaje, por parte de alguien pretendiendo ser el emisor.

Verificaciones de integridad

Para firmar de manera digital un mensaje se requiere generar un valor de suma de verificación a partir del contenido de un mensaje a través de un algoritmo de hash de cifrado, como el algoritmo de compendio de modificación SHA2. Se trata de un valor único que representando de manera acertada tamaño y contenido de su mensaje. Cualquier cambio al mensaje de cualquier tipo generará un valor diferente. Dicho valor ofrece una forma de revisar la integridad de los datos. Al

valor suele conocerse como MD5, y refleja el algoritmo de hash MD5, usado para cifrar el valor. El algoritmo MD5 ha sido remplazado por los algoritmos SHA2 más seguros.

Entonces se cifra el valor de MD5 con su clave privada. Cuando el usuario recibe su mensaje, descifra la firma digital con su clave pública. Luego el usuario genera un valor de MD5 del mensaje recibido y lo compara con el enviado por MD5. Si son iguales, el mensaje se autentifica (es el mensaje original enviado por usted, no uno falso enviado por un usuario que pretende ser usted). El usuario utiliza GnuPG (descrito en la sección “GNU Privacy Guard”) para descifrar y revisar firmas digitales.

Combinación de cifrado y firmas

En general, las firmas digitales se combinan con cifrado para proporcionar nivel mayor de seguridad de transmisión. El mensaje se cifra con la clave pública del destinatario y la firma digital cifra su clave privada. El usuario descifra ambos, el mensaje (con su propia clave privada) y después la firma (con su clave pública). Luego compara la firma con que generó el usuario del mensaje para autenticarlo. Cuando GnuPG descifra un mensaje, también descifra y revisa automáticamente una firma digital. En la Figura 16-1 se muestra el proceso de cifrado y firma digital de un mensaje.

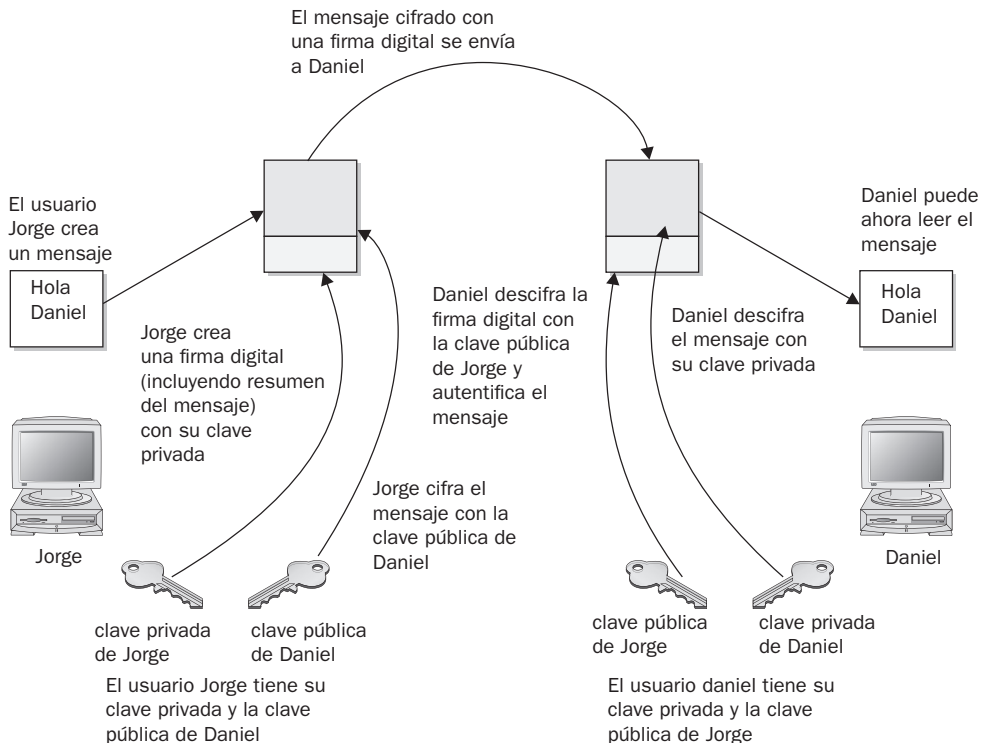


FIGURA 16-1 Cifrado de clave pública y firmas digitales

GNU Privacy Guard

Para proteger mensajes enviados por correo electrónico, casi todas las distribuciones de Linux proporcionan cifrado y autenticación de GNU Privacy Guard (GnuPG, en gnu.org). GnuPG es un software de fuente abierta de GNU, funcionando casi igual que el cifrado de PGP. Es la herramienta de cifrado y firma OpenPGP (OpenPGP es una versión de fuente abierta de PGP). Con GnuPG, se cifran y firman sus mensajes (protegiendo el mensaje y autenticando que sean de usted).

Actualmente, Evolution y Kmail soportan cifrado y autenticación de GnuPG, junto con Thunderbird con extensiones GPG agregadas. En Evolution, seleccione cifrado y firma de PGP, en el menú Seguridad para utilizar GnuPG (las opciones de PGP utilizan GnuPG). En Kmail, seleccione el cifrado a ser usado en el panel Seguridad, de la ventana Opciones. Para Thunderbird, use la extensión enigmail para soportar cifrado OpenPGP y PGP (enigmail.mozdev.org).

Las operaciones de GnuPG se realizan con el comando **gpg**, empleando ambos comandos y opciones para realizar tareas. Comandos y opciones de uso común se muestran en una lista en la tabla 16-2. Algunos comandos y opciones tienen forma corta que sólo utiliza un guión.

Generalmente, se usan dos guiones. Si sólo quiere verificar la validez de la firma digital, utilice en cambio **gpgv**. Es una versión reducida de **gpg**, ocupado para verificación de firmas.

Comandos de GPG	Descripción
-s, --sign	Firma un documento, creando una firma. Puede combinarse con --encrypt .
--clearsign	Crea una firma de texto simple.
-b, --detach-sign	Crea una firma individual.
-e, --encrypt	Cifra datos. Puede combinarse con --sign .
--decrypt [archivo]	Descifra al <i>archivo</i> (o stdin si no se especifica un archivo) y lo escribe en stdout (o el archivo especificado con --output). If the decrypted file is signed, the signature is verified.
--verify [[archivofirmado] [archivos-firmados]]	Verifica un archivo firmado. La firma puede estar contenida en el archivo o un archivo de firma individual.
--list-keys [nombres]	Muestra una lista de todas las claves del anillo de firmas o las que se especifican.
--list-public-keys [nombres]	Muestra una lista de todas las claves de los anillos públicos o las especificadas.
--list-secret-keys [nombres]	Muestra una lista de sus claves privadas (secretas).
--list-sigs [nombres]	Muestra una lista de sus claves junto con cualquier firma que tengan.
--check-sigs [nombres]	Muestra una lista de las claves y sus firmas y verifica las firmas.
--fingerprint [nombres]	Muestra una lista de huellas de claves específicas.
--gen-key	Genera un nuevo conjunto de claves privadas y públicas.

TABLA 16-2 Comandos y opciones de GPG

Comandos de GPG	Descripción
<code>--edit-key nombre</code>	Edita sus claves. Los comandos realizan la mayor parte de las operaciones de clave, como sign , para firmar una clave y passwd para cambiar su frase de contraseña.
<code>--sign-key nombre</code>	Firma una clave pública con su clave privada. Igual a sign en --edit-key .
<code>--delete-key nombre</code>	Elimina una clave pública desde el anillo público.
<code>--delete-secret-key nombre</code>	Elimina claves privadas y públicas de los anillos público y privado.
<code>--gen-revoke</code>	Genera un certificado de revocación para su propia clave.
<code>--export [nombres]</code>	Exporta una clave específica desde su anillo de claves. Sin argumentos, exporta todas las claves.
<code>--send-keys [nombres]</code>	Exporta y envía claves específicas al servidor de claves. La opción --keyserver debe utilizarse para asignar un nombre a este servidor de claves.
<code>--import [archivos]</code>	Importa claves contenidas en archivos a su anillo público.
Opciones GPG	Descripción
<code>-a, --armor</code>	Crea una salida ASCII blindada, la versión ASCII de datos cifrados.
<code>-o, --output archivo</code>	Escribe la salida en un archivo específico.
<code>--default-key nombre</code>	Especifica una clave privada predeterminada para utilizar con firmas.
<code>--keyserver sitio</code>	Busca claves públicas que no están en su anillo. También puede especificar el sitio a donde se enviará la clave pública. host -l pgp.net grep www.keys mostrará una lista de servidores de claves.
<code>-r, --recipient nombres</code>	Cifra datos para el usuario especificado, al utilizar la clave pública del usuario.
<code>--default-recipient nombres</code>	Especifica el recipiente predeterminado que se empleará para cifrado de datos.

TABLA 16-2 Comandos y opciones de GPG (continuación)

La primera vez que recurra a **gpg**, se creará un directorio **.gnupg** en su directorio home con un archivo llamado **options**. El archivo **.gnupg/gpg.conf** contiene opciones predeterminadas comentadas para operaciones de GPG. Edite este archivo y quite la marca de comentario o cambie cualquier opción predeterminada que quiera implementar para GPG. Utilice un archivo de opciones diferente, especificándolo con el parámetro **--options** cuando invoca **gpg**. Entre las opciones útiles se incluyen entradas de servidor de claves. El directorio **.gnupg** también almacena archivos de cifrado como **secring.gpg** para claves secretas (anillo secreto), **pubring.gpg** para sus claves públicas (anillo público) y **trustdb.gpg**, una base de datos para claves confiables.

SUGERENCIA Utilice el GNOME Keyring Manager (**gnome-anillo**) para administrar sus claves secretas PGP.

GnuPG Setup: gpg

Antes de utilizar GnuPG, necesitará generar sus claves privadas y públicas. En la línea de comandos (la ventana de terminal), inserte el comando **gpg** con **--gen-key**. El programa **gpg** enviará entonces diferentes opciones para crear sus claves públicas y privadas. Consulte también la página Man **gpg** para conocer más información al utilizar el programa **gpg**.

```
gpg --gen-key
```

Creación de su clave

Primero se le pide seleccionar el tipo de clave que desea. Usualmente, sólo selecciona la entrada predeterminada, que hace al oprimir ENTER. Después seleccione el tamaño de clave, que suele ser la opción predeterminada, 1024. Entonces especifique cuánto tiempo será válida la clave (por lo general, no tiene vencimiento). Se le pedirá que escriba ID de usuario, comentario y dirección de correo electrónico. Oprima ENTER para que se le pida confirmación con cada uno. Estos elementos, de los cuales puede usar cualquiera como nombre de clave, identifican la clave. Utilice el nombre de la clave cuando realice ciertas tareas GPG, como firmar una clave o crear un certificado de revocación. Por ejemplo, los siguientes elementos crean una clave para el usuario ricardolp con el comentario “autor” y la dirección de correo electrónico jorgeap@tortuga.mytrek.com:

```
Ricardo Prado (autor) ricardolp@tortuga.mytrek.com
```

Puede utilizar cualquier parte única de una identidad de la clave para hacer referencia a esa clave. Por ejemplo, la cadena “Ricardo” hará referencia a la clave anterior, siempre y cuando no existan otras claves incluyendo la cadena “Ricardo”. La cadena “ricardolp” también hará referencia a la clave, además de “autor”. Cuando una cadena coincide con más de una clave, se hará referencia a todas ellas.

Protección de su clave

El programa **gpg** después pedirá ingrese una frase de contraseña, utilizada para proteger su clave privada. Asegúrese de utilizar una frase real, incluyendo espacios, no sólo una contraseña. **gpg** genera entonces sus claves públicas y privadas, que coloca en el directorio **.gnupg**. Las claves privadas se mantienen en un archivo llamado **secring.gpg** en su directorio **.gnupg**. La clave pública se coloca en el archivo **pubring.gpg**, al que puede agregar claves públicas para otros usuarios. Muestre una lista de estas claves con el comando **--list-keys**.

En caso de que después necesite cambiar sus claves, cree un certificado de revocación para notificar a otros que la clave pública ya no es válida. Por ejemplo, si olvida su contraseña o alguien la descubre, utilice el certificado de revocación para decirles a otros que ya no debe utilizarse su clave pública. En el siguiente ejemplo, el usuario crea un certificado de revocación para la clave ricardolp y lo coloca en el archivo **mirevocación.asc**:

```
gpg --output mirevocación.asc --gen-revoke ricardolp
```

Puesta a disposición de su clave pública

Para que otros usuarios descifren sus mensajes, tiene que hacer que su clave pública esté disponible para ellos, quienes, a cambio, deberán enviarle sus claves públicas para descifrar cualquiera de los mensajes recibidos de ellos. En efecto, para habilitar las comunicaciones cifradas entre usuarios necesita que todos intercambien sus claves públicas. Después, cada usuario que las recibe debe verificar y firmar las claves públicas. Las claves públicas son confiables para descifrar mensajes de manera segura.

Si envía mensajes a unos cuantos usuarios, envíeles su clave pública de manera manual por correo electrónico. Para uso público general, publique su clave pública en un servidor de claves, con lo que cualquiera puede descargarla y utilizarla para descifrar cualquier mensaje recibido de usted. Es posible acceder a un servidor de claves mediante correo electrónico, LDAP o HTTP Horwitz Keyserver Protocol (HKP). El proyecto OpenPGP Public Keyserver se ubica en **pks.sourceforge.net**. Hay varios servidores de claves públicas disponibles. **hkp://subkeys.pgp.net** muestra en una lista de su archivo **.gnupg/gpg.conf**, aunque sin comentarios. Envíe directamente al servidor de claves con la opción **-keyserver** y el comando **--send-key**. Éste toma como argumento su dirección de correo electrónico. Sólo es necesario que envíe a un servidor de claves, porque éste compartirá automáticamente su clave con otros servidores de claves.

```
gpg --keyserver search.keyserver.net --send-key carlos@tortuga.mipista.com
```

Si quiere enviar su clave directamente a otro usuario, debe generar una versión de texto blindada de la clave que después puede enviar por correo electrónico. Haga esto con las opciones **--armor** y **--export**, utilizando la opción **--output** para especificar el archivo en que se colocará la clave. La opción **--armor** generará una versión de texto ASCII del archivo cifrado, para que se envíe directamente por correo electrónico, en vez de adjuntarlo como archivo binario. Los archivos que almacenan una versión cifrada ASCII del cifrado suelen tener una extensión **.asc**, por convención. Los archivos binarios cifrados normalmente utilizan la extensión **.gpg**. Luego envíe por correo electrónico el archivo a los usuarios a quienes quiere enviar los mensajes cifrados.

```
# gpg --armor --export ricardolp@tortuga.mipista.com --output jorgeap.asc
# mail -s 'miclavepublica' jorge@conejo.mipista.com < jorgeap.asc
```

Muchas compañías e instituciones hacen públicas sus claves en sitios Web, de donde pueden se descargar y usan para verificar las descargas de software o anuncios cifrados.

NOTA Algunos comandos y opciones para GPG tienen formas largas y cortas. Por ejemplo, el comando **--armor** puede escribirse como **-a**, **--output** como **-o**, **--sign** como **-s** y **--encrypt** como **-e**. Muchos otros, como **--export**, no tienen forma corta.

Obtención de claves públicas

Para descifrar mensajes de otros usuarios, necesita sus claves públicas. Ya sea que se las envíen o las descargue de un servidor de claves. Guarde el mensaje o página Web conteniendo la clave pública de un archivo. Después deberá importar, verificar y firmar la clave. Use el archivo que recibió para importar la clave pública a su archivo **pubring**. En el siguiente ejemplo, el usuario importa la clave pública de Jorge, que ha recibido como el archivo **clavejorge.asc**.

```
gpg --impor clavejorge.asc
```

Todos los sitios de distribución de Linux tienen sus propias claves públicas disponibles para descarga. Por ejemplo, debe descargar la clave pública de Red Hat, que puede encontrarse en el sitio de Red Hat, en su página de recursos de seguridad (**redhat.com**). Haga clic en el vínculo Public Encryption Key. Desde ahí, acceda a la página desplegando sólo la clave pública. Guarde esta página como archivo y utilice éste para importar la clave pública de Red Hat a su anillo. (Una distribución de Red Hat también coloca la clave pública de Red Hat en el directorio **/usr/share/doc/rpm4-1**, con versiones para cifrado de GPG y PGP, los archivos **RPM-GPG-KEY** y **RPM-PGP-KEY**).

En el siguiente ejemplo, el usuario guarda la página que muestra la clave pública de Red Hat como **miredhat.asc**, y después importa ese archivo.

```
gpg --import miredhat.asc
```

NOTA Puede eliminar cualquier clave, al incluir su propia clave privada, con los comandos **--delete-key** y **--delete-secret-key**.

Validación de claves

Para comprobar manualmente que el archivo de clave pública no fue modificado en la transmisión, puede revisar su huella. Éste es un valor de hash generado desde el contenido de la clave, muy similar a un compendio de modificación. Al utilizar la opción **--fingerprint**, puede generar un valor hash a partir de la clave instalada, después ponerse en contacto con el emisor y preguntarle qué valor hash debe estar ahí realmente. Si no son los mismos, sabe que la clave fue manipulada en la transmisión.

```
gpg --fingerprint jorge@conejo
```

No tiene que revisar la huella para hacer que **gpg** funcione. Ésta es sólo una precaución recomendable que puede tomar por cuenta propia. Lo importante es que necesita estar seguro de que la clave recibida es válida. Usualmente, puede aceptar la mayor parte de las claves de servidores públicos o sitios conocidos como válidas, aunque es sencillo revisar sus huellas publicadas. Una vez seguro de la validez de la clave, fírmela con su clave privada. Al firmar una clave se notifica a **gpg** que aceptó oficialmente la clave.

Para firmar una clave, utilice el comando **gpg** con el comando **--sign-key** y el nombre de la clave.

```
gpg --sign-key jorge@conejo
```

Como opción, edite la clave con el comando **--edit-key** para iniciar una sesión interactiva en que insertará el comando **sign**, para firmar la clave y **save**, para guardar el cambio. Para firmar una clave requiere acceder a su clave privada, así que se le pedirá su frase de contraseña. Cuando termine, deje la sesión interactiva con el comando **quit**.

Por lo general, querrá publicar una versión de su clave pública firmada por uno o más usuarios. Haga lo mismo para otros usuarios. Firmar una clave pública ofrece una forma de garantizar la validez de una clave. Indica que alguien más ya la revisó. Varios usuarios diferentes pueden firmar la misma clave pública. Una vez haya recibido y verificado una clave de otro usuario, firme y regrese la versión firmada a ese usuario. Después de firmar la clave, genere un archivo conteniendo la versión pública firmada. Envíe este archivo al usuario. Este proceso genera una telaraña de confianza, donde varios usuarios garantizan la validez de las claves públicas.

```
gpg -a --export jorge@conejo --output firmajorge.asc
```

El usuario después importa la clave firmada y exporta al servidor de claves.

TIP Si quiere iniciar desde el principio, puede borrar el directorio **.gnupg**, aunque esto es una medida drástica, ya que perderá cualquier clave recolectada.

Uso de GnuPG

El cifrado de GnuPG tiene soporte actualmente para casi todos los clientes de correo, incluidos Kmail, Thunderbird y Evolution. También puede utilizar GNU Privacy Assistant (GPA), una interfaz gráfica de usuario (GUI), para administrar sus tareas de GPG o el comando **gpg** para cifrar y descifrar mensajes manualmente, incluida la firma digital, si lo desea. Mientras realiza tareas de GPG, necesitará referir claves que tiene mediante sus nombres de clave. Tenga en cuenta que sólo necesita una subcadena de identificación única para seleccionar la clave requerida. GPG realiza una búsqueda de patrón en la cadena especificada como nombre de clave por cualquier comando dado. Si la cadena coincide con más de una clave, todas las coincidencias se seleccionarán. En el siguiente ejemplo, la cadena "Sendmail" selecciona las coincidencias de las identidades de dos claves.

```
# gpg --list-keys "Sendmail"
pub 1024R/CC374F2D 2000-12-14
    Sendmail Signing Key/2001 sendmail@Sendmail.ORG
pub 1024R/E35C5635 1999-12-13
    Sendmail Signing Key/2000 sendmail@Sendmail.ORG
```

Cifrado de mensajes

El comando **gpg** suministra varias opciones para administrar mensajes seguros. La opción **e** cifra mensajes, la opción **a** genera una versión de texto blindada y la opción **s** agrega una firma digital. Necesitará especificar la clave pública del destinatario, que ya debió importar a su archivo **pubring**. Es la clave usada para cifrar el mensaje. El destinatario entonces podrá descifrar el mensaje con su propia clave privada. Emplee la opción **--recipient** o **-r** para especificar el nombre de la clave del destinatario. Utilice cualquier subcadena única en el nombre de la clave pública del usuario. La dirección de correo electrónico suele ser suficiente. Use la opción **d** para descifrar mensajes recibidos. En el siguiente ejemplo, el usuario cifra (**e**) y firma (**s**) un archivo generado en formato de texto blindado (**a**). La opción **-r** indica el destinatario del mensaje (cuya clave pública se utiliza para cifrar el mensaje).

```
gpg -e -s -a -o miarchivo.asc -r jorge@conejo.mipista.com miarchivo
# mail jorge@conejo.mipista.com < miarchivo.asc
```

Puede dejar fuera la opción de blindaje ASCII, si quiere enviar o transferir el archivo como datos adjuntos binarios. Sin la opción **--armor** o **-a**, **gpg** genera un archivo binario cifrado, no un archivo de texto cifrado. Un archivo binario sólo puede transmitirse a través de un correo electrónico como archivo adjunto. Como se notó antes, las versiones blindadas ASCII suelen tener una extensión **.asc**, mientras la versión binaria utiliza **.gpg**.

NOTA Utilice **gpgsplit** para dividir un mensaje GPG para mostrar sus componentes para examinarlos por separado.

Descifrado de mensajes

Cuando otros usuarios reciben el archivo, lo guardan en un archivo llamado, algo así como **miarchivo.asc** y después lo descifran el archivo con la opción **-d**. La opción **-o** especificará un archivo en que se guardará la versión descifrada. GPG determinará automáticamente si es un archivo binario o una versión blindada ASCII.

```
gpg -d -o miarchivo.txt miarchivo.asc
```


Para revisar la firma digital del archivo, use el comando **gpg** con la opción **--verify**. Esto garantiza que el emisor firmó el archivo.

```
gpg --verify miarchivo.asc
```

Descifrado de una firma digital

Necesitará una clave pública del firmante para descifrar y revisar la firma digital. Si no, recibirá un mensaje indicando que la clave pública no se encontró. En este caso, primero debe obtener la clave pública del firmante. Acceda al servidor de claves asumiendo que puede tener la clave pública o pida la clave pública directamente desde un sitio Web o al firmante. Después importe la clave como se describió antes.

Firma de mensajes

No tiene por qué cifrar un archivo para firmarlo. Una firma digital es un componente separado. Puede combinar la firma con un archivo dado o generar uno por separado. Para combinar una firma con un archivo, genere una nueva versión incorporando ambos. Use la opción **--sign** o **-s** para generar una versión del documento incluyendo una firma digital. En el siguiente ejemplo, el archivo **midoc** se firma digitalmente con el archivo **midoc.gpg** que contiene el archivo original y la firma.

```
gpg -o midoc.gpg --sign midoc
```

Si, en cambio, sólo quiere generar un archivo de firma separado, use el comando **--detach-sig**. Tiene la ventaja de que no es necesario generar una copia completa del archivo original. El archivo permanece intacto. El archivo de firma suele tener una extensión como **.sig**. En el siguiente ejemplo, el usuario crea un archivo de firma llamado **midoc2sig** para el archivo **midoc2**.

```
gpg -o midoc2.sig --detach-sig midoc2
```

Para verificar el archivo con una firma separada, el destinatario especifica archivo de firma y archivo original.

```
gpg --verify midoc2.sig midoc2
```

Para verificar una firma confiable utilice **gpgv**.

También puede generar una firma de signo claro para utilizarse en archivos de texto. Una firma de *signo claro* es una versión de texto de la firma que puede adjuntarse a un archivo. El archivo de texto puede editarse más adelante con cualquier editor de texto. Utilice la opción **--clearsign** para crear una firma de signo claro. En el siguiente ejemplo se crea una versión de signo claro de un archivo llamado **minoticia.txt**.

```
gpg -o mifirmanoticia.txt --clearsign minoticia.txt
```

NOTA Varias interfaces GUI y filtros están disponibles para GnuPG en www.gnupg.org. GPA proporciona una interfaz basada en GNOME para cifrar y descifrar archivos de manera sencilla. Seleccione los archivos para cifrar, elija los destinatarios (claves públicas que habrán de utilizarse) y agregue una firma digital, si lo desea. También utilice GPA para descifrar archivos cifrados recibidos. Administre su colección de claves públicas, que son las de su archivo de anillo.

SUGERENCIA *Steganography es una forma de cifrado para ocultar datos en otros tipos de objetos, como imágenes. Utilice el software JPEG Hide and Seek (JPHS) para cifrar y recuperar datos en una imagen JPEG ([jphide](#) y [jpseek](#)). Consulte [linux01.gwdg.de/~alatham/stego.html](#) para conocer más detalles.*

Revisión de las firmas digitales de software

Una forma efectiva de utilizar firmas digitales consiste en verificar que un paquete de software no se haya modificado. Un paquete de software puede ser interceptado en la transmisión y algunos de sus archivos en el nivel del sistema pudieron ser cambiados o sustituidos. Los paquetes de software de su distribución, además de los acreditados por los proyectos GNU y Linux, están firmados digitalmente. La firma proporciona información de compendio de modificación con que puede revisar la integridad de los paquetes. La firma digital se incluye con el paquete de archivos o se publica como archivo separado. Use el comando **gpg** con la opción **--verify** para revisar la firma digital de un archivo.

Importación de claves públicas

Sin embargo, primero deberá asegurarse que tiene la clave pública del firmante. La clave pública fue cifrada con la clave privada del distribuidor de software; ese distribuidor es el firmante. Apenas tenga la clave pública del firmante, puede revisar cualquier dato recibido de ellos. En el caso de depósitos de software de terceros como [freshrpms.net](#), se le preguntará si quiere instalar su clave pública la primera vez que intenta instalar cualquier software desde ese sitio. Una vez la clave está instalada, no necesita instalarla nuevamente. Con Yellowdog Updater Modified (Yum), suele ser únicamente una petición para instalar la clave, que le pide confirmación con **y** o **n**; cuando no es así, se trata es un cuadro de diálogo, solicitando un clic en Aceptar. Los depósitos como [Livna](#) y [Freshrpms.net](#) incluirán sus claves con sus paquetes de configuración Yum. También, si desea, descárguelos e instálelos manualmente.

En el caso de un distribuidor de software, descargue la clave pública de su sitio Web o su keyserver. Una vez que la tenga, podrá revisar cualquier programa de dicho distribuidor.

Como se vio anteriormente, puede descargar la clave pública de Red Hat, desde el sitio Web de Red Hat, en la página de recursos de seguridad o utilice la versión instalada en el directorio de documentación Red Hat Package Manager (RPM). Una vez que obtenga la clave pública, agregue a su anillo la opción **-import**, especificando el nombre que dio al archivo de clave descargado (en este caso, **miredhat.asc**):

```
# gpg -import redhat.asc
gpg: key CBA29BF9: public key imported
gpg: Total number processed: 1
gpg: imported: 1 (RSA: 1)
```

Para descargar desde un servidor de claves, utilice la opción **--keyserver** y el nombre del servidor de claves.

Para importar una clave pública de Red Hat desde el directorio RPM, especifique el archivo. Éste es la clave proporcionada por la distribución Red Hat en su DVD-ROM o CD-ROM. Aunque se usa durante la instalación, la clave tiene que importarse para verificar nuevamente los paquetes después de instalados.

```
rpm --import /usr/share/doc/rpm-4.2.2/RPM-GPG-KEY
```

Validación de claves públicas

Puede utilizar la opción **--fingerprint** para revisar la validez de una clave, si lo desea. Si está seguro de que la clave es válida, entonces fírmela con el comando **--sign-key**. En el siguiente ejemplo, el usuario firma la clave Red Hat, al emplear la cadena "Red Hat" en el nombre de la clave para hacer referencia a éste. También se le pide al usuario ingrese su frase de contraseña para permitir el uso de su clave privada para firmar la clave pública de Red Hat.

```
# gpg --sign-key "Red Hat"
pub 1024R/CBA29BF9 created: 1996-02-20 expires: never trust: -/q
(1). Red Hat Software, Inc. <redhat@redhat.com>
pub 1024R/CBA29BF9 created: 1996-02-20 expires: never trust: -/q
Fingerprint: 6D 9C BA DF D9 60 52 06 23 46 75 4E 73 4C FB 50
Red Hat Software, Inc. <redhat@redhat.com>

Are you really sure that you want to sign this key
with your key: "Richard Petersen (author) <richlp@turtle.mytrek.com>"
Really sign? yes
You need a passphrase to unlock the secret key for
user: "Richard Petersen (author) <richlp@turtle.mytrek.com>"
1024-bit DSA key, ID 73F0A73C, created 2001-09-26
Enter passphrase:
#
```

Revisión de los paquetes RPM

Los paquetes RPM desde cualquier depósito Yum o Apt, revisarán automáticamente la clave pública. Si descarga un paquete RPM por separado, revise el paquete de manera manual. Una vez que tenga la clave pública del proveedor de software, revise cualquier paquete de software RPM con el comando **rpm** y la opción **-K**. En el siguiente ejemplo se revisa la validez del paquete de software xcdroast:

```
# rpm -K xcdroast-0.98alpha9-1.i386.rpm
xcdroast-0.98alpha9-1.i386.rpm: md5 OK
```

Muchos paquetes de software en forma de archivos comprimidos como, **.tar.gz** o **tar.bz2**, proporcionarán firmas en archivos separados terminando con la extensión **.sig**. Para revisar éstos, use el comando **gpg** con la opción **--verify**. Por ejemplo, el paquete Sendmail más reciente se distribuye en forma de archivo comprimido, **.tar.gz**. Su firma digital se proporciona en un archivo **.sig** separado. Primero descargue e instale la clave pública para el software Sendmail, obtenida desde el sitio Web de Sendmail (la clave puede tener el año como parte de su nombre).

```
# gpg --import sendmail2006.asc
```

Después, firme la clave pública de Sendmail que acaba de importar. En este ejemplo, la dirección de correo electrónico se utiliza para el nombre de clave

```
gpg --sign-key sendmail@Sendmail.ORG
```

También puede revisar la huella de la clave para su verificación agregada.

En seguida descargue el archivo comprimido y el archivo de la firma digital. Para el archivero comprimido (**.tar.gz**) use el archivero **.gz.sig**, y para el archivero descomprimido

utilice **.tar.sig**. Después, con el comando **gpg** y la opción **--verify**, utilice la firma digital en el archivo **.sig** para revisar la autenticidad e integridad del archivo comprimido del software.

```
# gpg --verify sendmail.8.13.8.tar.gz.sig sendmail.8.13.8.tar.gz
gpg: Signature made Tue 08 Aug 2006 10:24:45 PM PDT using RSA key ID AF959625
gpg: Good signature from "Sendmail Signing Key/2006 <sendmail@Sendmail.ORG>"
```

También Puede especificar sólo la firma digital y **gpg** buscará automáticamente un archivo seleccionado con el mismo nombre, pero sin la extensión **.sig** o **.asc**.

```
# gpg --verify sendmail.8.12.0.tar.sig
```

En el futuro, cuando descargue cualquier software del sitio de Sendmail empleando esta clave, sólo debe realizar la operación **--verify**. Sin embargo, tenga en cuenta que diferentes paquetes de software del mismo sitio pueden usar claves diferentes. Tendrá que asegurarse de que ha importado y firmado la clave apropiada para el software que está revisando.

Detección de intrusión: Tripwire y AIDE

Cuando alguien irrumpe en su sistema, generalmente tratará de obtener control creando sus propios cambios a los archivos de administración del sistema, como archivos de contraseñas. Ellos pueden crear su propia información de usuario y contraseña, permitiéndose el acceso en cualquier momento o sólo cambiarán la contraseña de root. También pueden remplazar programas enteros, como el programa de inicio de sesión, con su propia versión. Un método para detectar tales acciones consiste en utilizar una herramienta de revisión de integridad, como Tripwire o AIDE (Advanced Intrusion Detection Environment, entorno avanzado de detección de intrusión), para detectar cualquier cambio a los archivos de administración del sistema. AIDE es una opción a Tripwire. Brinda configuración de acceso y reportes detallados.

Una herramienta de revisión de integridad funciona creando primero una base de datos de identificadores únicos para cada archivo o programa que habrá de revisarse. Éstas incluyen características como permisos y tamaño de archivos, pero más importante, incluyen números de suma de verificación generados por algoritmos de cifrado desde el contenido del archivo. Por ejemplo, en Tripwire, los identificadores predeterminados son números de suma de verificación creados por algoritmos como SHA2, de compendio de modificación y Snefru (algoritmo de hash de secuencia de Xerox). Un valor de cifrado que proporciona tal identificación única de un archivo, es conocido como firma. En efecto, una firma proporciona una instantánea precisa del contenido de un archivo. Los archivos y programas se revisan entonces al generar de nuevo sus identificadores, comparándolos con los que se encuentran en la base de datos. Tripwire generará firmas de archivos y programas actuales, que comparará con los valores generados antes para su base de datos. Cualquier diferencia se observa como cambios en el archivo, y Tripwire notifica entonces los cambios.

NOTA También puede revisar sus archivos de registro en busca de cualquier actividad sospechosa. El archivo */var/log/messages*, en particular, es útil para revisar eventos críticos como inicios de sesión de usuario, conexiones FTP e inicios de sesión de superusuario.

Sistemas de archivos cifrados

Linux permite cifrar sistemas de archivos ajenos a la raíz o de intercambio, permitiendo acceder sólo a usuarios contando con la contraseña cifrada adecuada. Puede aplicar cifrado a los sistemas de archivos fijos y extraíbles, como dispositivos USB. Se recomienda usar herramientas de cifrado Luks (Linux Unified Key Setup) para cifrar sus sistemas de archivos. Utilice la herramienta **gnome-luks-format** o **cryptsetup** directamente para configurar su sistema de archivos cifrado. Si está disponible para su distribución, la manera más sencilla de configurar un sistema de archivos cifrado es mediante la herramienta. A través de esta herramienta, especifica el sistema de archivos, cifrador y frase de contraseña, así como tipo y nombre del sistema de archivos. Asegúrese de que el sistema de archivos no está montado.

Una vez formateado, reinicie su sistema. Después acceda a la partición cifrada o unidad extraíble. Para una unidad USB o disco, desde la ventana del sistema de archivos haga doble clic en el icono de la unidad USB. Esto abre una ventana en que se pide una contraseña con la opción para olvidarla, recordarla durante la sesión o recordarla siempre. Un mensaje le indica que el dispositivo está cifrado. Una vez inserte su contraseña, móntelo y acceda al dispositivo (haga doble clic en éste nuevamente). El nombre del volumen aparecerá con un icono en su escritorio. HAL manejará todo montaje y acceso a medios extraíbles. Recurra al mismo procedimiento para particiones fijas. En lugar de reiniciar su sistema tras la inicialización y formato, utilice **luks-setup** o **cryptsetup**, con la opción **luksOpen** para abrir un sistema de archivos cifrado. Si quiere administrar manualmente unidades fijas, coloque entradas en los archivos **/etc/crypttab** y **/etc/fstab** para éstos.

En vez de utilizar **gnome-luks-format**, utilice el comando **cryptsetup** directamente para configurar de forma manual su sistema de archivos cifrado. Primero utilice el comando **cryptsetup** con la opción **luksFormat** para inicializar y crear un volumen cifrado. Se le pedirá especifique una clave (o agregue el archivo de clave como argumento). Agregue una entrada para el volumen en el archivo **/etc/crypttab**. Después reinicie o utilice el comando **cryptsetup**, con la opción **luksOpen** para acceder al volumen. Se le pedirá una clave (o utilice **--keyfile** para especificar la clave). Después de dar formato al sistema, especificando su nombre y tipo, coloque una entrada para el nuevo sistema de archivos en el archivo **/etc/fstab**.

Si no utiliza Luks, deberá especificar un método de cifrado con la opción **cypher**. Utilice la opción **--cypher** con **cryptsetup** y la entrada **/etc/crypttab**. En el caso de un cifrador ESSIV, utilice **aes-cbc-essiv:sha256**. Para un cifrador simple, utilice **aes-cbc-plain**.

Linux con seguridad mejorada

Aunque existen varias herramientas de seguridad para proteger servicios específicos, además de información y datos de usuario, no ha existido una herramienta para proteger todo el sistema a nivel administrativo. Security-Enhanced Linux es un proyecto para dar protección administrativa integrada a aspectos de su sistema Linux. En lugar de depender de usuarios para proteger sus archivos o un programa de red específico para controlar el acceso, las medidas de seguridad están integradas en el sistema de administración de archivos básico y los métodos de acceso a la red. Un administrador maneja directamente todos los controles como parte de la administración del sistema Linux.

Security-Enhanced Linux (SELinux) es un proyecto desarrollado y mantenido por la National Security Agency (NSA), cuya elección es Linux como plataforma para implementar un sistema operativo seguro. Casi todas las distribuciones de Linux tienen SELinux incluido incorporada como característica estándar de esta distribución. La documentación detallada está disponible en los recursos mostrados en la lista de la Tabla 17-1, incluidos los sitios proporcionados por NSA y SourceForge. También revise el sitio de su distribución para adquirir manuales, preguntas más frecuentes o documentación acerca de SELinux.

Los sistemas Linux y Unix suelen usar un método de control de acceso discrecional (DAC, Discretionary Access Control) para restringir el acceso. Con este método, usuarios y objetos de su propiedad, como archivos, determinan los permisos. El usuario tiene discreción completa sobre los objetos que le pertenecen. El punto débil en muchos sistemas Linux y Unix ha sido las cuentas administrativas de usuario. Si un atacante logra obtener acceso a una cuenta administrativa, tendrá control completo sobre el servicio manejando la cuenta. El acceso al usuario root le dará control sobre todo el sistema, sus usuarios y cualquier servicio de red en ejecución. Para contrarrestar esta debilidad, NSA configura una estructura de control de acceso obligatorio (MAC, Mandatory Access Control). En vez de un conjunto de privilegios tipo todo o nada, basado en cuentas, servicios y tareas administrativas se dividen en secciones y controlan por separado con políticas detallando qué se puede hacer y qué no. El acceso no sólo se permite porque uno sea un usuario autenticado, sino porque se cumplen ciertos criterios de seguridad específicos. A usuarios, aplicaciones, archivos y dispositivos sólo se les da el acceso necesario para administrar su trabajo y nada más.

Arquitectura de Flask

La arquitectura de Flask organiza los componentes y datos del sistema operativo en temas y objetos. Los temas son procesos: aplicaciones, unidades, tareas de sistema actualmente en ejecución.

Recursos	Ubicación
NSA SELinux	nsa.gov/selinux
NSA SELinux FAQ	nsa.gov/selinux/info/faq.cfm
SELinux at sourceforge.net	selinux.sourceforge.net
Escritura de la directiva HOWTO de SELinux	Puede accederse desde el vínculo “SELinux resources at sourceforge” en selinux.sourceforge.net
Documentación de las directivas de SELinux	nsa.gov/selinux/info/docs.cfm
Configuración de las directivas de SELinux	Se tiene acceso desde Documentación de las directivas de SELinux
SELinux Reference Policy Project	http://oss.tresys.com/projects/refpolicy

TABLA 17-1 Recursos de SELinux

Los objetos son componentes fijos como archivos, directorios, conectores, interfaces de red y dispositivos. Se define un contexto de seguridad para cada sujeto y objeto. Un *contexto de seguridad* es un conjunto de atributos de seguridad determinando la manera en que se usa un tema u objeto. Este método proporciona control muy delicado sobre cada elemento en el sistema operativo, además de todos los datos en su computadora. Los atributos diseñados para contextos de seguridad y grado al que éstas se imponen, están determinados por una directiva de seguridad general. Un servidor de seguridad impone las directivas. Las distribuciones pueden proporcionar diferentes directivas configuradas con anterioridad a partir de las que trabajan. Por ejemplo, Fedora implementa tres directivas, cada una con su propio paquete: strict, targeted y mls, variación de una directiva de referencia.

SELinux utiliza una combinación de modelos de seguridad de imposición de tipo (TE, Type Enforcement), control de acceso basado en funciones (RBAC, Role Based Access Control) y seguridad de varios niveles (MLS, Multi-Level Security). La imposición de tipo se concentra en objetos y procesos, como directorios y aplicaciones, mientras la imposición de acceso basado en funciones controla el acceso de usuarios. Para el modelo de imposición de tipo, los atributos de seguridad asignados a un objeto conocen como dominios o tipos. Los tipos se utilizan para objetos fijos como archivos y los dominios se utilizan para procesos como ejecución aplicaciones. Para el acceso de usuario a procesos y objetos, SELinux hace uso del control de acceso basado en funciones. Cuando se crean nuevos procesos u objetos, las reglas de transición especifican el tipo o dominio al que pertenece, en los contextos de seguridad.

Con el modelo RBAC, a los usuarios se les asignan funciones para las que se definen permisos. Las funciones restringen objetos y procedimientos a los que un usuario accede. El contexto de seguridad para los procesos incluirá un atributo de función, controlando los objetos que puede evaluar. La nueva seguridad de varios niveles (MLS, Multi-Level Security) agrega un nivel de seguridad, conteniendo un valor de confidencialidad y capacidad.

A los usuarios se les da una identidad de usuario de SELinux separada. Aunque pueden tener el mismo nombre, no son los mismos identificadores. Las identidades estándar de Linux pueden cambiarse de manera sencilla con comandos como **setuid** y **su**. Los cambios al ID de usuario de Linux no afectarán el ID de SELinux. Esto significa que, pese un usuario cambiando su ID, SELinux todavía podrá rastrearlo, manteniendo el control sobre ese usuario.

Acceso de administración de sistema

Es críticamente importante se asegure que tiene acceso administrativo al sistema bajo SELinux, antes de imponer directivas. Esto es especialmente cierto si usa una directiva `strict` o `mls`; para imponer restricciones en acceso administrativo. Siempre debe utilizar SELinux en el modo permisivo, primero, y revisar cualquier mensaje denegando el acceso. Con SELinux impuesto, tal vez no importe si puede acceder al usuario `root`. Importa que su usuario, aun el `root`, tenga la función `sysadm_r` y acceso a objetos y el nivel de seguridad administrativa `sysadm_t`. Tal vez no pueda utilizar el comando `su` para acceder al usuario `root` y esperar el acceso administrativo del usuario `root`. Recuerde que SELinux mantiene sus propias identidades de seguridad, que difieren del ID de usuario de Linux. Aunque tal vez quiera cambiar su ID de usuario con `su`, todavía tiene que cambiar su ID de seguridad.

La directiva de destino configurará reglas para el acceso de administrador de sistema estándar mediante procedimientos de Linux normales. En general, el usuario `root` accederá la cuenta de usuario `root`. Sin embargo, en la directiva `strict`, el usuario `root` necesita acceder a su cuenta, mediante una ID de seguridad apropiada. Ambas son parte ahora de una sola directiva de referencia. Si quiere acceso administrativo mediante el comando `su` (desde otro usuario), primero emplee `su` para iniciar sesión como usuario `root`. Luego debe cambiar su función a `sysadm_r`, y ya estar configurado por las reglas de directivas SELinux para permitirse tome la función `sysadm_r`. Un usuario puede asumir varias funciones posibles.

Para cambiar la función, utilice el comando `newrole` con la opción `-r`.

```
newrole -r sysadm_r
```

Terminología

SELinux maneja varios términos con diferentes significados en otros contextos. La terminología puede resultar confusa porque algunos términos, como *dominio*, tienen diferentes significados en otras áreas relacionadas. Por ejemplo, un dominio en SELinux es un proceso opuesto a un objeto, mientras en redes el término refiere direcciones DNS de red.

Identidad

SELinux crea identificadores con los que controla el acceso. Las identidades no son lo mismo que los ID de usuario tradicionales. Al mismo tiempo, cada usuario suele tener una identidad de SELinux, aunque las dos no están vinculadas. Afectar un usuario no afecta a la identidad correspondiente de SELinux. Éste configura una identidad correspondiente separada para cada usuario, aunque en la las directivas menos seguras, como las orientadas, se manejan identidades generales. Una identidad de usuario general se emplea para usuarios normales, que restringe el acceso a nivel de usuario, mientras los administradores tienen identidades administrativas. Puede definir mucho más las identidades de seguridad para usuarios particulares.

La identidad crea parte de un contexto de seguridad determinando lo que puede o no hacer un usuario. Si un usuario cambia las ID de usuario, la identidad de seguridad del usuario no cambiará. Un usuario siempre tendrá la misma identidad de seguridad. En sistemas de Linux tradicionales, un usuario usa comandos como `su` para cambiar ID y convertirse en otro usuario. En SELinux, aunque un usuario puede cambiar su ID de usuario de Linux, el usuario aún retiene la misma ID de seguridad original. Siempre sabrá lo que está haciendo en su sistema cierta persona en particular, sin importar qué ID de usuario asuma la persona.

La identidad de seguridad puede tener acceso limitado. Así que, aunque un usuario pueda utilizar el comando `su` para convertirse en usuario `root`, la identidad de seguridad del usuario puede evitar que recurra a cualquier comando administrativo permitido para el usuario `root`. Como

ya se observó, para obtener acceso administrativo, también tendrá que cambiar la función de la identidad de seguridad.

Utilice `id -z` para ver qué contexto de seguridad tiene su identidad de seguridad, cuáles funciones tiene y a qué tipo de objetos puede acceder. Esto mostrará una lista del contexto de seguridad de usuario iniciando con el ID de seguridad, seguido por dos puntos y las funciones que tiene un usuario, así como los objetos que puede controlar. Las identidades de seguridad pueden tener funciones controlando lo que pueden hacer. Una función de usuario es `user_r` y una función de administración del sistema es `system_r`. La identidad de seguridad general es `user_u`, mientras una identidad de seguridad particular suele manejar el nombre de usuario. En el siguiente ejemplo se muestra un usuario estándar con identidad de seguridad generada:

```
$ id -z
user_u: user_r:user_t
```

En este ejemplo el usuario tiene una identidad llamada `jorge`:

```
$ id -z
jorge:user_r:user_t
```

Puede usar el comando `newrole` para cambiar la función permitida al usuario. Al cambiar a una función administrativa de sistema, el usuario puede tener acceso equivalente a `root`.

```
$ id -z
jroge:sysadm_r:sysadm_t
```

Dominios

Los *dominios* se emplean para identificar un proceso de control. A cada proceso se asigna un dominio dentro del cual puede ejecutarse. Un dominio establece restricciones acerca de los procesos que puede usar. Generalmente, a un proceso se daba un ID de usuario para determinar que podía hacer y muchos debían tener un ID de usuario `root` para acceder a todo el sistema de archivos. También podía usarse para tener acceso administrativo completo sobre todo el sistema. Un dominio, por otro lado, puede hacerse a la medida para acceder ciertas áreas pero no otras. Los intentos de irrumpir en un dominio, como el administrativo, se bloquearán. Por ejemplo, el dominio administrativo es `sysadm_t`, mientras el servidor DNS utiliza sólo `namad_t` y los usuarios tienen un dominio `user_t`.

Tipos

Mientras los dominios controlan procesos, los *tipos* controlan objetos como archivos y directorios. Los archivos y directorios se agrupan en tipos que pueden utilizarse para controlar quién accede a éstos. Los nombres de tipo tienen el mismo formato que los de dominio, terminando con un sufijo `_t`. A diferencia de los dominios, los tipos refieren a objetos, incluidos archivos, dispositivos e interfaces de red.

Funciones

Los tipos y dominios se asignan a funciones. Los usuarios (identidades de seguridad) con una función dada acceden tipos y dominios asignados a esa función. Por ejemplo, la mayoría de usuarios acceden objetos tipo `user_t`, pero no objetos `sysadm_t`. Los tipos y dominios a que puede acceder un usuario se establecen en la entrada de la función en los archivos de configuración. En el siguiente ejemplo se permite a los usuarios acceder objetos con el tipo de contraseña de usuario:

```
role user_r types user_passwd_t
```

Contexto de seguridad

Cada objeto tiene un contexto de seguridad estableciendo atributos de seguridad. Éste incluye identidad, función, dominio y tipo. Un archivo tendrá un contexto de seguridad mostrando una lista del tipo de identidad a la que accede, la función bajo la que tiene acceso y tipo de seguridad al que pertenece. Cada componente agrega su propio nivel refinado de seguridad. Los objetos pasivos suelen asignarse a una función genérica, `objet_r`, sin efecto, porque los objetos no pueden iniciar acciones.

Un archivo normal creado por usuarios en sus propios directorios tendrá identidad, función y tipo siguientes. La identidad es un usuario y la función es la de un objeto. El tipo es el directorio home del usuario. El tipo se maneja para todos los subdirectorios y archivos creados dentro de un directorio home del usuario.

```
user_u:object_r:user_home_t
```

Un archivo o directorio creado por el mismo usuario en una parte diferente del sistema de archivos tendrá un tipo diferente. Por ejemplo, el tipo de archivos creados en el directorio `/tmp` será `tmp_t`

```
user_u:object_r:tmp_t
```

Transición: etiquetado

Una *transición*, conocida también como etiquetado, asigna contexto de seguridad a un proceso o archivo. En el caso de un archivo, el contexto de seguridad se asigna cuando se crea, mientras un proceso el contexto de seguridad se determina cuando el proceso se ejecuta.

Asegúrese de que cada archivo tiene un contexto de seguridad apropiado llamado *etiquetado*. Para agregar otro sistema de archivos requiere etiquetar (agregue contextos de seguridad) los directorios y archivos en éste. El etiquetado varía, dependiendo de la directiva usada. Cada directiva puede tener contextos de seguridad diferentes para objetos y procesos. El reetiquetado se lleva a cabo con el comando **fixfiles** en el directorio de origen de la directiva.

```
fixfiles relabel
```

Directivas

Una *directiva* es un conjunto de reglas para determinar relaciones entre usuarios, funciones y tipos o dominios. Estas reglas declaran a qué tipos accede una función y qué funciones tiene un usuario.

Multi-Level Security (MLS) y Multi-Category Security (MCS)

Multi-Level Security (MLS, seguridad de varios niveles) agrega un método de seguridad de acceso más refinado, diseñado para servidores. MLS agrega un valor de nivel de seguridad a los recursos. Sólo los usuarios con acceso a ciertos niveles acceden archivos y aplicaciones correspondientes. Dentro de cada nivel, el acceso se controla aún más con el uso de categorías. Las categorías funcionan de manera muy similar a los grupos, permitiendo acceso sólo a usuarios autorizados para esa categoría. El acceso se vuelve más refinado, en vez de presentarse una situación todo o nada.

Multi-Category Security (MCS, seguridad de varias categorías) extiende SELinux para ser usado no sólo por administradores, sino usuarios. Éstos configuran categorías restringiendo y controlando el acceso a sus archivos y aplicaciones. Aunque está basado en MLS, sólo utiliza categorías, no niveles de seguridad. Los usuarios seleccionan una categoría para un archivo, pero sólo el

administrador puede crear una categoría y determinar qué usuarios tienen acceso a ella. Aunque es similar en concepto a las listas de control de acceso (ACL, Access Control List), difiere en que hace uso de la estructura de seguridad SELinux, ofreciendo implementación de control en el nivel usuario impuesto por SELinux.

Las operaciones de administración para SELinux

Ciertas operaciones básicas, como revisión del estado de SELinux y contexto de seguridad de un usuario o archivo, o despliegue de SELinux en el inicio del sistema, son muy útiles.

Desactivación de SELinux

Si quiere desactivar SELinux aun antes de iniciar su sistema, hágalo en el indicador de comandos de inicio. Sólo agregue el siguiente parámetro, al final de su línea GRUB de inicios.

```
selinux:0
```

Para desactivar SELinux de manera permanente, cambie la variable **SELINUX** en el archivo **/etc/selinux/config** a **disabled**:

```
SELINUX:disabled
```

Para desactivar SELinux de manera temporal (modo permisivo), sin reiniciar, use el comando **setenforce** con la opción **0**; utilice **1** para activarlo nuevamente (modo de imposición). También utilice los términos **permissive** o **enforcing** en los argumentos, en lugar de **0** o **1**. Primero debe tener la función **sysadm_r**, que obtiene al iniciar sesión como usuario **root**.

```
setenforce 1
```

Revisión del estado y las estadísticas

Para revisar el estado actual de su sistema SELinux, emplee **sestatus**. Al agregar la opción **-v** también desplegará proceso y contextos de archivos, como se muestra en la lista de **/etc/sestatus.conf**. Los contextos especificarán funciones y tipos asignados a un proceso, archivo o directorio particular.

```
sestatus -v
```

Utilice el comando **seinfo** para desplegar sus estadísticas SELinux actuales:

```
#seinfo
Statistics for policy file: /etc/selinux/targeted/policy/policy.21
Policy Version & Type: v.21 (binary, MLS)
```

Classes:	55	Permissions:	206
Types:	1043	Attributes:	85
Booleans:	135	Cond. Expr.:	138
Sensitivities:	1	Categories:	256
Allow:	46050	Neverallow:	0
Auditallow:	97	Dontaudit:	3465
Role Allow:	5	Role Trans:	0
Type_trans:	987	Type_change:	14

Type_member:	0	Range_trans:	10
Fs_use:	12	Genfscon:	52
Portcon:	190	Netifcon:	0
Nodecon:	8	Initial SIDs:	0

Revisión del contexto de seguridad

La opción **-z**, usada con los comandos **ls**, **id** y **ps** se utiliza para revisar el contexto de seguridad de los archivos, usuarios y procesos, respectivamente. El contexto de seguridad indica las funciones que los usuarios deben tener para acceder procesos u objetos dados.

```
ls -lZ
id -Z
ps -eZ
```

Herramientas de administración de SELinux

SELinux ofrece varias herramientas para administrar su configuración e implementar directivas de SELinux, incluido semanage para configurar su directiva. La colección setools proporciona herramientas de configuración y análisis de SELinux incluido **apol**, la herramienta Security Policy Analysis, para análisis de transición de dominio, **sediffx** para diferencias entre directivas y **seaudit** para examinar los registros de auditd (véase la tabla 17-2). Las herramientas de administración de usuario de línea de comandos, **useradd**, **usermod** y **userdel**, todas tienen opciones SELinux y pueden aplicarse cuando SELinux se instala. Además, la herramienta **audit2allow** convertirá mensajes de negación de SELinux en módulos de directivas que permitirán el acceso.

Comando	Descripción
seinfo	Despliega estadísticas de directivas.
sestatus	Revisa el estado de SELinux en su sistema, incluidos los contextos de procesos y archivos.
sesearch	Busca reglas y directivas para imposición de tipo.
seaudit	Examina los archivos de registro de SELinux.
sediffx	Examina las diferencias entre directivas de SELinux.
autid2allow	Genera directivas que permiten reglas para módulos al utilizar mensajes de negación AVC audit.
apol	La herramienta SELinux Policy Analysis.
checkpolicy	El compilador de directivas de SELinux.
fixfiles	Revisa sistemas de archivos y establece contextos de seguridad.
restorecon	Establece características de seguridad para archivos particulares.
newrole	Asigna una nueva función.
setfiles	Establece contextos de seguridad para archivos.
chcon	Cambia el contexto.
chsid	Cambia el ID de seguridad.

TABLA 17-2 Herramientas de SELinux

Con la versión modular de SELinux, la administración de directivas ya no se maneja mediante edición directa de archivos de configuración. En cambio, use las herramientas de administración SELinux, como la herramienta de línea de comandos **semanage**. Tales herramientas usan los archivos de interfaz para generar directivas cambiadas.

semanage

semanage cambia su configuración SELinux sin editar directamente los archivos fuente de SELinux. Cubre casi todas las categorías principales incluidos usuarios, puertos, contextos de archivo e inicios de sesión. Consulte las páginas Man de **semanage** para conocer descripciones detalladas. Las opciones permiten modificar características de seguridad específicas como **-s** para nombre de usuario, **-R** para la función, **-t** para el tipo y **-r** para el rango de seguridad MLS. En el siguiente ejemplo agrega un usuario con la función `user_r`.

```
semanage user -a -R user_r julio
```

semanage se configura con el archivo `/etc/selinux/semanage.conf`, donde configura **semanage** para escribir en módulos directamente (el predeterminado) o para trabajar en el origen.

La herramienta Security Policy Analysis: apol

La herramienta Security Policy Analysis de SELinux, **apol**, ofrece un análisis complejo y detallado de una directiva seleccionada. Seleccione la entrada **apol** en el menú Administración, para iniciarlo.

Revisión de mensajes SELinux: seaudit

Ahora los mensajes AVC de SELinux se guardan en el archivo `/var/log/audit/audit.log`. Son especialmente importantes si usa el modo permisivo para probar una directiva que quiere imponer después. Debe saber si se le niega el acceso cuando es apropiado y si permite control cuando se necesita. Para ver sólo el mensaje de SELinux, utilice la herramienta **seaudit**. Los mensajes de inicio para el servicio SELinux todavía se registran en `/var/log/messages`.

Permiso de acceso: chcon y audit2allow

Siempre que SELinux niega acceso a un archivo o aplicación, el kernel envía una noticia de AVC. En muchos casos, el problema puede arreglarse con sólo cambiar el nombre del contexto de seguridad de un archivo para permitir el acceso. Utilice el comando **chcon** para cambiar un contexto de seguridad del archivo. En este cambio, necesita el acceso al servidor Samba, al archivo `log.richard3` y el directorio `/var/lib/samba`.

```
chcon -R -t samba_share_t log.richard3
```

Problemas más complicados, sobre todo los desconocidos, tal vez requieran crear una nueva directiva de módulo, usando los mensajes de AVC en el registro audit. Para esto, utilice el comando **audit2allows**. Este comando tomará mensajes de AVC de `audit` y generará comandos para acceder SELinux. El registro de audit es `/var/log/audit/audit.log`. Este registro envía su salida a **audit2allow**, que después causa que su opción **-M** cree un módulo de directiva.

```
cat /var/log/audit/audit.log | audit2allow -M local
```

Luego utilice el comando **semodule** para cargar el módulo:

```
semodule -i local.pp
```

Si primero quiere editar las entradas permitidas, utilice lo siguiente para crear un archivo **.te** del módulo local, **local.te**, que después puede editar:

```
audit2allow -m local -i /var/log/audit/audit.log > local.te
```

Una vez haya editado el archivo **.te**, use **checkmodule** para compilar el módulo y después **semodule_package** para crear el módulo de directiva, **local.pp**. Después puede instalarlo en el módulo **semodule**. Primero cree un archivo **.mod** con **checkmodule** y después un archivo **.pp** con **semodule_package**.

```
checkmodule -M -m -o local.mod local.te
semodule_package -o local.pp -m local.mod
```

```
semodule -i local.pp
```

En este ejemplo el módulo de directiva se llama **local**. Si quiere crear un nuevo módulo con **audit2allow**, debe usar un nombre diferente o adjuntar la salida al archivo **.te** con la opción **-o**.

SUGERENCIA En las distribuciones de Red Hat y Fedora, utilice el SELinux Troubleshooter para detectar problemas de acceso a SELinux.

Directiva de referencia de SELinux

Un sistema se asegura mediante una directiva. SELinux ahora utiliza una sola directiva, la de referencia, en lugar de dos directivas dirigidas y estrictas utilizadas en ediciones previas (visite serefpolicy.sourceforge.net). En lugar de dar a los usuarios sólo dos opciones, estricta y orientada, el proyecto de directiva de referencia de SELinux se concentra en proporcionar una directiva básica adaptable y fácilmente expansible, según necesite. La directiva de referencia de SELinux configura SELinux en módulos que pueden administrarse por separado. Todavía tiene las directivas estricta y orientada, pero son variaciones de la directiva de referencia básica. Además, tiene la directiva MLS para Multi-Level Security. La directiva orientada se instala como opción predeterminada, y puede instalar la directiva estricta o MLS.

En algunas distribuciones, como Fedora, tal vez ya tenga configuraciones de directivas separadas. Por ejemplo, Fedora proporciona tres directivas efectivas: orientada, estricta y mls. La directiva orientada se usa para controlar servidores específicos; por ejemplo, servidores de red e Internet, como servidores Web, DNS y FTP. También controla servicios locales con conexiones de red. La directiva no afectará sólo al daemon, sino todos los recursos empleados en el sistema.

La directiva estricta proporciona control completo sobre su sistema. Es bajo este tipo de directiva que sus usuarios, incluso administradores, pueden quedar bloqueados en el sistema inadvertidamente. Es necesario probar con cuidado una directiva estricta, para asegurarse de negar el acceso o permitirlo cuando sea apropiado.

Habrán subdirectorios **targeted**, **strict** y **mls** en su directorio **/etc/selinux**, pero ahora con un directorio **modules**. Es aquí donde puede encontrar sus configuraciones de SELinux.

Multi-Level Security (MLS)

Multi-Level Security (MLS) añade un método de seguridad de acceso más refinado. Agrega un valor de nivel de seguridad a los recursos. Sólo los usuarios con acceso a ciertos niveles acceden archivos y aplicaciones correspondientes. Dentro de cada nivel, el acceso se controla aún más con el uso de categorías. Las categorías funcionan de manera muy similar a los grupos, permitiendo el acceso sólo a usuarios autorizados para esa categoría. El acceso se vuelve más refinado, en vez de presentarse una situación de todo o nada.

Multi-Category Security (MCS)

Multi-Category Security (MCS) extiende SELinux para ser usado no sólo por administradores, sino usuarios. Éstos configuran categorías que restringen y controlan el acceso a sus archivos y aplicaciones. Aunque está basado en MLS, sólo utiliza categorías, no niveles de seguridad. Los usuarios seleccionan una categoría para un archivo, pero sólo el administrador puede crear una categoría y determinar qué usuarios tienen acceso a ella. Aunque es similar en concepto a las listas de control de acceso (ACL), sólo difiere en uso de la estructura de seguridad SELinux, brindando implementación de control en el nivel usuario impuesto por SELinux.

Métodos de directiva

Los servicios y componentes de un sistema operativo se clasifican en categorías en SELinux por tipo y función. Las reglas controlando estos objetos pueden basarse en tipos o funciones. Las directivas se implementan al utilizar dos tipos diferentes de reglas, imposición de tipo (TE) y control de acceso basado en funciones (RBAC). Multi-Level Security (MLS) es un método adicional restringiendo más el acceso por nivel de seguridad. El contexto de seguridad ahora presenta la función de un objeto, como usuario y nivel de seguridad de ese objeto.

Imposición de tipo

Con una estructura de tipo, los recursos del sistema operativo se particionan en tipos y a cada objeto se le asigna un tipo. Los procedimientos se asignan a dominios. Los usuarios se restringen a ciertos dominios y sólo se les permite utilizar objetos accesibles en estos dominios.

Control de acceso basado en funciones

Un método basado en función se enfoca en controlar los usuarios. A éstos se les asignan funciones definiendo los recursos usados. En un sistema estándar, los permisos de archivos, como los otorgados a grupos, controlan el acceso de los usuarios a archivos y directorios. Sin funciones, los permisos se vuelven más flexibles y refinados. Ciertos usuarios pueden acceder a más servicios que otros.

Usuarios de SELinux

Los usuarios retendrán el permiso disponible en un sistema estándar. Además, SELinux configura controles propios para un usuario dado, definiendo una función para ese usuario. Entre las identidades de seguridad general creadas por SELinux se incluyen:

- **system_u** El usuario para los procesos del sistema
- **user_u** Para permitir que los usuarios normales utilicen un servicio
- **root** Para el usuario root

Archivos de directiva

Las directivas se implementan en archivos de directiva. Se trata de archivos binarios compilados a partir de archivos fuente. En el caso de un archivo de directiva orientada preconfigurado, los archivos binarios de directiva están en los subdirectorios de directivas, en el directorio de configuración `/etc/selinux`, `/etc/selinux/targeted`. Por ejemplo, el archivo de directiva para la directiva orientada es:

```
/etc/selinux/targeted/policy/policy/.20
```

Los archivos de desarrollo orientados almacenando los archivos de interfaz están instalados en `/usr/share/selinux`.

```
/usr/share/selinux/targeted
```

Use los archivos de desarrollo para crear sus propios módulos que después podrá cargar.

Configuración de SELinux

La configuración de las opciones del servidor SELinux se realiza en el archivo `/etc/selinux/config`. Actualmente en el archivo se crean sólo dos configuraciones: estado y directiva. Configure la variable `SELINUX` para el estado, como `enforcing` o `permissive` y la variable `SELINUXTYPE` para el tipo de variable requerido para la directiva. Éstas corresponden a las opciones de SELinux en el nivel configuración de seguridad para deshabilitar e imponer, además de la directiva usada, como orientada (el nombre `targeted` puede ser ligeramente distinto en distribuciones diferentes, como `refpolicy-targeted` utilizado en Debian).

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - SELinux is fully disabled.
SELINUX=permissive
# SELINUXTYPE= type of policy in use. Possible values are:
#   targeted - Only targeted network daemons are protected.
#   strict - Full SELinux protection.
SELINUXTYPE=targeted
```

Las reglas de las directivas de SELinux

Las reglas de directivas pueden ser de cualquier tipo (imposición de tipo o TE) o RBAC (control de acceso basado en funciones), junto con niveles de seguridad (Multi-Level Security). Un tipo de expresión puede ser tipo, atributo de declaración o transición, cambio o regla de afirmación. Las instrucciones RBAC pueden ser declaraciones de función o dominancia, así como permitir funciones. Un nivel de seguridad especifica un número correspondiente al nivel de acceso permitido. La configuración de directiva es difícil, al usar reglas extensas y complicadas. Por esta razón, muchas reglas se implementan mediante macros M4 en archivos `fi`, que a cambio generarán las reglas apropiadas (Sendmail utiliza macros M4 de una forma similar). Encontrará estas reglas en archivos del paquete de código fuente de las directivas de referencia de SELinux que necesita descargar e instalar.

Declaraciones de tipo y función

Una declaración de tipo comienza con la palabra clave **type**, seguida por el nombre del tipo (identificador) y cualquier atributo o alias opcional. El nombre de tipo tendrá un sufijo `a_t`. En el caso de un objeto, como archivos, se incluyen definiciones de tipo estándar. El siguiente es un tipo predeterminado para cualquier archivo, con atributos `file_type` y `sysadmfile`:

```
type file_t, file_type, sysadmfile;
```

`root` tendrá su propia declaración de tipo:

```
type root_t, file_type, sysadmfile;
```

Los directorios especializados, como `boot`, también tendrán su propio tipo:

```
type boot_t, file_type, sysadmfile;
```

Reglas más especializadas se configuran en destinos específicos como el servidor Amanda. En el siguiente ejemplo se presenta una definición general de tipo para objetos de `amanda_t`, como los utilizados por el servidor de respaldo Amanda, como se muestra en la lista en el archivo **src/program/amanda.te** de la directiva orientada:

```
type amanda_t, domain, privlog, auth, nscd_client_domain;
```

Una declaración de función determina las funciones que acceden objetos de cierto tipo. Estas reglas comienzan con la palabra clave **role** seguida por la función y objetos asociados con esa función. En este ejemplo, un usuario o proceso con la función de sistema (`system_r`) tiene acceso a los objetos de `amanda` (`amanda_t`):

```
role system_r types amanda_t;
```

Se proporciona un tipo de declaración específico para ejecutables, como el siguiente para el servidor Amanda (`amanda_exec_t`). Esto define el ejecutable Amanda como un archivo ejecutable controlado por la administración del sistema.

```
type amanda_exec_t, file_type, sysadmfile, exec_type;
```

Los archivos de configuración asociados, a menudo tienen sus propias reglas:

```
type amanda_config_t, file_type, sysadmfile;
```

En la directiva orientada, se crea un tipo general ilimitado al que acceden las funciones de usuario y sistema, dando un apertura completa sin restricciones a todo el sistema. Reglas más específicas restringirán el acceso a ciertos objetivos, como el servidor Web.

```
type unconfined_t, domain, privuser, privhome, privrole, privowner, admin,
auth_write, fs_domain, privmem;
role system_r types unconfined_t;
role user_r types unconfined_t;
role sysadm_r types unconfined_t;
```

Los tipos también se configuran para archivos creados en el directorio `home` del usuario:

```
type user_home_t, file_type, sysadmfile, home_type;
type user_home_dir_t, file_type, sysadmfile, home_dir_type;
```

Contextos de archivo

Los contextos de archivo asocian archivos específicos con contextos de seguridad. El archivo o archivos se muestran primero en una lista, que varios archivos se representan con expresiones regulares. Después, se especifican función, tipo y nivel de seguridad. En el siguiente ejemplo crea un contexto de seguridad para todos los archivos del directorio **/etc** (archivos de configuración). El usuario del sistema (**system_u**) tiene acceso a éstos y son objetos de tipo **etc_t**, con un nivel de seguridad **0, s0**.

```
/etc(/.*)?                system_u:object_r:etc_t:s0
```

Ciertos archivos pertenecen a otros tipos; en realidad, el archivo de configuración **resolv.conf** pertenece al tipo **net_conf**:

```
/etc/resolv\*.conf.*      --      system_u:object_r:amanda_config_t:s0
```

Certain services (Ojo falta traducción)

```
/etc/amanda(/.*)?        system_u:object_r:amanda_config_t:s0
```

Los contextos de archivo se localizan en el archivo **file_contexts**, en el directorio de contexto de la directiva, como **/etc/selinux/targeted/contexts/files/file_contexts**. La versión usada para crear o modificar la directiva, se localiza en el directorio activo de módulos de directiva, como en **targeted/modules/active/file_contexts**.

Funciones de usuario

Las funciones de usuario definen qué funciones puede tomar un usuario. Dicha función comienza con la palabra clave **user**, seguida por el nombre del usuario, después la palabra clave **roles** y finalmente las funciones usadas. Encontrará esas reglas en los archivos de código fuente de la directiva de referencia. En el siguiente ejemplo se muestra una definición del usuario **system_u**:

```
user system_u roles system_r;
```

Si un usuario puede tener varias funciones, entonces se muestran en una lista entre llaves. La siguiente es la definición de la función de usuario estándar en la directiva orientada, que permite a los usuarios tomar funciones de administración de sistema:

```
user user_u roles { user_r sysadm_r system_r };
```

La directiva estricta sólo se muestra en la lista la función **user_r**:

```
user user_u { user_r };
```

Las reglas de vector de acceso: allow

Las reglas de vector de acceso se utilizan para definir permisos para objetos y procesos. La palabra clave **allow** es seguida por el objeto o tipo de proceso y después los tipos que tienen acceso o a los que se tiene acceso, además del permiso utilizado. En el siguiente ejemplo se permite que los procesos del dominio **amanda_t** busquen en los directorios de configuración de Amanda (cualquier directorio del tipo **amanda_config_t**):

```
allow amanda_t amanda_config_t:dir search;
```

En el siguiente ejemplo se permite que Amanda lea los archivos en un directorio home de usuario:

```
allow amanda_t user_home_type:file { getattr read };
```

En el siguiente ejemplo se permite que Amanda lea, busque y escriba archivos en los directorios Amanda:

```
allow amanda_t amanda_data_t:dir { read search write };
```

Reglas de permisos de función

Las funciones también tienen reglas de permisos. Aunque se usan para dominios y objetos, suelen emplearse para controlar transiciones de función, al especificar si una función pasa a otra. Estas reglas se muestran en una lista en el archivo de configuración RBAC. La siguiente entrada permite al usuario transitar a una función de administrador de sistema:

```
allow user_r sysadm_r;
```

Macros de transición y regla de vector

Las reglas del tipo de transición establecen el tipo utilizado para reglas de creación de objetos. Las reglas de transición también requieren reglas de vector de acceso correspondientes, a fin de habilitar los permisos para objetos o procesos. En vez de crear reglas separadas, se utilizan macros que generarán las reglas necesarias. En el siguiente ejemplo se establecen las reglas de transición y acceso para archivos de usuarios en el directorio home, al utilizar la macro `file_type_auto_trans`:

```
file_type_auto_trans(privhome, user_home_dir_t, user_home_t)
```

En el siguiente ejemplo se configura el proceso de transición y reglas de acceso de Amanda para crear procesos:

```
domain_auto_trans(inetd_t, amanda_inetd_exec_t, amanda_t)
```

Reglas de restricción

Las restricciones se colocan más adelante en procesos como transiciones para certificar mayor seguridad. Éstas se implementan con definiciones en el archivo de restricciones. Las reglas de restricciones a menudo aplican en operaciones de transición, como imponer que, en un proceso de transición, las identidades de usuario permanezcan iguales, o el proceso 1 sea un dominio que tiene el atributo `privuser` y el proceso 2 sea un dominio con el atributo `userdomain`. Los caracteres `u`, `t` y `r` se refieren a usuario, tipo y función, respectivamente.

```
constrain process transition
( u1 == u2 or ( t1 == privuser y t2 == userdomain )
```

Archivos de configuración de directiva de SELinux

Los archivos de configuración suelen cambiar al utilizar archivos `.te` y `.fc`. Estos están ausentes de los encabezados de módulos en `/usr/share/selinux`. Si agrega un módulo, necesitará crear los archivos `.te` y `.fc` para éste. Después, puede crear un módulo y agregarlo como se describe en la siguiente sección. Si quiere crear o modificar su propia directiva, necesitará descargar e instalar los archivos de código fuente para la directiva de referencia de SELinux, como se describe en la sección

“Uso de la configuración fuente de SELinux”. El código de la directiva de referencia almacena un conjunto completo de archivos de configuración **.te** y **.fc**.

Compilación de módulos SELinux

En lugar de compilar todo el código fuente cada vez que quiera hacer un cambio, sólo compile el módulo del área cambiada. El directorio de módulos almacena diferentes módulos. Cada uno construido a partir de un archivo **.te** correspondiente. El comando **checkmodule** se utiliza para crear un archivo de módulo **.mod** a partir del archivo **.te**, después el comando **semodule_package** se emplea para crear el archivo de módulo **.pp** a cargar, además del archivo para contexto de archivo **.fc**. Como se observó en la documentación de SELinux, si necesita cambiar la configuración de **syslogd**, primero utilice lo siguiente para crear un archivo **syslogd.mod** con **syslogd.te**. La opción **-M** especifica el soporte para niveles de seguridad MLS.

```
checkmodule -M -m syslogd.te -o syslogd.mod
```

Después, utilice el comando **semodule_package** para crear un archivo **syslogd.pp** a partir del archivo **syslogd.mod**. La opción **-f** especifica el archivo para contexto de archivo.

```
semodule_package -m syslogd.mod -o syslogd.pp -f syslogd.fc
```

Para agregar el módulo emplee **semodule** y la opción **-i**. Revise si un módulo está cargado con la opción **-l**.

```
semodule -i syslogd.pp
```

Los cambios a la directiva base se hacen en el archivo **policy.conf**, compilada en el módulo **base.pp**.

Uso de la configuración fuente de SELinux

Para implementar su propia configuración, tendrá que descargar e instalar el archivo de código fuente para la directiva de referencia de SELinux. En el caso de distribuciones RPM, éste será el archivo SRPMS. Los archivos **.te** utilizados para configurar SELinux ya no son parte de los paquetes binarios de SELinux.

NOTA En las distribuciones Red Hat o Fedora, el archivo comprimido de la fuente, un archivo **tgz**, junto con otros de configuración de directiva, se instalarán en **/usr/src/redhat/SOURCES**. (Asegúrese de tener **rpm-build** instalado; no se instala de manera predeterminada). Utilice una operación **rpmbuild** con el archivo **security-policy.spec** para extraer el archivo en el directorio **serefpolicy** en **/usr/src/redhat/BUILD**.

Vaya al directorio **seref-policy** y ejecute el siguiente comando para instalar el archivo fuente de SELinux en **/etc/selinux/serefpolicy/src**.

```
make install-src
```

Las reglas se almacenan en archivos de configuración, localizados en subdirectorios del directorio de la directiva **src**. Dentro de este encontrará un subdirectorio **policy/modules**. Allí, organizados en varios directorios, como **admin** y **apps**, encontrará los archivos de configuración **.tc**, **.fc** y **.if**.

Tendrá los archivos de configuración para imposición de tipo y contextos de seguridad. Los archivos de imposición de tipo tienen la extensión `.te`, mientras los contextos de seguridad tienen una extensión `.sc`.

Para reflejar el control tan fino que proporciona SELinux, cuenta con varios archivos de configuración de módulos para muchos tipos de objetos y procesos en su sistema. Los principales archivos y directorios de configuración se muestran en una lista en la tabla 17-3, pero varios se expanden para mostrar listas detalladas de archivos y directorios.

Archivos de interfaz

Los archivos *interfaz* del archivo permiten administrar herramientas para generar módulos de directivas. Éstos definen macros de interfaz para su directiva actual. El archivo fuente de SELinux, `refpolicy`, almacenará archivos `.if` para cada módulos, junto con los archivos `.te` y `.fc`. También, los archivos `.if` del directorio `/usr/share/selinux/devel` se utilizan para generar módulos.

Directorios y archivos	Descripción
<code>assert.te</code>	Aserciones de vector de acceso
<code>config/appconfig*</code>	Archivos de configuración para el motor en tiempo de ejecución de una aplicación
<code>policy/booleans.conf</code>	Características que se pueden ajustar
<code>file_contexts</code>	Contextos de seguridad para archivos y directorios
<code>policy/flask</code>	Configuración de Flask
<code>policy/mcs</code>	Configuración de Multi-Category Security (MCS)
<code>doc</code>	Soporte a documentación de directivas
<code>policy/modules</code>	Módulos de directiva de seguridad
<code>policy/modules.conf</code>	Lista y uso de módulo
<code>policy/modules/admin</code>	Módulos de administración
<code>policy/modules/apps</code>	Módulos de aplicación
<code>policy/modules/kernel</code>	Módulos de kernel
<code>policy/modules/services</code>	Módulos de servicios y servidor
<code>policy/modules/system</code>	Módulos de sistema
<code>policy/rolemap</code>	Tipos y funciones de dominio de usuario
<code>policy/users</code>	Definición general de usuarios
<code>config/local.users</code>	Sus propios usuarios de SELinux
<code>policy/constraints</code>	Restricciones adicionales para transición de función y acceso de objetos
<code>policygentool</code>	Secuencia de comandos para generar directivas
<code>policy/global_tunables</code>	Directivas que se pueden ajustar para personalización
<code>policy/mls</code>	Configuración de Multi-level Security (MLS)

TABLA 17-3 Archivos de configuración de directivas de SELinux

Archivos de tipos

En la directiva orientada, el directorio de módulos que define los tipos contiene varios archivos, incluidos los de configuración **nfs.te** y **network.te**. Aquí encontrará declaraciones de tipo para diferentes tipos de objetos en su sistema. Los archivos **.te** ya no se incluyen en su instalación de SELinux estándar. En cambio, debe descargar e instalar el paquete fuente **serefpolicy**. Se trata del código fuente original y permite reconfigurar completamente su directiva SELinux, en lugar de administrar módulos con herramientas de administración como **semanage**. El directorio de módulos almacenará archivos **.te** para cada módulo, mostrando una lista de sus reglas TE.

Archivos de módulo

Los módulos se localizan entre directorios en **policy/modules**. Aquí encontrará tres archivos correspondientes para cada aplicación o servicio. Habrá un archivo **.te** conteniendo las reglas reales de imposición de tipo, uno **.if**, para la interfaz (un archivo permitiendo que otras aplicaciones interactúen con el módulo) y varios archivos **.fc** que definen los contextos de archivo.

Archivos de contexto de seguridad

Los contextos de seguridad para diferentes archivos se detallan en los archivos de contexto de seguridad. El archivo **file_contexts** almacena configuraciones de contexto de seguridad para diferentes grupos, directorios y archivos. Cada archivo de configuración tiene una extensión **.fc**. El archivo **types.fc** almacena contextos de seguridad para varios archivos y directorios del sistema, sobre todo, el acceso a archivos de configuración en el directorio **/etc**. En la fuente de SELinux, cada módulo tendrá su propio archivo **.fc**, junto con archivos **.te** e **if** correspondientes. El archivo **distros.fc** define las configuraciones dependientes de la distribución. El archivo **homedir_template** define contextos de seguridad para archivos tipo “punto” que pueden estar configurados en un directorio home del usuario, como **.mozilla**, **.gconf** y **.java**.

Un directorio de módulos tiene archivos de contexto de archivo para aplicaciones y servicios particulares. Por ejemplo, **apache.fc** tiene contextos de seguridad para todos los archivos y directorios utilizados por el servidor Web Apache, como **/var/www** y **/etc/httpd**.

Configuración de usuario: funciones

La configuración de usuario global está definida en el archivo **users** del directorio de la directiva. Aquí encontrará definiciones de usuario y funciones para usuarios (**user_u**) y administradores (**admin_u**) estándar. Para agregar sus propios usuarios, utilice el archivo **local-users**. Aquí encontrará ejemplos para insertar sus propios usuarios de SELinux. Las directivas estricta y orientada recurren a identidades para usuarios de SELinux **user_u** generales. Para configurar una identidad de SELinux separada para un usuario, defina ese usuario en el archivo **local.users**.

El archivo **rbac** define funciones permitidas a las que una función puede hacer la transición. Por ejemplo, ¿puede una función de usuario transitar a una de administración de sistema? La directiva orientada tiene varias entradas permitiendo a un usuario transformarse de manera libre en administrador y viceversa. La directiva estricta no tiene esas definiciones.

Las transiciones de función tienen mayores restricciones debido a las reglas del archivo **constraints**. Aquí, controla el cambio a otros usuarios y restringe el cambio a contextos de seguridad de objeto (etiquetado).

Herramientas de módulos de directivas

Para crear un módulo de directiva y cargarlo, use varias herramientas. En primer lugar, el comando **checkmodule** se utiliza para crear el archivo **.mod** a partir de un archivo **.te**. Después, la herramienta

semodule_package toma el archivo **.mod** y cualquier archivo **.fc** de soporte y genera un archivo de paquete de directiva de módulo, **.pp**. Por último, la herramienta **semodule** toma el archivo de paquete de la directiva y lo instala como parte de su directiva de SELinux.

Configuración de aplicaciones: appdconfig

Ciertos servicios y aplicaciones son conscientes de la seguridad y pedirán contextos de seguridad y tipos predeterminados a SELinux (consulte también la sección “Contextos de seguridad y tipos en motores en tiempo de ejecución: contextos”). La configuración se almacena en archivos ubicados en el directorio **policy/config/appconfig-***. El archivo **default_types** almacena opciones predeterminadas de tipos; **default_contexts** aloja contextos de seguridad predeterminados. El archivo **initrc_context** tiene el contexto de seguridad predeterminado para ejecutar secuencias de comandos **/etc/rc.d**. Un archivo **root_default_context** predeterminado muestra detalles de la manera en que accede al usuario root. El archivo **removable_context** almacena contextos de seguridad para dispositivos extraíbles y **media** muestra una lista de dispositivos de medios, como cdrom para CD-ROM. Los valores del motor en tiempo de ejecución también se insertan en archivos correspondientes, en el directorio de contextos de directiva, como **/etc/selinux/targeted/contexts**.

Creación de una directiva de SELinux: make y checkpolicy

Si quiere crear una nueva directiva completa, use el código fuente de la directiva de referencia SELinux, **/etc/selinux/serefpolicy**. Una vez que ha configurado su directiva, créela con los comandos **make policy** y **checkpolicy**. El primer comando genera un archivo **policy.conf** para sus archivos de configuración, que después usará **checkpolicy** para generar un archivo binario de directiva. Este tipo de archivo se creará en el subdirectorio **policy**, con una extensión numérica para la versión de la directiva, como **policy.20**.

Deberá generar un nuevo archivo **policy.conf**. Para esto, inserte el siguiente comando en el directorio **src** de la directiva, que será **/etc/selinux/serefpolicy/src/policy**.

```
make policy
```

Después utilice **checkpolicy** para crear la nueva directiva.

En vez de compilar todo el código fuente, cada vez que quiera crear un cambio, sólo compile un módulo para el área que cambió. (En la versión previa de SELinux, siempre tenía que recopilar toda la directiva, cada vez que hacía un cambio.) El directorio de módulos almacena diferentes módulos. Cada uno se construye a partir de un archivo **.te** correspondiente. El comando **checkmodule** se usa para crear un archivo de módulo **.mod** a partir del archivo **.te**, después el comando **semanage_module** se utiliza para crear un archivo de módulo **.pp** de paquete de directiva que puede cargarse. Como se observa en la documentación SELinux, si necesita cambiar la configuración de **syslogd**, primero usaría la siguiente línea para crear un archivo **syslogd.mod** con **syslogd.te**. La opción **-M** especifica el soporte para niveles de seguridad MLS.

```
checkmodule -M -m syslogd.te -o syslogd.mod
```

Después emplee el comando **semanage_module** para crear un archivo **syslogd.pp**, a partir del archivo **syslogd.mod**. La opción **-f** especifica el archivo de contexto de archivo.

```
semanage_module -m syslogd.mod -o syslogd.pp -f syslogd.fc
```

Para agregar el módulo, use **semodule** y la opción **-i**. Puede revisar si un módulo está cargado con la opción **-l**.

```
semodule -i syslogd.pp
```


Los cambios a la directiva base se hacen en el archivo **policy.conf**, compilada en el módulo **base.pp**.

Para realizar sus propias configuraciones, ahora debe descargar los archivos de código fuente. Los archivos **.te** empleados para configurar SELinux ya no son parte de los paquetes binarios SELinux. Una vez instalado, el código fuente estará en el directorio **safepolicy** en **/etc/selinux**.

SELinux: operaciones administrativas

Existen varias tareas que puede realizar en su sistema SELinux sin recompilar toda la configuración. Los contextos de seguridad para ciertos archivos y directorios pueden cambiarse conforme sea necesario. Por ejemplo, cuando agrega un nuevo sistema de archivos, necesita etiquetarlo con los contextos de seguridad apropiados. Además, cuando agrega usuarios, tal vez deba hacer que el sistema preste atención especial al usuario.

Uso de contextos de seguridad: fixfiles, setfiles, restorecon y chcon

Hay varias herramientas disponibles para cambiar el contexto de seguridad de sus objetos. El comando **fixfiles** establece el contexto de seguridad para el sistema de archivos. Use la opción **relabel** para establecer contextos de seguridad y **check** para ver qué debe cambiarse. La herramienta **fixfiles** es una secuencia de comandos empleando **setfiles** y **restorecon** para hacer cambios reales.

El comando **restorecon** permitirá restaurar el contexto de seguridad para archivos y directorios, pero **setfiles** es una herramienta básica para configurar contextos de seguridad. Puede aplicarla en archivos individuales o directorios. Se utiliza para etiquetar el archivo cuando una directiva se instala por primera vez.

Con **chcon**, puede cambiar los permisos de archivos y directorios individuales, de manera muy similar a como **chmod** hace para permisos generales.

Adición de nuevos usuarios

Si un usuario nuevo no necesita acceso especial, generalmente sólo usa la identidad genérica **user_u** de SELinux. Sin embargo, si necesita permitir que el usuario tome funciones que de otra forma estarían restringidas, como una función de administrador de sistema en la directiva estricta, necesita configurar al usuario de acuerdo con esto. Para hacerlo, agregue el usuario al archivo **local.users** en el directorio de las directivas de usuarios, como **/etc/selinux/targeted/policy/users/local.users**. Observe que es diferente del archivo **local.users** en el directorio **src**, compilado directamente en la directiva. Las reglas de usuario tienen la sintaxis

```
user nombredeusuario roles { listadefunciones };
```

En el siguiente ejemplo se agrega la función **sysadm** al usuario **jorge**:

```
user jorge roles { user_r sysadm_r };
```

Una vez agrega la función, debe cargar la directiva de nuevo.

```
make reload
```

También puede administrar usuarios con el comando **semanage** y la opción **user**. Para ver cuáles usuarios están activos, haga una lista de éstos con el comando **semanage user** y la opción **-l**.


```
# semanage user -l
system_u: system_r
user_u: user_r sysadm_r system_r
root: user_r sysadm_r system_r
```

El comando **semanage user** tiene las opciones **a**, **d**, **m** para agregar, eliminar o cambiar usuarios, respectivamente. Las opciones **a** y **m** permiten especificar funciones que se agregarán a un usuario, mientras la opción **d** eliminará un usuario.

Contextos de seguridad y tipos en el motor en tiempo de ejecución: contextos

Varias aplicaciones y servicios están conscientes de la seguridad y necesitarán información de configuración de seguridad predeterminada, como los contextos de seguridad. Las configuraciones del motor en tiempo de ejecución para contextos de seguridad y tipos se almacenan en archivos localizados en el directorio de contextos de directivas, como **/etc/selinux/targeted/contexts**. Los archivos de tipos tendrán el sufijo **_types**, y los de contexto de seguridad usarán **_context**. Por ejemplo, el contexto de seguridad predeterminado para archivos extraíbles se ubica en el archivo **removable_context**. El contenido de ese archivo se muestra aquí:

```
system_u: object_r:removable_t
```

El archivo **default_context** se utiliza para asignar un contexto de seguridad predeterminado a aplicaciones. La directiva estricta se utiliza para controlar el acceso a la administración de sistema, proporcionándolo cuando se necesite; por ejemplo, durante el proceso de inicio de sesión.

En el siguiente ejemplo se establecen las funciones predeterminados para usuarios, en el proceso de inicio de sesión:

```
system_r:local_login_t user_r:user_t
```

Esto permite a los usuarios iniciar sesión como administradores o usuarios regulares.

```
system_r:local_login_t sysadm_r:sysadm_t user_r:user_t
```

El siguiente ejemplo es para inicios de sesión de usuarios remotos, para quienes la administración de sistema no se incluye:

```
system_r:remote_login_t user_r:user_t staff_r:staff_t
```

El archivo **default_types** define los tipos predeterminados para funciones. Estos archivos tienen entradas de función y tipo, cuando una transición toma lugar a una nueva función, se utiliza el tipo predeterminado que se especifica aquí. Por ejemplo, el tipo predeterminado para la función **sysadm_r** es **sysadm_t**.

```
sysadm_r:sysadm_t
user_r:user_t
```

El archivo **initrc_context** representa un interés particular, porque configura el contexto para la ejecución de secuencias de comandos del sistema en el directorio **/etc/rc.d**. En la directiva destino están abiertos a todos los usuarios.

```
user_u:system_r:unconfined_t
```

En la directiva estricta se limitan al usuario del sistema.

```
system_u:system_r:initrc_t
```

users

Es probable que deban configurarse los contextos de seguridad predeterminados para usuarios particulares como root. En el archivo **sesusers** encontrará una entrada root mostrando una lista de funciones, tipos y niveles de seguridad tomados por el usuario root, como en el siguiente ejemplo para su operación (en algunas distribuciones, puede ser el directorio **users** con archivos separados para los diferentes usuarios):

```
sysadm_r:sysadm_su_t sysadm_r:sysadm_t starff_r:staff_t user_r:user_t
```

context y files

Los contextos de seguridad predeterminados para sus archivos y directorios se localizan en el directorio **contexts/files**. El directorio **file_contexts** muestra una lista de contextos de seguridad predeterminados para todos sus archivos y directorios, como se determinó en su directiva. El directorio **file_context.homedirs** establece los contextos de archivo para archivos del directorio home, además del directorio raíz, incluidos archivos de configuración tipo “punto” como **.mozilla** y **.gconf**. El archivo media configura el contexto predeterminado para dispositivos multimedia como CD-ROM y discos.

```
cdrom system_u:object_r:removable_device_t
floppy system_u:object_r:removable_device_t
disk system_u:object_r:fixed_disk_device_t
```


IPsec y redes privadas virtuales

El protocolo de seguridad de Internet (IPsec, Internet Security Protocol), incorpora seguridad para transmisión de red, directamente en el protocolo de Internet (IP, Internet Protocol). IPsec está integrado en un nuevo protocolo IPv6 (Internet Protocol, versión 6). También se usa con el protocolo más antiguo IPv4. IPsec ofrece métodos para codificar datos y autenticar el host o red adonde se envía. El proceso puede administrarse manual o automáticamente mediante la herramienta de intercambio de claves **racoon** de IPsec. Con IPsec, el kernel detecta y descifra automáticamente transmisiones entrantes, además de cifrar las salientes. También puede utilizar IPsec para implementar redes privadas virtuales, cifrando los datos enviados por Internet de una red local a otra. Aunque IPsec es un método de seguridad relativamente nuevo, su integración en IP con el tiempo le proporcionará gran aceptación. Consulte el HOWTO de IPsec para conocer una explicación más detallada de la implementación IPsec en Linux, ipsec_howto.org.

Varios proyectos ofrecen desarrollo e implementación de herramientas IPsec (véase la tabla 18-1). El proyecto KAME (kame.net) contribuye con herramientas IPsec originales. Las versiones actuales se obtienen de ipsec-tools.sourceforge.net. Otros proyectos de herramientas IPsec incluyen el proyecto OpenSecure/Wide Area Network (Openswan) en openswan.org, ofreciendo implementación de herramientas IPsec en Linux, y VPN Consortium (VPNC) en vpnc.org, soportando versiones de Windows y Macintosh. La documentación se localizará en `/usr/doc/openswan-versión`. La documentación detallada se almacena en el paquete **openswan-doc**, que se instalará en `/usr/doc/openswan-doc-versión`.

Protocolos IPsec

IPsec está integrado por varios protocolos que proporcionan autenticación, cifrado y claves de cifrado de intercambio seguro. El protocolo de encabezado de autenticación (AH, Authentication Header) confirma que el paquete se envió por el emisor y no otra persona. IPsec también incluye una verificación de integridad detectando cualquier modificación en la transferencia. Los paquetes se cifran mediante carga de seguridad de encapsulado (ESP, Encapsulating Security Payload). Cifrado y descifrado se realizan mediante claves secretas compartidas por emisor y receptor. Estas claves se transmiten por sí solas a través del protocolo de intercambio de claves de Internet (IKE, Internet Key Exchange), que proporciona un intercambio seguro. El cifrado ESP degrada ciertos métodos de compresión de transmisión, como PPP, para conexiones a Internet por marcado

Sitio Web	Proyecto
kame.net	Proyecto KAME para herramientas IPsec
openswan.org	Proyecto Open Secure/Wide Area Network
vpnc.org	Consorcio VPN
ipsec-howto.org	Documentación HOWTO de IPsec
ipsec-tools.sourceforge.net	Herramientas y recursos de IPsec

TABLA 18-1 Recursos de IPsec

telefónico. Para acomodar estos métodos de compresión, IPsec brinda el protocolo de compresión de carga de IP (IPComp, IP Payload Compression Protocol), con el que se comprimen los paquetes antes de enviarse.

Autenticación de cifrado y verificaciones de integridad se incluyen usando códigos de autenticación con métodos de hash (HMAC, Hash Methods Authentication Codes) generados por métodos de seguridad hash como SHA2, mediante el uso de una clave secreta. HMAC se incluye en el encabezado IPsec, que el receptor revisa después con la clave secreta. El cifrado de datos transmitidos se realiza con métodos de cifrado simétrico como 3DES, Blowfish y DES.

Los protocolos AH, ESP e IPComp se incorporan en el kernel de Linux. El protocolo IKE se implementa como un daemon separado. Sólo ofrece una forma de compartir claves secretas y puede reemplazarse con otros métodos de intercambio.

Modos IPsec

Puede utilizar las capacidades de IPsec para transporte normal o entunelamiento de paquetes. Con el transporte normal, los paquetes se codifican y envían al siguiente destino. El modo de transporte normal se utiliza para implementar cifrado directo de host a host, donde cada host maneja el proceso de cifrado IPsec. El entunelamiento de paquetes se maneja para codificar transmisiones entre puertas de enlace, permitiendo que éstas manejen el proceso de cifrado IPsec para el tráfico dirigido a una red completa, o procedente de ésta, en lugar de configurar el cifrado IPsec para cada host. Con el entunelamiento de paquetes, los paquetes se encapsulan con nuevos encabezados para un destino específico, permitiéndole implementar redes privadas virtuales (VPN, Virtual Private Networks). Los paquetes se dirigen a puertas de enlace VPN, que cifran y envían paquetes de red local.

NOTA Tiene la opción de elegir que se cifren paquetes para ciertos hosts o los que pasan por puertos específicos.

Bases de datos de seguridad de IPsec

Los paquetes que decida cifrar son designados por la base de datos de directiva de seguridad (SPD, Security Policy Database), de IPsec. El método que use para cifrarlos es determinado por la Security Association Database (SAD, base de datos de la asociación de seguridad) de IPsec. SAD asocia el método de cifrado y clave con una conexión particular o tipo de conexión. Las conexiones cifradas se designan en base de datos de directiva de seguridad.

Herramienta	Descripción
plainrsa-gen	Genera una clave RSA simple.
setkey	Administra bases de datos de directiva (SPD) y de asociación (SAD).
racoon	Configura e implementa intercambios de clave seguros al utilizar IPsec Key Exchange (IKE, intercambio de claves de IPsec).
racoonctl	Administra conexiones IPsec.

TABLA 18-2 Herramientas IPsec

Herramientas IPsec

Se proporcionan varias herramientas IPsec para administración de conexiones IPsec (véase la tabla 18-2). Con **setkey**, administra bases de datos de directiva y asociación. La herramienta **racoon** configura el proceso de intercambio de claves para implementar intercambios de claves de descifrado entre conexiones. Para administrar sus conexiones IPsec, utilice **racoonctl**. Por ejemplo, la opción **show-sa** desplegará sus asociaciones de seguridad y **vpn-connect** establecerá una conexión VPN.

***NOTA** Para habilitar IPse en el kernel, asegúrese de habilitar las opciones PF_KEY, AH y ES en Opciones criptográficas.*

Configuración de conexiones con setkey

Para configurar sus conexiones IPsec, cuenta con la herramienta **setkey**. Contiene instrucciones para administrar reglas en las bases de datos de directiva y seguridad de IPsec. Utilice la instrucción **add**, para agregar una asociación de seguridad a la base de datos de seguridad (SAD) y la instrucción **spdadd**, para añadir una directiva a la de directivas (SPD). El término **ah** indica que la instrucción se aplica al encabezado de autenticación (AH) y **esp** indica que el cifrado es implementado por la carga de seguridad de cifrado (ESP). Para implementar operaciones **setkey**, es mejor usar una secuencia de comandos invocando a **setkey** con la opción **-f** y mostrar una lista de instrucciones de **setkey**. En el siguiente ejemplo se crea una secuencia de comandos simple para agregar instrucciones de autenticación y cifrado para una conexión particular, además de generar una directiva de seguridad para ésta:

```
#!/sbin/setkey -f
add 192.168.0.2 192.168.0.5 ah 15700 -A hmac-md5 "clave secreta";
add 192.168.0.2 192.168.0.5 esp 15701 -E 3des-cbc "clave secreta";
spdadd 192.168.0.2 192.168.0.5 any -P out ipsec
    esp/transport//require
    ah/transport//require;
```

Asociaciones de seguridad: SA

Las asociaciones de seguridad se utilizan para indicar que quiere cifrar el encabezado de autenticación y la carga de cifrado (ESP). Una conexión particular, como la que existe entre dos host, puede hacer que se cifren esos encabezados de autenticación de host empleando métodos de cifrado específicos y claves secretas designadas. Lo mismo se hace para la carga de cifrado,

el contenido principal de las transmisiones. Una clave secreta se determina de manera manual o automática al utilizar intercambios de claves. En el siguiente ejemplo se especifica eso para la conexión entre 192.168.0.2 y 192.168.0.5, el método de autenticación **hmac-md5** y se utilizará una clave secreta (aquí designada por el contenedor **clave secreta**) para el encabezado de autenticación, **ah**.

```
add 192.168.0.2 192.168.0.5 ah 15700 -A hmac-md5 "clave secreta";
```

La asociación de seguridad para encryption payload utiliza el método de cifrado 3des-cbc y una clave secreta diferente:

```
add 192.168.0.2 192.168.0.5 esp 15701 -E 3des-cbc "clave secreta";
```

Cada instrucción se identifica con un índice de parámetro de seguridad (SPI, Security Parameter Index), en este caso 15700 y 15701. En realidad, instrucciones idénticas con diferentes SPI se consideran instrucciones diferentes.

Tenga en cuenta que las asociaciones de seguridad sólo especifican procedimientos de cifrado posibles. No los implementan. Para eso necesita establecer directivas de seguridad.

Directiva de seguridad: SP

Una directiva de seguridad implementará un procedimiento de seguridad IPsec para una conexión. Designe un host o puerto de conexión. Una vez que una directiva se configura para una conexión, el kernel determinará qué asociaciones de seguridad aplicar, al utilizar una base de datos de SAD. Una directiva de seguridad se agrega con la instrucción **spdadd**. Se requiere cifrado, autenticación o ambas.

En el siguiente ejemplo, se codificarán y autenticarán las transmisiones entre los host 192.168.0.2 y 192.168.0.5. Cualquier transmisión saliente entre estos host se codificará y autenticará:

```
spdadd 192.168.0.2 192.168.0.5 any -P out ipsec esp/transport/require;
```

En la instrucción **spdadd**, necesitará especificar la conexión, como una entre los dos host o redes. Para dos host, utilice sus direcciones IP, en este ejemplo, 192.168.0.2 y 192.168.0.5. Después especifique el tipo de paquete y su dirección, en este caso cualquier paquete saliente, **any -P out**. Después especifique las instrucciones **ipsec** para el protocolo ESP o AH, o ambos. Para cada entrada, especifique el modo (transport o tunnel), los host involucrados (es diferente en modo tunnel) y la directiva para el cifrado, por lo general **require**. En este ejemplo se muestra que el protocolo ESP se utilizará para el modo transport para conexiones entre 192.168.0.2 y 192.168.0.5, y es obligatorio:

```
esp/transport/192.168.0.2-192.168.0.5/require
```

Puede dejar fuera la información de host, si es la misma, como en el ejemplo anterior.

```
esp/transport//require
```

Hosts receptores

Para que un host reciba y cifre una transmisión IPsec, debe tener instrucciones de asociación de seguridad correspondientes en su propia base de datos de SAD, que le indiquen cómo autenticar y descifrar las instrucciones recibidas. Las instrucciones de asociación de seguridad reflejan las

instrucciones del emisor, al utilizar el mismo método de cifrado, claves secretas e índices de seguridad. Aunque no se requiere una directiva correspondiente.

```
#!/sbin/setkey -f
add 192.168.0.2 192.168.0.5 ah 15700 -A hmac-md5 "clave secreta";
add 192.168.0.2 192.168.0.5 esp 15701 -E 3des-cbc "clave secreta";
```

Es probable que los hosts que reciben quieran configurar directivas para filtrar paquetes entrantes en conexiones seguras, descartando los que no estén cifrados. La siguiente directiva aceptará sólo las transmisiones cifradas y autenticadas IPsec de entrada de 192.168.0.2.

```
spdadd 192.168.0.2 192.168.0.5 any -P in ipsec esp/transport//require
ah/transport//require;
```

Transmisiones de dos vías

En el ejemplo anterior, se configuró una conexión segura entre dos host que iban de una dirección a otra, de 192.168.0.2 a 192.168.0.5, no en el otro sentido, de 192.168.0.5 a 192.168.0.2. Para implementar transmisiones seguras de dos vías entre dos host, ambos necesitan configurarse como emisor y receptor, con asociaciones de seguridad correspondientes que coincidan. Las siguientes secuencias de comandos están basadas en ejemplos comunes de una conexión IPsec de dos vías entre dos host. Configuran una conexión IPsec de dos vías seguras entre los host 192.168.0.2 y 192.168.0.5. También se incluyen las directivas entrantes correspondientes, pero no son necesarias.

Primero está la configuración del host 192.168.0.2:

```
#!/sbin/setkey -f
add 192.168.0.2 192.168.0.5 ah 15700 -A hmac-md5 "clave secreta";
add 192.168.0.5 192.168.0.2 ah 24500 -A hmac-md5 "clave secreta";

add 192.168.0.2 192.168.0.5 esp 15701 -E 3des-cbc "clave secreta";
add 192.168.0.5 192.168.0.2 esp 24501 -E 3des-cbc "clave secreta";

spdadd 192.168.0.2 192.168.0.5 any -P out ipsec esp/transport//require
ah/transport//require;
spdadd 192.168.0.5 192.168.0.2 any -P in ipsec esp/transport//require
ah/transport//require;
```

El host correspondiente, 192.168.0.5, utiliza las mismas instrucciones pero con las conexiones de red IP invertidas. Observe cómo corresponden los índices de seguridad para las instrucciones del emisor y el receptor en cada extremo:

```
#!/sbin/setkey -f
add 192.168.0.5 192.168.0.2 ah 15700 -A hmac-md5 "clave secreta";
add 192.168.0.2 192.168.0.5 ah 24500 -A hmac-md5 "clave secreta";

add 192.168.0.5 192.168.0.2 esp 15701 -E 3des-cbc "clave secreta";
add 192.168.0.2 192.168.0.5 esp 24501 -E 3des-cbc "clave secreta";

spdadd 192.168.0.5 192.168.0.2 any -P out ipsec esp/transport//require ah/
transport//require;
spdadd 192.168.0.2 192.168.0.5 any -P in ipsec esp/transport//require
ah/transport//require;
```


Configuración de IPsec con racoon: IKE

Las claves IPsec se implementan como claves manuales, compartidas o con certificados. Las claves manuales se intercambien de manera explícita y tienden a tener problemas de seguridad. Las claves compartidas y los certificados se administran al utilizar el protocolo de intercambio de claves de IPsec (IKE), que intercambiará claves de manera automática, cambiándolas de manera aleatoria para evitar la detección.

Una de las ventajas de utilizar IKE es que generará de manera automática cualquier asociación de seguridad necesaria, si no se proporciona ninguna. Esto significa que para configurar las conexiones de seguridad con IKE necesita especificar una directiva de seguridad, no las asociaciones de seguridad.

La herramienta **racoon** es un daemon de intercambio de clave para el protocolo IKE de IPsec. En el caso de claves compartidas, **racoon** autentica los hosts de manera dinámica al utilizar claves secretas compartidas previamente. Con los métodos de certificación, los hosts se autentifican al utilizar archivos de certificado. El archivo de configuración de **racoon** se localiza en **/etc/racoon/racoon.conf**. Aquí se configuran los parámetros generales. Utilice el archivo **racoon.conf** para la mayor parte de las conexiones.

La configuración de **racoon** consta de estrofas que contienen parámetros para conexiones posibles. Una configuración muy simple se muestra en el siguiente ejemplo, que utiliza una clave secreta simple compartida. La ubicación se especifica con la opción **path pre_shared_key**, en este caso **/etc/racoon/psk.txt**. Las claves de certificado, un método más seguro que utiliza claves públicas y privadas, se analiza más adelante.

```
path pre_shared_key "/etc/racoon/psk.txt";

remote anonymous
{
    exchange_mode aggressive,main;
    doi ipsec_doi;
    situation identity_only;

    my_identifier address;

    lifetime time 2 min;      # sec,min,hour
    initial_contact on;
    proposal_check obey;     # obey, strict or claim

    proposal {
        encryption_algorithm 3des;
        hash_algorithm sha1;
        authentication_method pre_shared_key;
        dh_group 2 ;
    }
}

sainfo anonymous
{
    pfs_group 1;
    lifetime time 2 min;
    encryption_algorithm 3des, blowfish, des, cast128, rijndael ;
    authentication_algorithm hmac_sha1, hmac_md5;
    compression_algorithm deflate ;
}
```

Esta configuración define fragmentos para conexiones predeterminadas (anónimas). El fragmento **remote anonymous** define parámetros para conectarse a sistemas remotos, y la sección **sainfo anonymous** proporciona información para instrucciones de asociación de seguridad, como los métodos de autenticación y cifrado que se utilizarán.

Certificados

Para utilizar certificados en lugar de claves compartidas, primero tiene que crearlos con OpenSSL. Después instruya a **racoon** para que los utilice. Debe especificar la ruta a los certificados.

```
path certificate "/usr/local/etc/racoon/certs";
```

Ahora, puede configurar **racoon** para que utilice las claves públicas y privadas generadas por el certificado. En el fragmento apropiado del archivo **/etc/racoon/racoon.conf**, la instrucción **certificate_type** especifica las claves públicas y privadas para este sistema. La instrucción **peers_certfile** especifica la ubicación de la clave pública del sistema. La instrucción **authentication_method** tiene ahora el valor **rsasig**, como las claves pública y privada para RSA. Asegúrese de que cada sistema tiene las claves privada y pública correspondientes.

```
certificate_type x509 "192.168.0.2.public" "192.168.0.2.private";
peers-certfile "192.168.0.5.public";
authentication_method rsasig;
```

Configuración de conexión con racoon

Con **racoon**, sólo necesitará especificar la directiva de seguridad para la configuración de conexión, como se muestra aquí para el emisor. El receptor tendrá las directivas correspondientes:

```
spdadd 192.168.0.5 192.168.0.2 any -P out ipsec
    esp/transport//require
    ah/transport//require;
spdadd 192.168.0.2 192.168.0.5 any -P in ipsec
    esp/transport//require
    ah/transport//require;
```

Tablas de Ipsec e IP: Net Traversal

El filtro de red IPtables detendrá muchos paquetes IPsec. Para permitir que IPtables pase paquetes IPsec, utilice los siguientes comandos de IPtables. El número para un protocolo AH es 51, y para el protocolo ESP, es 50. Para permitir paquetes IPsec, debe establecer reglas de directivas como las siguientes:

```
iptables -A INPUT -p 50 -j ACCEPT
iptables -A OUTPUT -p 51 -j ACCEPT
```

En el caso de filtros de red que implementan enmascaramiento IP, necesitará agregar la opción **net_traversal** a su configuración IPsec de racoon. Con Net Traversal, la conexión IPsec omite la sustitución de dirección IP realizada por IPtables cuando se enmascaren las direcciones IP. Además, la opción **nat_keepalive** mantendrá la conexión y con la opción **iskamp_natt**, especificará una dirección IP y el puerto al que se conecta.

Modo de túnel de IPsec: redes privadas virtuales

En lugar de cifrar dos host directamente, utilice IPsec para codificar sólo puertas de enlace entre las redes a las que pertenecen los host, suponiendo que la comunicación dentro de esas redes es confiable. Esto reduce significativamente la configuración de cifrado, permitiendo que los host de toda una red alcancen a las de otras redes, al utilizar una conexión IPsec intermediaria segura entre las puertas de enlace. En el caso de conexiones entre puertas de enlace, las transmisiones se envían a través de rutas de intervención que pueden entunelarse. A esto se le conoce como el modo de túnel para IPsec, que se utiliza para implementar redes privadas virtuales (VPN). Las transmisiones cifradas entre puertas de enlace implementan de manera efectiva las VPN, asegurando las transmisiones a través de una red más grande, de una red local a otra.

Para entunelar transmisiones de un host a través de una puerta de enlace con una red, utilice la opción **-m tunnel**. La conexión IPsec es entre dos puertas de enlace. En el siguiente ejemplo se muestra la asociación de seguridad en la puerta 10.0.0.1 que codifica las transmisiones de la puerta de enlace 10.0.0.1 a la 10.0.23.5. Los ejemplos utilizados aquí son para conexiones de puerta a puerta, configurada con una conexión directa entre dos hosts al utilizar claves manuales.

```
add 10.0.0.1 10.0.23.5 esp 34501 -m tunnel -E 3des-cbc "clavesecreta";
```

La directiva de seguridad en 10.0.0.1 implementa entonces el cifrado para la comunicación de una red a otra, al utilizar las puertas de enlace respectivas. Las dos redes son 192.168.0.0 y 192.162.1.0. Su puerta de enlace 10.0.0.1 cifra las transmisiones del host en la red 192.168.0.0, y las envía a la puerta de enlace de la red 192.168.1.0, que es 10.0.23.5, en que se descifran.

```
spdadd 192.168.0.0/24 192.168.1.0/24 any -P out ipsec esp/tunnel/10.0.0.1-10.0.23.5/require;
```

Observe que la dirección IP de la puerta de enlace es especificada en la instrucción **spdadd** de la directiva **ipsec**. El modo de túnel es especificado en vez del modo de transporte.

```
ipsec esp/tunnel/10.0.0.1-10.0.23.5/require
```

La puerta de enlace que recibe, 10.0.23.5, tendrá una asociación y directiva de seguridad correspondiente, como se muestra aquí. La directiva se establece para transmisiones entrantes. En ambas configuraciones de puerta de enlace, aparte de especificar la opción *tunnel* y utilizar las direcciones de red en la directiva de seguridad, las asociaciones de seguridad y directivas son las mismas que las utilizadas para conexiones de host a host.

```
add 10.0.0.1 10.0.23.5 esp 34501 -m tunnel -E 3des-cbc "clavesecreta";
```

```
spdadd 192.168.0.0/24 192.168.1.0/16 any -P in ipsec esp/tunnel/10.0.0.1-10.0.23.5/require;
```

Para configurar una comunicación completa de dos vías, las dos puertas de enlace tienen asociaciones y directivas de seguridad correspondientes para administrar el tráfico en ambas direcciones. En el siguiente ejemplo se muestra la configuración de la puerta de enlace 10.0.0.1 y se administra el tráfico de dos vías con la puerta 10.0.23.5, que tiene una configuración similar:

```
add 10.0.0.1 10.0.23.5 esp 34501 -m tunnel -E 3des-cbc "clavesecreta";
add 10.0.23.5 10.0.0.1 esp 34501 -m tunnel -E 3des-cbc "clavesecreta";
```

```
spdadd 192.168.0.0/24 192.168.1.0/24 any -P out ipsec esp/tunnel/10.0.0.1-  
10.0.23.5/require;
```

```
spdadd 192.168.1.0/16 192.168.0.0/24 any -P in ipsec esp/tunnel/10.0.0.1-  
10.0.23.5/require;
```

Si utiliza **racoon** para configurar las conexiones de puerta de enlace, tiene que establecer sólo las directivas de seguridad para cada puerta, permitiendo que el servidor **racoon** genere las asociaciones de seguridad necesarias.

Secure Shell y Kerberos

Para proteger las conexiones remotas de host fuera de su red, es posible cifrar las transmisiones (véase la Tabla 19-1). En el caso de sistemas Linux, se utiliza el conjunto de programas de Secure Shell (SSH, shell segura) para codificar y autenticar transmisiones, previniendo que alguien más las lea o modifique, además de confirmar la identidad del emisor. Los programas SSH están hechos para remplazar herramientas remotas como rsh y rcp, que no realizaban cifrado e incluían riesgos de seguridad como transmitir contraseñas en texto simple. Los servidores Kerberos controlan la autenticación de usuarios para ciertos servicios. También ofrece otro nivel de seguridad, con el que pueden protegerse servicios individuales, permitiendo el uso de un servicio, sólo para usuarios con acceso autorizado.

Secure Shell: OpenSSH

Aunque un firewall protege una red contra intentos de irrumpir en ella desde el exterior, todavía existe el problema de garantizar comunicaciones legítimas de fuentes externas a la red. Un problema particular es el de usuarios intentando conectarse a su red de manera remota. Tales conexiones pueden monitorearse y la información como contraseñas e ID de usuario utilizadas, cuando el usuario inicia sesión en su red puede copiarse y emplearse después para entrar en su sistema. Una solución consiste en utilizar SSH para inicios de sesión remotos y otros tipos de conexiones remotas, como transferencias FTP. SSH cifra cualquier comunicación entre el usuario remoto y un sistema en su red.

Dos diferentes implementaciones de SSH usan lo que son, en efecto, dos protocolos diferentes e incompatibles. La primera versión de SSH, conocida como SSH1, usa el protocolo SSH original. La versión 2.0, conocida como SSH2, utiliza una versión totalmente reescrita del protocolo SSH. El cifrado se realiza en formas diferentes, cifrando diferentes partes de un paquete. SSH1 utiliza claves de servidor y host para autenticar sistemas, mientras SSH2 sólo recurre a claves de host. Además, ciertas funciones, como sftp, sólo tienen soporte en SSH2.

NOTA Una versión comercial de SSH está disponible en SSH Communications Security, cuyo sitio Web es ssh.com. SSH Communications Security brinda una versión totalmente comercial llamada SSH Tectia, diseñada para uso en empresas o gobierno. El paquete más antiguo y no comercial todavía está disponible de manera gratuita, puede descargarlo y usarlo.

Sitio Web	Descripción
openssh.org	Versión de código fuente de OpenSSH de SSH
ssh.com	SSH Communications Security, versión comercial de SSH
web.mit.edu/kerberos	Autenticación de Kerberos

TABLA 19-1 Recursos de SSH y Kerberos

El protocolo SSH se ha vuelto estándar de Internet Engineering Task Force (IETF) oficial. El proyecto OpenSSH, con soporte técnico del proyecto OpenBSD desarrolla y mantiene una versión gratuita y de fuente abierta. OpenSSH es la versión ofrecida con la mayor parte de distribuciones de Linux, incluidos Fedora, Red Hat, Novell y Debian. Conozca más acerca de OpenSSH en **openssh.org**, donde se descargan casi todas las versiones recientes, aunque su distribución proporcionará versiones RPM actuales.

Cifrado y autenticación de SSH

SSH asegura conexiones autenticando usuarios y cifrando sus transmisiones. El proceso de autenticación se maneja con el cifrado de clave pública. Una vez autenticado, las transmisiones se cifran con un codificador acordado entre servidor y cliente SSH, para utilizarse en una sesión particular. SSH soporta varios codificadores. La autenticación aplica para hosts y usuarios. SSH primero autentifica un host en particular, verificando se trate de un host válido para SSH, con el que puede comunicarse de manera segura. Después el usuario se autentifica, verificando que el usuario es quien dice ser.

SSH utiliza métodos de cifrado poderosos y su exportación desde Estados Unidos puede estar restringida. Actualmente, SSH se ocupa de los siguientes tipos de ataques:

- Engaño de IP, donde un host remoto envía paquetes pretendiendo venir de otro host confiable.
- Enrutamiento del origen IP, donde un host pretende que un paquete IP viene de otro host confiable.
- Engaño DNS, donde el atacante falsifica los registros del servidor de nombre
- Intercepción de contraseñas de texto simple y otros datos mediante host intermediarios
- Manipulación de datos por parte de gente en control de host intermediarios
- Ataques basados en escucha de datos de autenticación X y conexiones engañando al servidor X11

Cifrado

El cifrado de claves públicas usado en la autenticación SSH, hace uso de dos claves: las claves pública y privada. La *clave pública* se utiliza para cifrar datos, mientras la *clave privada* los descifra. Cada host o usuario tiene sus propias claves pública y privada. La clave pública se distribuye a otros hosts, usándola para cifrar datos de autenticación que sólo la clave privada del host puede descifrar. Por ejemplo, cuando un host envía datos a un usuario en otro sistema, cifra datos de

autenticación con una clave pública, recibida previamente de ese usuario. Los datos se descifran sólo con la clave privada correspondiente del usuario. La clave pública se envía abiertamente de manera segura de un host a otro, permitiéndole que se instale de manera segura en diferentes hosts. Considere como si el proceso tuviera lugar entre cliente y servidor. Cuando el cliente envía datos al servidor, primero los cifra con la clave pública del servidor. El servidor decodifica entonces datos al utilizar su propia clave privada.

Se recomienda las transmisiones SSH se autentifiquen con claves públicas y privadas, controladas por frases de contraseña. Una vez autenticados, los participantes aceptan un codificador común para cifrar transmisiones. La autenticación verificará la identidad de los participantes. Cada usuario intentando usar SSH para acceder una cuenta remota primero necesita crear las claves públicas y privadas junto con una frase de contraseña para utilizarlas en el proceso de autenticación. Luego, un usuario envía la clave pública a la cuenta remota que quiere usar para acceder e instalar la clave pública en esa cuenta. Cuando el usuario intenta acceder la cuenta remota, esa cuenta utiliza la clave pública del usuario para autenticar que el usuario es quien dice ser. En el proceso se supone que la cuenta remota ha configurado sus propias claves públicas y privadas de SSH. Para que el usuario acceda a una cuenta remota, necesita conocer la frase de contraseña SSH de la cuenta remota. SSH a menudo se usa en situaciones donde un usuario tiene dos o más cuentas localizadas en diferentes sistemas y quiere acceder de manera segura a éstas desde la otra. En este caso, el usuario ya tiene acceso a cada cuenta e instala SSH en cada una, al dar a cada una sus propias claves privada y pública, junto con frases de contraseña.

Autenticación

El mecanismo de autenticación en las versiones 1 y 2 de SSH difiere un poco. Sin embargo, el procedimiento es el mismo para usuarios. En esencia, un usuario crea las claves privada y pública. Para ello, utiliza el comando **ssh-keygen**. Luego la clave pública del usuario debe distribuirse a los usuarios que el usuario original quiere acceder. El usuario original necesitará conocer la frase de contraseña del otro usuario para acceder a él.

SSH versión 1 usa autenticación RSA. Cuando un usuario remoto intenta iniciar sesión en una cuenta, se revisa ésta para ver si tiene la clave pública del usuario remoto. Entonces se utiliza esa clave para cifrar un desafío (por usualmente, un número aleatorio) cifrado sólo con la clave privada del usuario remoto. Cuando el usuario remoto recibe el desafío cifrado, ese usuario descifra el desafío con su clave privada. La versión 2 de SSH utiliza autenticación de RSA o DSA. El usuario remoto primero cifra un identificador de sesión con su clave privada, firmándola. Luego, la cuenta descifra el identificador de sesión con la clave pública del usuario. SSH ha configurado previamente el identificador de sesión para esa sesión.

La autenticación SSH se realiza primero con el host y después los usuarios. Cada host tiene sus propias claves públicas y privadas, utilizadas para autenticación. Una vez el host se autentifica, consulta al usuario. Cada usuario tiene claves pública y privada propias. Los usuarios de un servidor SSH que quieren recibir conexiones de usuarios remotos deberán mantener una lista de claves públicas del usuario remoto. De manera similar, un host SSH mantendrá una lista de claves públicas de otros host SSH.

Herramientas de SSH

SSH se implementa en sistemas Linux con OpenSSH. El conjunto completo de paquetes OpenSSH incluye el paquete general OpenSSH (openssh), servidor OpenSSH (openssh-server) y clientes OpenSSH (openssh-clients). Estos paquetes también requieren OpenSSL (openssl), que instala las bibliotecas criptográficas usadas por SSH.

Aplicación	Descripción
Ssh	Cliente SSH
Sshd	Servidor SSH (daemon)
Sftp	Cliente FTP de SSH, Secure File Transfer Program. Sólo la versión 2. Utilice ? para mostrar una lista de comandos de sftp (protocolo SFTP)
sftp-server	Servidor FTP de SSH. Sólo la versión 2 (protocolo SFTP))
Scp	Cliente de copia de comandos de SSH
ssh-keygen	Utilería para generar claves. -h para ayuda
ssh-keyscan	Herramienta para recopilar claves de host públicas para generar archivos de ssh_known_hosts
ssh-add	Agrega identidades RSD y DSA al agente de autenticación
ssh-agent	Agente de autenticación de SSH que almacena claves privadas para autenticación de clave pública (RSA, DSA)
ssh-askpass	Utilería de X Window System para consultar contraseñas, se invoca con ssh-add (openssh-askpass)
ssh-askpass-gnome	Utilería de GNOME para consultar contraseñas, se invoca con ssh-add
ssh-signer	Firma paquetes de autenticación basados en host. Sólo la versión 2. Debe de ser suid root (realizado por la instalación)
Slogin	Inicio de sesión remoto (versión 1)

TABLA 19-2 Herramientas de SSH

Las herramientas SSH se muestran en la lista de la tabla 19-2. Incluyen varios programas de cliente como scp y ssh, además del servidor ssh. Éste (sshd) ofrece conexiones seguras a cualquiera del exterior usando el cliente ssh para conectarse. También incluye utilerías de configuración, como ssh-add, que agrega host válidos al agente de autenticación y ssh-keygen, para generar la clave utilizada para cifrado.

Para la versión 2, los nombres de herramientas actuales tienen un sufijo 2. Las herramientas de la versión 1 tienen 1 como sufijo. Sin embargo, durante la instalación se establecen vínculos con cada herramienta para usar sólo el nombre con el sufijo. Por ejemplo, si tiene instalada la versión 2, existe un vínculo llamado scp a la aplicación scp2. Entonces puede usar el vínculo para invocar la herramienta. Al utilizar scp se inicia scp2. En la tabla 19-2 se especifican sólo nombres de vínculo, porque son los mismos para cada versión. Aunque recuerde, algunas aplicaciones, como sftp, están disponibles sólo con la versión 2.

Configuración de SSH

El uso de SSH requiere la creación de claves públicas y privadas propias, después la distribución de su clave pública a otros usuarios a los que quiere acceder. Puede tratarse de usuarios diferentes o simplemente de cuentas de usuario en sistemas remotos. A menudo la gente inicia sesión de manera remota desde un cliente local en una cuenta de un servidor remoto, tal vez desde una computadora en casa a una empresarial. Su equipo casero sería su cuenta de cliente y la cuenta en el equipo de la empresa sería su cuenta de servidor. En su cuenta de cliente, necesita generar sus claves pública y privada, después colocar una copia de su clave pública en la cuenta del servidor. Para esto, simplemente debe enviar por correo electrónico el archivo de la clave o copiar el archivo desde un

disco flexible. Una vez la cuenta en su servidor tenga copia de la clave pública del usuario cliente, puede acceder la cuenta del servidor desde la cuenta cliente. También se le pedirá la frase de contraseña de la cuenta del servidor. Deberá saberla para acceder la cuenta. En la figura 19-1 se ilustra la configuración de SSH que permite al usuario **jorge** acceder la cuenta **cecilia**.

Para que se le permita el uso de SSH para acceder otras cuentas:

- Debe crear las claves pública y privada en su cuenta, junto con la frase de contraseña. Necesitará usar la frase de contraseña para acceder su cuenta desde otra cuenta.
- Debe distribuir su clave pública a otras cuentas que quiera acceder, colocándolas en el archivo `.ssh/authorized_keys`.
- Otras cuentas también deben configurar las claves pública y privada, junto con la frase de contraseña.
- Debe conocer las frases de contraseña de otras cuentas para acceder a éstas.

Creación de claves de SSH con ssh-keygen

Sus claves pública y privada se crean al usar el comando `ssh-keygen`. Necesita especificar el tipo de cifrado que quiere. Utilice cifrado DSA o RSA. Indique el tipo mediante la opción `-t` y el nombre del cifrado en minúsculas (`dsa` o `rsa`). En el siguiente ejemplo, el usuario crea una clave con cifrado RSA:

```
ssh-keygen -t rsa
```

El comando `ssh-keygen` pide su frase de contraseña, que utilizará como tipo de contraseña para proteger su clave privada. La frase de contraseña debe estar formada por varias palabras. También se le pide escriba un nombre de archivo para las claves. Si no ingresa uno, SSH utilizará sus opciones predeterminadas. A la clave pública se le dará la extensión `.pub`. El comando `ssh-keygen` genera la clave pública y la coloca en su propio archivo `.ssh/id_dsa.pub` o `.ssh/id_rsa.pub`,

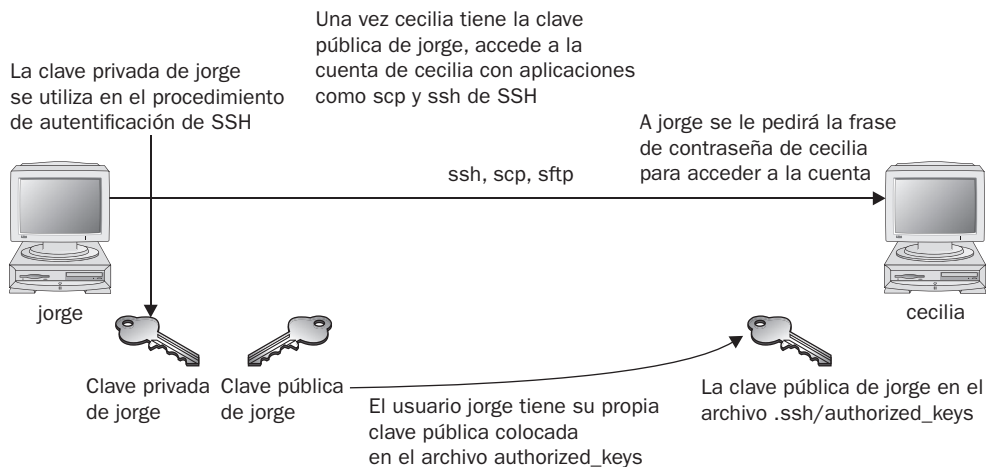


FIGURA 19-1 Configuración y acceso a SSH

dependiendo del tipo de clave especificado; coloca la clave privada en el archivo This is de `.ssh/id_dsa.pub` o `.ssh/id_rsa.pub` correspondiente.

NOTA El nombre de archivo `.ssh/identity` se usa en la versión 1 de SSH; puede estar instalado de forma predeterminada en versiones de distribuciones más antiguas. La versión 2 de SSH usa un nombre de archivo diferente, `.ssh/id_dsa` o `.ssh/id_rsa`, si se utiliza autenticación DSA o RSA, respectivamente.

Si necesita cambiar su frase de contraseña, hágalo con el comando `ssh-keygen` y la opción `-p`. Cada usuario tendrá su propio directorio de configuración de SSH, llamado `.ssh` y ubicado en el directorio `home`. Las claves pública y privada, además de archivos de configuración SSH, se colocan aquí. Si construye desde el código fuente, la operación `make install` ejecutará automáticamente `ssh-keygen`. En la tabla 19-3 se muestra una lista de archivos de configuración de SSH.

Archivo	Descripción
<code>\$HOME/.ssh/known_hosts</code>	Registra claves de host para todos los hosts en que haya iniciado sesión el usuario (que no están en <code>/etc/ssh/ssh_known_hosts</code>).
<code>\$HOME/.ssh/random_seed</code>	Siembra el generador de número aleatorio.
<code>\$HOME/.ssh/id_rsa</code>	Contiene la identidad de autenticación RSA del usuario.
<code>\$HOME/.ssh/id_dsa</code>	Contiene la identidad de autenticación DSA del usuario.
<code>\$HOME/.ssh/id_rsa.pub</code>	Contiene la clave pública RSA para autenticación. El contenido de este archivo se debe agregar a <code>\$HOME/.ssh/authorized_keys</code> en todas las máquinas donde quiera iniciar sesión al utilizar la autenticación RSA.
<code>\$HOME/.ssh/id_dsa.pub</code>	Contiene la clave pública DSA para autenticación. El contenido de este archivo se debe agregar a <code>\$HOME/.ssh/authorized_keys</code> en todas las máquinas donde quiera iniciar sesión al utilizar la autenticación DSA.
<code>\$HOME/.ssh/config</code>	El archivo de configuración por usuario.
<code>\$HOME/.ssh/authorized_keys</code>	Muestra una lista de claves RSA o DSA usadas para inicio de sesión como ese usuario.
<code>/etc/ssh/ssh_known_hosts</code>	Contiene una lista de todas las claves host conocidas en todo el sistema.
<code>/etc/ssh/ssh_config</code>	Contiene el archivo de configuración de todo el sistema. Este archivo ofrece opciones predeterminadas para los valores sin especificación en el archivo de configuración del usuario.
<code>/etc/ssh/sshd_config</code>	Contiene el archivo de configuración del servidor.
<code>/etc/ssh/sshrdrc</code>	Contiene las opciones predeterminadas del sistema. <code>ssh</code> ejecuta los comandos en este archivo cuando el usuario inicia sesión, justo antes de que la shell del usuario (o indicador de comandos) inicie.
<code>\$HOME/.ssh/rc</code>	Contiene comandos ejecutados por <code>ssh</code> cuando el usuario inicia sesión, justo antes de que la shell del usuario (o indicador de comandos) inicie.

TABLA 19-3 Archivos de configuración de SSH

Claves autorizadas

Una clave pública se usa para validar un usuario y su host. Puede utilizar la clave privada en un sistema remoto para permitir el acceso a usuarios. La clave pública se coloca en un archivo **.ssh/authorized_keys** de la cuenta del usuario. Recuerde que la clave pública para DSA se almacena en el archivo **.ssh/id_dsa.pub**. Si un usuario quiere iniciar sesión en forma remota desde una cuenta local en una cuenta de un sistema remoto, primero debería colocar la clave pública para DSA en el archivo **.ssh/authorized_keys** en la cuenta del sistema remoto al que quieren acceder. Si el usuario **larisa** en **tortuga.mipista.com** quiere acceder la cuenta **alicia** en **conejo.mytrek.com**, primero debe colocar la clave pública de **larisa** en **/home/larisa/.ssh/id_dsa.pub** en el archivo **authorized_keys** de **alicia**, **/home/alicia/.ssh/authorized_keys**. El usuario **larisa** envía la clave o hace que se copie. Una simple operación **cat** adjunta una clave al archivo de clave autorizada. En el siguiente ejemplo, el usuario agrega la clave pública para **alicia** en el archivo **larisa.pub** al archivo de clave autorizada. El archivo **larisa.pub** es una copia del archivo **/home/larisa/.ssh/id_dsa.pub** que el usuario recibió antes.

```
cat larisa.pub >> .ssh/authorized_keys
```

Carga de claves

Si hace conexiones frecuentes a diversos hosts remotos, use el comando **ssh-agent** para colocar claves privadas en la memoria, adonde se tiene acceso rápido para decodificar transmisiones recibidas. El comando **ssh-agent** está hecho para usarse al principio de un inicio de sesión. En el caso de GNOME, maneje la utilidad **openssh-askpass-gnome**, invocada con **ssh-add**, que le permite escribir una contraseña cuando inicia sesión en GNOME. GNOME proporcionará de manera automática esa contraseña siempre que use el cliente SSH.

Aunque el comando **ssh-agent** permite el uso de claves privadas en la memoria, también debe cargar sus claves privadas específicamente en la memoria con el comando **ssh-add**. **ssh-add** sin argumentos carga su clave privada desde el archivo **.ssh/id_dsa** o **.ssh/id_rsa.pub**. Se le pide su frase de contraseña para esta clave privada. Para eliminar la clave de la memoria, utilice **ssh-add** con la opción **-d**. Si tiene varias claves privadas, puede cargar todas en la memoria; **ssh-add** con la opción **-l** muestra una lista de todas las claves cargadas en ese momento.

Clientes SSH

SSH se diseñó originalmente para remplazar operaciones de acceso remoto, como **rlogin**, **rcp** y **Telnet**, que no realizaban cifrado e introducían riesgos de seguridad, como transmitir contraseñas en texto simple. También puede emplear SSH para cifrar sesiones de servidor X, además de transmisiones FTP (**sftp**). El paquete de clientes **ssh** contiene clientes SSH correspondientes para remplazar estas aplicaciones. Con **slogin** o **ssh**, inicie sesión desde un host remoto para ejecutar comandos y aplicaciones, como haría con **rlogin** y **rsh**. Con **scp**, copie archivos entre el host remoto y un host de red, como con **rcp**. Con **sftp**, transfiera archivos RTP asegurados por cifrado.

ssh

Con **ssh**, inicie sesión de manera remota desde un cliente local hasta un sistema remoto en su red, al operar como servidor SSH. El término *cliente local* alude aquí alguien fuera de la red, como su computadora en casa y el término *remoto* se refiere a un sistema host, en la red a la que está conectado. En efecto, usted se conecta desde su sistema local al host de red remoto. Está diseñado para remplazar **rlogin**, realizando inicios de sesión remotos y **rsh**, que ejecuta comandos remotos. Con **ssh**, puede iniciar sesión desde un sitio local hasta un host remoto de su red y después enviar

comandos, que habrán de ejecutarse en ese host. El comando `ssh` también soporta conexiones con X Window System. Esta característica se habilita automáticamente si hace una conexión `ssh` desde un entorno de X Window System, como GNOME o KDE. Se configura una conexión entre su servidor X y el servidor X remoto. El host remoto configura un servidor X modelo y envía cualquier dato de X Window System, a través de éste, a su sistema local, para ser procesado por su servidor X local.

La operación de inicio de sesión `ssh` funciona casi igual al comando `rlogin`. Inserte el comando `ssh` con la dirección del host remoto, seguida por la opción `-l` y el nombre de inicio de sesión (nombre de usuario) de la cuenta remota que inicia. En el siguiente ejemplo inicia sesión en la cuenta de usuario **alicia** en el host **conejo.mipista.com**:

```
ssh conejo.mipista.com -l alicia
```

También puede usar el nombre de usuario con formato de dirección con `ssh`, como

```
ssh alicia@conejo.mipista.com
```

En la siguiente lista se muestra cómo accede el usuario **jorge** a la cuenta **cecilia** en **tortuga.mipista.com**:

```
[jorge@tortuga jorge]$ ssh tortuga.mipista.com -l cecilia
cecilia@tortuga.mipista.com's password:
[cecilia@tortuga cecilia]$
```

Hay varias opciones permitiéndole configurar su conexión. Casi todas tienen opciones de configuración que se pueden modificar en el archivo de configuración. Por ejemplo, con la opción `-c`, indica el método de cifrado que quiere utilizar, como **idea**, **des**, **blowfish** o **arcfour**. Con la opción `-i`, elige una clave privada particular. La opción `-c` permite transmisiones comprimidas en niveles específicos (consulte la página Man de `ssh`, para conocer una lista completa de opciones).

scp

`scp` se usa para copiar archivos de un host a otro en una red. Diseñado para remplazar a `rcp`, `scp` utiliza `ssh` para transferir datos, empleando los mismos métodos de autenticación y cifrado. Si la autenticación lo requiere, `scp` pide una contraseña o frase de contraseña. El programa `scp` opera casi igual que `rcp`. Los directorios y archivos de hosts remotos se especifican mediante el nombre de usuario y dirección de host antes del nombre de archivo o directorio. El nombre de usuario especifica la cuenta de usuario remoto a la que está accediendo `scp`, mientras el host es el sistema remoto donde esa cuenta se localiza. Separe al usuario de la dirección host con una `@`, y la dirección del host del nombre de archivo o directorio con dos puntos. En el siguiente ejemplo se copia el archivo **fiesta** de un directorio actual de usuario, al directorio **aniversario** del usuario **alicia**, ubicado en el host **conejo.mipista.com**:

```
scp party alicia@conejo.mipista.com:/aniversario/fiesta/
```

La opción `-r` (repetir) reviste un interés particular, porque permite copiar todos los directorios. Consulte la página Man de `scp` para conocer una lista completa de opciones. En el siguiente ejemplo, el usuario copia todo el directorio **informes** al directorio **proyectos**, del usuario **julio**:

```
scp -r informes julio@conejo.mipista.com:/proyectos
```

En el siguiente ejemplo, el usuario **jorge** copia el archivo **midoc1** desde el directorio **home** de **cecilia**:

```
[jorge@tortuga jorge]$ scp cecilia@tortuga.mipista.com:midoc1
cecilia@tortuga.mipista.com's password:
midoc1      0% |                               | 0 ---
ETA
midoc1    100% |*****| 17 00:00
[jorge@tortuga jorge]$
```

Desde un sistema Windows, puede usar clientes **scp** como **winscp**, que interactuará con sistemas scp habilitados de Linux.

sftp y sftp-server

Con **sftp**, puede transferir archivos FTP asegurados por cifrado. El programa **sftp** usa los mismos comandos que **ftp**. Este cliente, trabaja sólo con la versión 2 de ssh, opera de manera muy similar a **ftp**, con muchos de los mismos comandos. Utilice **sftp** en vez de **ftp** para invocar el cliente sftp.

```
sftp ftp.redhat.com
```

Con el fin de emplear el cliente sftp para conectarse al servidor FTP, ese servidor necesita usar la aplicación sftp-server. El servidor ssh invoca a sftp-server para proporcionar transmisiones FTP cifradas para quienes usan el cliente sftp. Servidor y cliente sftp emplean el protocolo de transferencia de archivos de SSH (SFTP, SSH File Transfer Protocol) para realizar operaciones FTP de manera segura.

Reenvío de puerto (entunelamiento)

Si, por alguna razón, sólo se conecta a un host seguro pasando por un host inseguro, ssh proporciona una característica llamada reenvío de puerto. Con *reenvío de puerto*, se asegura el segmento inseguro de su conexión. Esto sólo requiere la especificación del puerto en el que el host inseguro está conectado al seguro. Esto configura una conexión directa entre el host local y el remoto, a través del host inseguro intermediario. Los datos cifrados pasan directamente. A este proceso se le conoce como entunelamiento, porque crea un túnel de datos cifrados a través de servidores conectados.

Tiene la opción de configurar el reenvío de puerto a uno en el sistema remoto u otro de su sistema local. Para reenviar un puerto del sistema remoto a un puerto en su sistema local, use **ssh** con la opción **-R**, seguida por un argumento con el puerto local, dirección de host remoto y puerto remoto al que se reenviará, cada uno separado por dos puntos. Esto funciona al asignar un conector para escuchar el puerto en el lado remoto. Cada vez que hace una conexión a este puerto, se reenvía a través del canal seguro, y hace una conexión a un puerto remoto desde la máquina local. En el siguiente ejemplo, el puerto 22 del sistema local se conecta al puerto 23 del sistema remoto **conejo.mipista.com**:

```
ssh -R 22:conejo.mipista.com:23
```

Para reenviar un puerto de su sistema local al puerto de uno remoto, use la opción **ssh -L**, seguida por un argumento con el puerto local, dirección de host remoto y puerto remoto al que se reenviará, separando con dos puntos cada uno de los argumentos. Se asigna un conector para escuchar el puerto en el lado local. Siempre que haga una conexión en este puerto, ésta se reenvía a través del canal seguro y hace una conexión a un puerto remoto desde la máquina remota. En el

siguiente ejemplo, el puerto 22 del sistema local se conecta al puerto 23 del sistema remoto **conejo.mipista.com**:

```
ssh -L 22: conejo.mipista.com:23
```

Use las opciones `LocalForward` y `RemoteForward` en su archivo `.ssh/config`, para configurar el reenvío de puerto a un host particular o especificar uno predeterminado para todos los hosts a que se conecta.

Configuración de SSH

El archivo de configuración SSH para cada usuario está en el archivo `.ssh/config`. El archivo `/etc/ssh/ssh_config` se usa para configurar opciones predeterminadas para todo el sistema. En el archivo de configuración, puede establecer varias opciones, como las mostradas en la lista del documento `Man ssh_config`. El archivo de configuración está diseñado para especificar opciones para hosts remotos, diferentes a los que quizás quiera conectarse. Está organizado en segmentos, donde cada segmento comienza con la palabra clave **HOST**, seguida por una dirección IP para el host. Las siguientes líneas almacenan opciones que puede haber configurado para ese host. Un segmento termina en la siguiente entrada **HOST**. Son de particular interés las opciones **User** y **Cipher**. La opción **User** se utiliza para especificar nombres de usuarios en el sistema remoto a los que se permitió el acceso. Con la opción **Cipher**, puede elegir el método de cifrado que usará para un host particular. Los métodos de cifrado incluyen IDEA, DES (estándar), triple-DES (3DES), Blowfish (128 bits), Arfour (RC4 de RSA) y Twofish. En el siguiente ejemplo se permite el acceso desde **larisa** en **tortuga.mipista.com** y usa el cifrado Blowfish para transmisión:

```
Host tortuga.mipista.com
    User larisa
    Compression no
    Cipher blowfish
```

Para especificar opciones globales aplicables a cualquier host que se conecte, cree la entrada **HOST** con el asterisco como su host, **HOST ***. Esta entrada debe colocarse al final del archivo de configuración, porque una opción sólo cambia la primera vez que se configura. Cualquier entrada posterior para una opción se ignora. Debido a que los hosts coinciden en su propia entrada y la entrada global, su entrada específica debe estar antes de la entrada global. El asterisco (*) y signo de interrogación (?) son comodines de operadores de coincidencia permitiendo especificar un grupo de hosts con los mismos sufijo o prefijo.

```
Host *
    FallBackToRsh yes
    KeepAlive no
    Cipher idea
```

Kerberos

La autenticación de usuario puede controlarse aún más para ciertos servicios con servidores Kerberos, analizados en esta sección. La autenticación Kerberos proporciona otro nivel de seguridad, por medio del que se protegen servicios individuales, permitiendo el uso de un servicio sólo a usuarios con acceso autorizado. Los servidores Kerberos se habilitan y configuran con **authconfig-gtk** (Autenticación en el menú Herramientas del sistema).

Kerberos es un protocolo de autenticación de red que proporciona autenticación cifrada a conexiones entre un cliente y servidor. Como protocolo de autenticación, Kerberos requiere que un cliente pruebe su identidad usando métodos de cifrado antes de acceder al servidor. Una vez autenticado, el cliente y servidor conducen todas las comunicaciones mediante cifrado. Mientras las firewall sólo protegen de ataques del exterior, Kerberos está diseñado para proteger también de ataques desde la propia red. Es probable que los usuarios de la red intenten entrar a sus servidores locales. Para prevenir esto, Kerberos coloca protección alrededor de los propios servidores, en vez de toda una red o equipo. Hay una versión gratuita disponible en el Instituto Tecnológico de Massachussets, en web.mit.edu/kerberos, bajo la licencia pública MIT, similar a la licencia pública de GNU. El nombre *Kerberos* viene de la mitología griega y es el nombre del perro guardián de tres cabezas de Hades. Asegúrese de revisar el sitio web.mit.edu/kerberos para estar al tanto de las actualizaciones y conocer documentación detallada, incluyendo preguntas frecuentes, manuales y tutoriales.

SUGERENCIA El paquete Kerberos V5 incluye su propia versión de herramientas de red como Telnet, RCP, FTP y RSH. Éstas proporcionan acceso autenticado seguro a usuarios remotos. Las herramientas funcionan de la misma forma que sus contrapartes originales. El paquete también contiene una versión Kerberos de sus comandos de inicio de sesión administrativos, *ksu*.

Servidores Kerberos

La clave de Kerberos es un servidor Kerberos a través del que se canalizan todas las peticiones de cualquier servicio de servidor. Después, el servidor Kerberos autentifica a un cliente, identificando a éste y validando el derecho del cliente a usar un servidor particular. El servidor mantiene una base de datos de usuarios autorizados. Luego, Kerberos envía al cliente un boleto cifrado usado por este para obtener acceso al servidor. Por ejemplo, si un usuario necesita revisar su correo, una solicitud de uso del servidor de correo se envía al servidor Kerberos, autenticando al usuario y enviando un boleto usado para acceder al servidor de correo. Sin el boleto enviado por Kerberos, nadie puede acceder a los servidores. Originalmente, este proceso requiere que los usuarios se sometan a un proceso de autenticación separado, para cada servidor al que quieran acceder. Sin embargo, ahora los usuarios necesitan realizar una autenticación inicial, válida para todos los servidores.

Este proceso requiere el uso de dos servidores, un servidor de autenticación (Analysis Services, Authentication Server) y un servidor de otorgamiento de boletos (TGS, Ticket-Granting Server). Juntos crean lo que se conoce como centro de distribución de claves (KDC, Key Distribution Center). En efecto, distribuyen claves utilizadas para desbloquear el acceso a servicios. El servidor de autenticación primero valida la identidad del usuario. AS envía un boleto, llamado boleto que otorga un boleto (TGT, Ticket-Granting Ticket), permitiendo al usuario acceder al servidor TGS, que después envía al usuario otro boleto para acceder realmente al servicio. De esta forma, el usuario nunca tiene acceso directo a ningún tipo de servidor durante el proceso de autenticación. El proceso es, en cierta forma, más complejo que lo descrito. Junto con el boleto, se envía un autenticador usando información como hora actual, suma de verificación y una clave de cifrado opcional, cifrándose con la clave de la sesión. Un servicio utiliza esta autenticación para verificar su identidad.

NOTA Consulte su lista de boletos actuales con el comando **klist**.

Proceso de autenticación

El servidor de autenticación valida un usuario empleando información de su base de datos de usuarios. Cada usuario necesita estar registrado en la base de datos del servidor de autenticación.

La base de datos incluirá una contraseña de usuario y otra información de éste. Para acceder al servidor de autenticación, el usuario proporciona nombre de usuario y contraseña. Esta última se utiliza para generar una clave de usuario con la que se cifra la comunicación entre AS y usuario. Éste tendrá su propia copia de la clave del usuario, con la que descifrará la comunicación. El proceso de autenticación se ilustra en la figura 19-2.

Para acceder al servicio con Kerberos deben darse los siguientes pasos:

1. El servidor de autenticación debe validar al usuario y se tiene que dar acceso al TGS con un boleto de clave de acceso. Haga esto enviando el comando **kinit**, que le pedirá inserte su nombre de usuario de Kerberos y después lo enviará al servidor de autenticación (el nombre de usuario de Kerberos suele ser el mismo que su nombre de usuario).
- `kinit`
2. AS genera un boleto que otorga un boleto con el que accede al servidor de otorgamiento de boletos. Este boleto incluirá una clave de sesión que se usará para permitirle acceder a TGS. El TGT enviará de regreso el cifrado con su clave de usuario (contraseña).
 3. El programa **kinit** pide entonces que escriba su contraseña de Kerberos, que después podrá usar para decodificar el TGT. Administre su contraseña de Kerberos con el comando **kpasswd**.
 4. Ahora puede usar un programa cliente, como el cliente de correo, para acceder al servidor de correo, por ejemplo. Cuando hace esto, el TGT accede a TGS, que después genera un boleto para acceder al servidor de correo. TGS genera una nueva clave de sesión que sólo se usará con el servidor de correo. Esto se otorga en el boleto enviado para acceder a dicho servidor. En efecto, existe una clave de sesión de TGT usada para acceder TGS, y una clave de sesión de correo para utilizarse en el servidor de correo. El boleto para servidor de correo se envía cifrado con la clave de sesión de TGS.

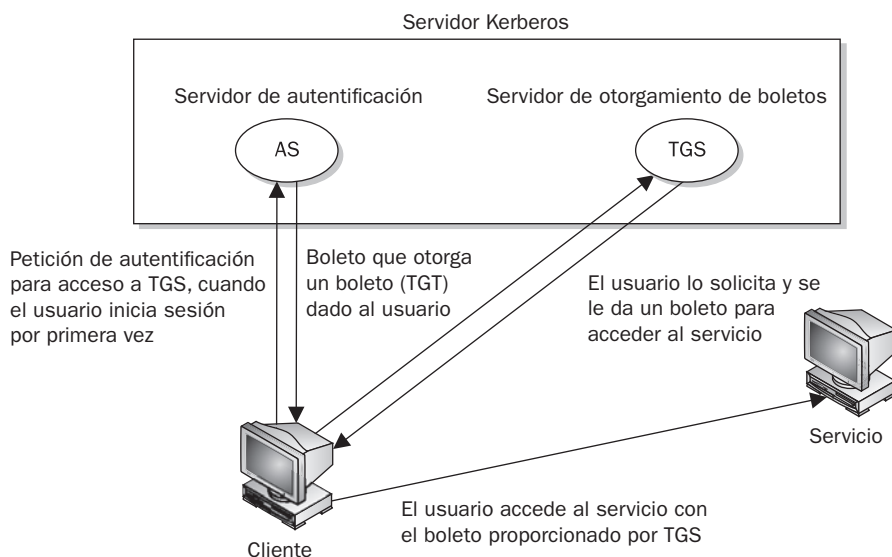


FIGURA 19-2 Autenticación de Kerberos

5. Entonces el cliente utiliza el boleto recibido del TGS para acceder al servidor de correo.
6. Si quiere utilizar otro servicio, como FTP, cuando su cliente FTP solicite un boleto a TGS, éste obtendrá automáticamente la autorización desde el servidor de autenticación y enviará un boleto FTP con la clave de sesión de FTP. Este tipo de soporte permanece activo por un periodo limitado, generalmente de algunas horas, tras lo que debe usar kinit nuevamente para someterse al proceso de autenticación y acceder a TGS. Destruya los boletos que tenga de manera manual con el comando **kdestroy**.

NOTA Con Kerberos v5 (versión 5), se proporciona una utilidad de inicio de sesión de Kerberos, por lo cual los usuarios obtienen un boleto que otorga un boleto cuando inician sesión normalmente. Esto evita el uso de kinit para obtener un TGT manualmente.

Servicios kerberizados

Configurar un servicio particular para utilizar Kerberos (conocido como kerberizar) llega a ser un proceso complicado. Para kerberizar un servicio se necesita revisar la identidad y las credenciales del usuario, buscar si hay un boleto para el servicio y, si no existe, obtener uno. Una vez que configurados, utilizar los servicios kerberizados es casi transparente para el usuario. Los boletos se envían automáticamente y la autenticación se lleva a cabo sin esfuerzo adicional para el usuario. El archivo **/etc/services** debe contener una lista de configuraciones de servicios kerberizados. Se trata de servicios como **kpasswd**, **ksu** y **klogin**, que proporcionan contraseña, acceso a superusuario y servicios de inicio de sesión de Kerberos.

Kerberos también proporciona sus propias herramientas de red kerberizadas para ftp, rsh, rcp y rlogin. Éstas se ubican en **/usr/kerberos/bin**, y casi todas tienen el mismo nombre que las herramientas de red originales. La secuencia de comandos **/etc/profile.d/krb5-workstation.sh** da prioridad a las herramientas de Kerberos, porque coloca el directorio de Kerberos antes que todos en la variable **PATH** del usuario.

Configuración de servidores Kerberos

La instalación y configuración de un servidor Kerberos también es un proceso complejo. Revise con cuidado la documentación para instalar las versiones actuales. Algunas de las áreas clave se muestran en una lista aquí. En el archivo de configuración de Kerberos, **krb5.conf**, puede configurar características como el método de cifrado utilizado y el nombre de la base de datos. Cuando instale Kerberos, asegúrese de seguir las instrucciones con cuidado para proporcionar acceso administrativo. Para ejecutar Kerberos, inicie el servidor Kerberos con el comando **service** y las secuencias de comandos **krb5kdc**, **kadmin** y **krb524**.

Necesitará configurar el servidor para su red, junto con clientes para cada host (el paquete **krb5-server** para servidores y **krb5-workstation** para clientes). Para configurar su servidor, primero especifique su reino y su dominio, al reemplazar manualmente la entradas **example.com** en minúsculas con **EXAMPLE.COM** en mayúsculas en los archivos **/etc/krb5.conf** y **/var/kerberos/krb5kdc/kdc.conf** con su propio nombre de dominio. Mantenga el mismo caso para cada entrada. Los reinos se especifican en mayúsculas y los nombres de host y dominio simples en minúsculas. Entonces cree una base de datos con el comando **kdb5_util** y la opción **create**. Se le pedirá que inserte una clave maestra.

```
kdb5_util create-s
```

El acceso administrativo completo al servidor se controla por las entradas de usuario en el archivo **var/kerberos/krb5kdc/kadm5.acl**. Remplace el texto **EXAMPLE.COM** con su reino de

Kerberos (por lo general, su nombre de dominio en mayúsculas). Después necesita agregar un principal local, un usuario local con acceso administrativo total desde el host en que se ejecuta el servidor. Inicie la herramienta `kadmin.local` y utilice el comando `addprinc` para agregar el principal local. Entonces puede iniciar sus servidores `krb5kdc5`, `kadmin` y `krb524`.

En cada host de cliente, utilice la herramienta `kadmin` con el comando `addprincipal` para agregar un principal al host. También agregue un host principal para cada host en su red con el calificador `host/`, como en `host/conejo.mipista.com`. Utilice la opción `-randkey` para especificar una clave aleatoria. Después guarde la copia local de las claves host, al utilizar el comando `ktadd` para guardarlas en el archivo `/etc/krb5.keytab`. Cada host debe tener también el mismo archivo de configuración `/etc/krb5.conf` en sus sistemas, al especificar el servidor Kerberos y el host `kdc`.

NOTA Cuando configure Kerberos con la herramienta *Authentication*, podrá insertar el reino, el servidor *kde* y el servidor Kerberos. Las entradas predeterminadas se desplegarán al utilizar el dominio “*example.com*”. Asegúrese de especificar el reino en mayúsculas. Una nueva entrada para su reino se hará en el segmento de reinos de `/etc/krb5.conf`, mostrando una lista de las entradas `kdc` y `server` que hizo.

Firewalls

Casi todos los sistemas conectados actualmente a Internet están abiertos a que usuarios externos intenten ganar acceso sin autorización. Para ello, tratan de configurar una conexión ilegal, interceptar comunicaciones válidas de usuarios conectados remotamente al sistema o pretenden ser un usuario válido. Firewalls, cifrado y procedimientos de autenticación son formas de protegerse ante esos ataques. Una *firewall* previene cualquier intento de acceso directo sin autorización, el *cifrado* protege transmisiones de usuarios remotos autorizados y la *autenticación* verifica que un usuario pidiendo acceso tenga derecho a él. El kernel actual de Linux incorpora soporte a firewalls mediante el filtrado de paquetes Netfilter (IPtables; la versión previa, IP Chains, se usa en versiones kernel más antiguas). Para implementar una firewall, sólo indique una serie de reglas que regirán el tipo de acceso a permitir en su sistema. Si ese sistema también es una puerta de enlace para su red privada, la capacidad de firewall del sistema puede ayudar de manera efectiva a proteger la red de ataques externos.

Para proteger las comunicaciones remotas, las transmisiones simplemente se cifran. En el caso de sistemas Linux, use el conjunto de programas Secure Shell (SSH) para codificar cualquier transmisión, evitando ser leídas por cualquier otra persona. La autenticación Kerberos ofrece otro nivel de seguridad, por lo que se protegen servicios individuales, permitiendo el uso de un servicio únicamente a usuarios con autorización para el acceso. Es probable que los usuarios externos también intenten obtener acceso sin autorización a través de cualquier servicio de Internet huésped, como un sitio Web. En ese caso, puede configurar un proxy para proteger su sitio de un ataque. En el caso de sistemas Linux, utilice el software Squid, para configurar un proxy que proteja su servidor Web. En la tabla 20-1 se muestra una lista de aplicaciones de seguridad de red utilizadas de manera común en Linux.

Firewalls: IPtables, Nat e ip6tables

Una buena base para la seguridad de su red consiste en configurar el sistema Linux para funcionar como firewall para su red, protegiéndola de acceso no autorizado. Una firewall se utiliza para implementar el filtrado de paquetes o proxies. El *filtrado de paquetes* es simplemente el proceso para decidir si un paquete recibido por el host de la firewall debe pasar a la red local. El software de filtrado de paquete revisa fuente y direcciones de destino del paquete y envía el paquete, si es permitido. Aunque su sistema no sea parte de una red, pero se conecte directamente a Internet, todavía usará la característica firewall para ofrecer control de acceso a su sistema. Por supuesto, también le brinda mucha más seguridad.

Sitio Web	Aplicación de seguridad
netfilter.org	Proyecto Netfilter, IPtables y NAT
netfilter.org/ipchains	Firewall IP Chains
openssh.org	Cifrado de Secure Shell
squid-cache.org	Servidor proxy de Web Squid
web.mit.edu/Kerberos	Autenticación de red Kerberos

TABLA 20-1 Aplicaciones de seguridad de red

Con los proxies, tiene la opción de controlar el acceso a servicios específicos, como servidores Web o FTP. Necesita un proxy para cada servicio que desee controlar. El servidor Web tiene su propio proxy Web, mientras un servidor FTP tiene proxy FTP. Los proxies también se usan para almacenar datos de uso común en caché, como páginas Web, para que los usuarios no necesiten acceder al sitio donde se originan. El software de proxy que suele utilizarse en Linux es Squid.

Una tarea adicional realizada por las firewalls es la traducción de direcciones de red (NAT, Network Address Translation), que redirige paquetes a destinos apropiados. Realiza tareas como redirigir paquetes a ciertos host, reenviar paquetes a otras redes y cambiar el origen del host enviando los paquetes para implementar enmascarado IP.

***NOTA** El paquete IP Chains es el precursor de IPtables, usado en sistemas Linux con el kernel 2.2. Todavía se usa en muchos sistemas Linux. El sitio Web de Linux para IP Chains, el sucesor de ipfwadm, utilizado en versiones anteriores de Linux, es actualmente netfilter.org/ipchains. IP Chains ya no se incluye en muchas distribuciones de Linux.*

El paquete de software Netfilter implementa tareas de filtrado de paquete y NAT para el kernel 2.4 y superior de Linux. El software Netfilter es desarrollado por el Netfilter Project, del que conocerá más acerca en netfilter.org.

IPtables

El comando empleado para ejecutar tareas de filtrado de paquetes y NAT es **iptables**, y al software suele conocersele como IPtables. Sin embargo, Netfilter implementa tareas de filtrado de paquete y NAT, en forma separada, usando diferentes tablas y comandos. Una tabla almacenará el conjunto de comandos para su aplicación. Este método simplifica la tarea de filtrado de paquetes, permitiendo que IPtables realice revisiones de filtrado de paquete sin gastar recursos, que involucra traducir también las direcciones. Además, se liberan operaciones NAT de mezcla con verificaciones del filtrado de paquetes. El comando **iptables** se usa para tareas de filtrado de paquetes y NAT, pero para NAT, use la opción **-nat**. El software IPtables puede integrarse directamente en el kernel o cargarse como un módulo de kernel, **iptable_filter.o**.

ip6tables

El paquete ip6tables proporciona soporte a direccionamiento IPv6. Es idéntico a IPtables, excepto que permite usar direcciones IPv6 en vez de IPv4. Las tablas de filtrado y “planchado” tienen soporte en ip6tables, pero no para tablas NAT. Las tablas de filtrado soportan las mismas opciones y comandos que IPtables. Las tablas de planchado permitirán cambios especializados a paquetes

como los de IPtables, usando reglas PREROUTING, INPUT, OUTPUT, FORWARD y POSTROUTING. Algunas extensiones tienen etiquetas `ipv6` en sus nombres, como `ipv6-icmp`, correspondiente a la extensión `icmp` de IPtables. La extensión `ipv6headers` se utiliza para seleccionar encabezados IPv6.

Módulos

A diferencia de su predecesor, IP Chains, Netfilter está diseñado para ser modular y extensible. Las capacidades se agregan en forma de módulos como el módulo de estado, que agrega rastreo de conexiones. Casi todos los módulos se cargan a la par del servicio IPtables. Otros son opcionales; seleccione cargarlos antes de instalar las reglas. Los módulos de IPtables se ubican en `/usr/lib/versión-kernel/kernel/net/ipv4/netfilter`, donde *versión-kernel* es el número de kernel. En el caso de módulos IPv6, revise el directorio `ipv6/netfilter`. Los módulos cargados automáticamente tendrán un prefijo `ipt_` y los opcionales tendrán un prefijo `ip_`. Si está escribiendo sus propias secuencias de comandos de IPtables, tal vez deba agregar el comando `modprobe` para cargar directamente módulos opcionales.

Filtrado de paquetes

Netfilter es, en esencia, un marco conceptual para administración de paquetes, revisando aquellos en busca de protocolos de red particulares y notificando las partes del kernel que los escuchan. El sistema de selección de paquetes está integrado en el marco de Netfilter implementado por IPtables. Con IPtables, pueden configurarse diferentes tablas de reglas para seleccionar paquetes de acuerdo con distintos criterios. Actualmente, Netfilter soporta tres tablas: `filter`, `nat` y `mangle`. El filtrado de paquetes se implementa mediante una tabla de filtro que almacena reglas para aceptar o desechar paquetes. Las operaciones de traducción de direcciones de red, como enmascaramiento de IP, se implementan con la tabla NAT, que almacena reglas de enmascaramiento de IP. La tabla de `planchado` se utiliza para cambios especializados a paquetes. Los cambios se hacen a los paquetes antes de enviarlos, cuando se reciben o mientras se reenvían. Esta estructura es expansible en el sentido de que los módulos definen sus propias tablas con sus propias reglas. También mejora en gran medida la eficiencia. En vez de que todos los paquetes revisen una gran tabla, sólo acceden a la tabla de las reglas que necesitan.

Las reglas de IPtables se administran mediante el comando `iptables`. Para este comando, necesita especificar la tabla que quiere administrar. La predeterminada es la tabla `filter`, que no necesita especificarse. Puede desplegar una lista de reglas, agregadas en cualquier momento, con las opciones `-L` y `-n`, como se muestra a continuación. La opción `-n` indica que sólo use la salida numérica para direcciones IP y puertos, evitando una búsqueda DNS para nombres de host. Sin embargo, puede utilizar la opción `-L` para ver las etiquetas de puertos y nombres de host:

```
iptables -L -n
```

NOTA En los comando de IPtables, los nombres de cadena se tienen que introducir en mayúsculas, como `INPUT`, `OUTPUT` y `FORWARD`.

Cadenas

Las reglas se combinan en diferentes cadenas. El kernel utiliza cadenas para administrar paquetes recibidos y enviados. Una *cadena* es sólo una lista de reglas. Estas reglas especifican qué acción tomar en caso de que los paquetes contengan ciertos encabezados. Las reglas operan con estructura `if-then-else`.

Destino	Función
ACCEPT	Permite el paso al paquete a través del firewall.
DROP	Niega el acceso al paquete.
REJECT	Niega el acceso y notifica al emisor.
QUEUE	Envía los paquetes al espacio del usuario.
RETURN	Va hasta el final de la cadena y permite que el destino predeterminado lo procese.

TABLA 20-2 Destinos de IPtables

Si un paquete no coincide con la primera regla, entonces se revisa la siguiente regla, etc. Si un paquete no coincide con alguna regla, el kernel consulta la directiva de la cadena. En general, a partir de este punto se rechaza el paquete. Si coincide con una regla, pasa a su destino, determinando qué hacer con el paquete. Los destinos se muestran en la lista de la tabla 10-2. Si un paquete no coincide con una regla, pasa al destino predeterminado de la cadena.

Destinos

A su vez, un *destino* es otra cadena de reglas, incluso una definida por el usuario. Un paquete puede pasar por varias cadenas antes de llegar al destino. En el caso de cadenas definidas por usuarios, el destino predeterminado es siempre la siguiente regla en la cadena, desde la que se llama. Esto configura un flujo de control, similar al procedimiento o llamada a función, como el encontrado en lenguajes de programación. Cuando una regla tiene una cadena definida por el usuario como destino, ésta se ejecuta al activarse. Si no coincide una regla, la ejecución regresa a la siguiente regla en la cadena donde se origina.

***SUGERENCIA** Los destinos y opciones especializados se agregan por medio de parches de kernel proporcionados por el sitio Netfilter. Por ejemplo, el parche SAME regresa la misma dirección para todas las conexiones. Una opción patch-o-matic para el archivo make de Netfilter parchará su código fuente del kernel, agregando soporte para el nuevo destino y las opciones. Luego podrá reconstruir e instalar su kernel.*

Cadenas de firewall y NAT

El kernel usa tres cadenas de firewall: INPUT, OUTPUT y FORWARD. Cuando un paquete se recibe a través de una interfaz, la cadena INPUT se maneja para determinar qué hacer con éste. El kernel utiliza luego su información de ruta para decidir a dónde enviarlo. Si el kernel envía el paquete a otro host, se revisa la cadena FORWARD. Antes de que el paquete realmente se envíe, se implementa PREROUTING para manejar enmascaramiento y modificaciones de dirección de paquete. Las cadenas integradas de Netfilter se muestran en la lista de la tabla 20-3.

Adición y cambio de reglas

Las reglas de cadena se agregan y modifican con los comandos de **iptables**. Un comando **iptables** incluye el comando **iptables**, seguido por un argumento denotando el comando a ejecutarse. Por ejemplo, **iptables -A** es el comando para agregar una nueva regla, mientras **iptables -D** es el comando para eliminar una regla. En la tabla 20-4 se muestra una lista de comandos de **iptables**.

Cadena	Descripción
INPUT	Reglas para paquetes entrantes
OUTPUT	Reglas para paquetes salientes
FORWARD	Reglas para paquetes reenviados
PREROUTING	Reglas para redirigir o modificar paquetes entrantes, sólo tabla NAT
POSTROUTING	Reglas para redirigir o modificar paquetes salientes, sólo tabla NAT

TABLA 20-3 Cadenas integradas de Netfilter

El siguiente comando sólo muestra una lista de cadenas, junto con las reglas definidas para su sistema. La salida muestra los valores predeterminados creados por los comandos **iptables**.

```
iptables -L -n
Chain input (policy ACCEPT):
Chain forward (policy ACCEPT):
Chain output (policy ACCEPT):
```

Para agregar una nueva regla a la cadena, emplee **-A**. Utilice **-D** para eliminarla y **-R** para reemplazarla. Después del comando, se muestra una lista de la cadena a la que aplica la regla, como INPUT, OUTPUT o FORWARD, o una cadena definida por el usuario. Después, despliega una lista con diferentes opciones especificando las acciones que quiere se tomen (casi todas son las mismas usadas para IP Chains, con ciertas excepciones).

Opción	Función
-A <i>cadena</i>	Adjunta una regla a la cadena.
-D <i>cadena</i> [<i>numderegla</i>]	Elimina reglas de coincidencia de una cadena. Elimina la regla <i>numderegla</i> (1 = la primera) de la <i>cadena</i> .
-I <i>cadena</i> [<i>numderegla</i>]	Inserta la <i>cadena</i> como <i>numderegla</i> (1 = la primera).
-R <i>cadena numderegla</i>	Reemplaza la regla <i>numderegla</i> (1 = la primera) en la <i>cadena</i> .
-L [<i>cadena</i>]	Muestra una lista de reglas en <i>cadena</i> o en todas las cadenas.
-E [<i>cadena</i>]	Cambia el nombre una cadena.
-F [<i>cadena</i>]	Elimina (limpia) todas las reglas de <i>cadena</i> o de todas las cadenas.
-R <i>cadena</i>	Reemplaza una regla; las reglas se numeran a partir del 1.
-Z [<i>cadena</i>]	Marcadores en cero en la <i>cadena</i> o en todas las cadenas.
-N <i>cadena</i>	Crea una nueva cadena definida por el usuario.
-X <i>cadena</i>	Elimina una cadena definida por el usuario.
-P <i>destino cadena</i>	Cambia la directiva en la <i>cadena</i> al <i>destino</i> .

TABLA 20-4 Comandos de IPtables

La opción **-s** especifica la dirección de origen adjunta al paquete, **-d** indica la dirección de destino y **-j** especifica el destino de la regla. El destino ACCEPT permitirá que un paquete pase. La opción **-i** ahora indica el dispositivo de entrada usado sólo en las cadenas INPUT y FORWARD. La opción **-o** indica el dispositivo de salida y se utiliza sólo con las cadenas OUTPUT y FORWARD. En la Tabla 20-5 se muestra una lista de varias opciones.

Opción	Función
-p [!] <i>protocolo</i>	Especifica un protocolo, como TCP, UDP, ICMP o ALL.
-s [!] <i>dirección</i> [/máscara] [!] [<i>puerto</i> : <i>puerto</i>]	Especifica una dirección de origen que habrá de coincidir. Con el argumento <i>puerto</i> , especifique el puerto.
--sport [!] [<i>puerto</i> : <i>puerto</i>]	Especifica el puerto de origen. Especifique el rango de puertos al utilizar dos puntos, <i>puerto:puerto</i> .
-d [!] <i>dirección</i> [/máscara] [!] [<i>puerto</i> : <i>puerto</i>]	Especifica la dirección de destino que habrá de coincidir. Con el argumento <i>puerto</i> , especifique el puerto.
--dport [!] [<i>puerto</i> : <i>puerto</i>]	Especifica el puerto de destino.
--icmp-type [!] <i>nombredetipo</i>	Especifica un tipo ICMP.
-i [!] <i>nombre</i> [+]	Especifica una interfaz de entrada de red al utilizar su nombre (por ejemplo, eth0). El símbolo + funciona como comodín. El signo +, adjunto al final del nombre, coincide con todas las interfaces con ese prefijo (eth+ coincide con todas las interfaces Ethernet). Sólo se utiliza con la cadena INPUT.
-j <i>destino</i> [<i>protocolo</i>]	Especifica el destino para una regla (especifique [<i>port</i>] para el destino REDIRECT).
--to-source [!] [-< <i>direccionip</i> >][: <i>puerto</i> - <i>puerto</i>]	Utilizada con el destino SNAT, vuelve a escribir los paquetes con la nueva dirección IP de origen.
--to-dirección [!] [<i>protocolo</i>]	Utilizada con el destino DNAT, vuelve a escribir los paquetes con la nueva dirección IP de destino.
-n	Especifica la salida numérica de direcciones y puertos, utilizada con -L .
-o [!] <i>nombre</i> [+]	Especifica una interfaz de salida de red al utilizar su nombre (por ejemplo, eth0). Sólo se utiliza con las cadenas FORWARD y OUTPUT.).
-t <i>tabla</i>	Especifica una tabla que habrá de utilizarse, como en -t nat para la tabla NAT.
-v	Modo extenso, muestra detalles de la regla, se utiliza con -L .
-x	Expande los números (despliega los valores exactos), se utiliza con -L .
[!] -f	Equipara del segundo al último fragmento de un paquete fragmentado.
[!] -v	Imprime la versión del paquete.
!	Niega una opción o dirección.

TABLA 20-5 Opciones de IPtables (continúa)

Opción	Función
-m	Especifica un módulo que habrá de utilizarse, como state.
--state	Especifica opciones para el módulo state como NEW, INVALID, RELATED y ESTABLISHED. Se utiliza para detectar el estado del paquete. REW hace referencia a paquetes SYN (nuevas conexiones).
--syn	Paquetes SYN, nuevas conexiones
--tcp-flags	Marcas de TCP: SYN, ACK, FIN, RST, URG, PS y ALL para todas las marcas.
--limit	Opción para el módulo limit (-m limit). Se utiliza para controlar la cantidad de coincidencias, al coincidir con un número de veces por segundo.
--limit-burst	Opción para el módulo limit (-m limit). Especifica el máximo de apariciones antes de alcanzar el límite. Se utiliza para controlar ataques de negación de servicio.

TABLA 20-5 Opciones de IPtables (continuación)

Opciones de IPtables

El paquete IPtables está diseñado para ser extensible, y es posible incluir en él opciones con selección de criterios. Por ejemplo, la extensión TCP incluye la opción `-syn`, para revisar paquetes SYN. La extensión ICMP proporciona la opción `--icmp-type` para especificar que los paquetes ICMP se utilicen en operaciones de ping. El límite de expansión incluye la opción `--limit`, con que se limita el número máximo de paquetes coincidentes en un periodo específico, como un segundo.

En el siguiente ejemplo, el usuario agrega una regla a la cadena INPUT para aceptar todos los paquetes originados en la dirección 192.168.0.55. Cualquier paquete que se reciba (**INPUT**) cuya dirección de fuente (`-s`) coincida con 192.168.0.55 se acepta y deja pasar (`-j ACCEPT`):

```

iptables -A INPUT -s 192.168.0.55 -j ACCEPT

```

Aceptación y rechazo de paquetes: DROP y ACCEPT

Existen dos destinos integrados, DROP y ACCEPT. Otros son cadenas definidas por el usuario o extensiones agregadas, como REJECT. Dos destinos especiales usados para administrar cadenas, RETURN y QUEUE. RETURN indica el final de una cadena y regresa la cadena desde donde empezó. QUEUE se utiliza para enviar paquetes al espacio de usuario.

```

iptables -A INPUT -s www.mibasura.com -j DROP

```

Puede convertir una regla en su inverso con el símbolo `!` Por ejemplo, para aceptar todos los paquetes entrantes, excepto los de una dirección específica, coloque un símbolo `!` antes de la opción `-s` y esa dirección. En el siguiente ejemplo se aceptarán todos los paquetes, excepto los de la dirección IP 192.168.0.45:

```

iptables -A INPUT -j ACCEPT ! -s 192.168.0.45

```

Puede especificar una dirección individual usando su nombre de dominio o número IP. En el caso de un rango de direcciones, use el número IP de su red y máscara IP de ésta. La máscara IP es un número IP o sólo el número de bits integrando la máscara. Por ejemplo, todas las direcciones de red 192.168.0 se representan con 192.168.0.0/225.255.255.0 o 192.168.0.0/24. Para especificar cualquier dirección, utilice 0.0.0.0/0.0.0.0 o sólo 0/0. Como opción predeterminada, las reglas hacen referencia a cualquier dirección, si no existe una especificación **-s** o **-d**. En el siguiente ejemplo se aceptan mensajes entrantes de cualquier (origen) host en la red 192.168.0.0 y que van (destino) a cualquier lugar (la opción **-d** se deja fuera o puede escribirse como **-d 0/0**):

```
iptables -A INPUT -s 192.168.0.0/24 -j ACCEPT
```

Las reglas de IPtables suelen aplicarse a una interfaz de red específica, como Ethernet usada para concertar a Internet. En el caso de haber un solo sistema conectado a Internet, tendrá dos interfaces, una para su conexión de Internet y una interfaz loopback (**lo**) para conexiones internas entre usuarios de su sistema. Se hace referencia a la interfaz de red para Internet mediante el nombre de dispositivo de su interfaz. Por ejemplo, se haría referencia a una tarjeta Ethernet con el nombre de dispositivo **/dev/eth0** en **eth0**. Un módem usando protocolos PPP con el nombre de dispositivo **/dev/ppp0**, podría tener el nombre **ppp0**. En las reglas de IPtables, se utiliza la opción **-i** para indicar el dispositivo de entrada; sólo se usa con las cadenas INPUT y FORWARD. La opción **-o** indica dispositivo de salida y usado sólo con cadenas OUTPUT y FORWARD. Después, las reglas se aplican en paquetes que llegan y salen de dispositivos de red particulares. En los siguientes ejemplos, la primera regla hace referencia al dispositivo Ethernet **eth0**, y la segunda al host local:

```
iptables -A INPUT -j DROP -i eth0 -s 192.168.0.45
iptables -A INPUT -j ACCEPT -i lo
```

Cadenas definidas por el usuario

Con IPtables, las cadenas FORWARD e INPUT se evalúan por separado. Una no se alimenta de la otra. Esto significa que si quiere bloquear por completo ciertas direcciones para no pasar a su sistema, necesitará agregar la regla FORWARD y una regla INPUT para éstas.

```
iptables -A INPUT -j DROP -i eth0 -s 192.168.0.45
iptables -A FORWARD -j DROP -i eth0 -s 192.168.0.45
```

Un método común para reducir reglas INPUT y FORWARD repetidas es crear una cadena de usuario donde las cadenas INPUT y FORWARD se alimenten entre sí. Una cadena de usuario se define con la opción **-N**. En el siguiente ejemplo se muestra el formato básico para esta organización. Se crea una nueva cadena llamada entrante (puede tener cualquier nombre que elija). Las reglas que defina para sus cadenas INPUT y FORWARD ahora se establecen para la cadena entrante. Las cadenas INPUT y FORWARD utilizan entonces la cadena entrante como destino, saltándose directamente a ésta y utilizando las reglas para procesar cualquier paquete que reciban.

```
iptables -N entrante

iptables -A entrante -j DROP -i eth0 -s 192.168.0.45
iptables -A entrante -j ACCEPT -i lo
iptables -A FORWARD -j entrante
iptables -A INPUT -j entrante
```

Paquetes ICMP

Las firewalls suelen bloquear ciertos mensajes de protocolo de mensajes de control de Internet (ICPM, Internet Control Message Protocol). Para redirigir mensajes, ICMP, en particular, toma control de sus tareas de enrutamiento. Sin embargo, necesita habilitar algunos mensajes ICMP como los necesarios para ping, traceroute y operaciones de destino inalcanzable, de manera particular. En casi todos los casos, siempre necesita asegurarse de que estén permitidos los paquetes de destino inalcanzable; de lo contrario, podrían suspenderse las consultas de nombre de dominio. Algunos tipos de paquetes ICMP más comunes se muestran en la lista de la tabla 20-6. Habilite un tipo de paquete ICMP con la opción `--icmp-type`, tomando como argumento número o nombre que representa el mensaje. En los siguientes ejemplos se habilita el uso de respuesta de eco, solicitud de eco y destino inalcanzable, con los números 0, 8 y 3:

```
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp -type echo-reply -d 10.0.0.1
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type echo-request -d 10.0.0.1
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type destination-unreachable -d 10.0.0.1
```

Su lista de reglas se verá así:

ACCEPT	ICMP	--	0.0.0.0/0	10.0.0.1	ICMP TYPE 0
ACCEPT	ICMP	--	0.0.0.0/0	10.0.0.1	ICMP TYPE 8
ACCEPT	ICMP	--	0.0.0.0/0	10.0.0.1	ICMP TYPE 3

Las operaciones de ping necesitan controlarse aún más para evitar la amenaza de seguridad llamada ping de la muerte. Puede hacer esto de varias formas. Una forma consiste en denegar cualquier fragmento ping. Los paquetes ping suelen ser muy pequeños. Bloquee los ataques ping de la muerte al denegar cualquier paquete ICMP que sea un fragmento. Utilice la opción `-f` para indicar los fragmentos.

```
iptables -A INPUT -p incmp -j DROP -f
```

Otra forma consiste en limitar el número de coincidencias recibidas por paquetes de ping. Utilice el módulo `limit` para controlar el número de coincidencias en la operación de ping de ICMP. Utilice `-m limit` para emplear el módulo de límite y `--limit` para especificar número de coincidencias permitidas. `1/s` permitirá una coincidencia por segundo.

```
iptables -A FORWARD -ip icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT
```

Número	Nombre	Necesario para
0	echo-reply	ping
3	destination-unreachable	Cualquier tráfico TCP/UDP
5	redirect	Enrutamiento, si no existe un daemon de enrutamiento en ejecución
8	echo-request	ping
11	time-exceeded	traceroute

TABLA 20-6 Paquetes de ICMP comunes

Control del acceso a puerto

Si su sistema es host de un servicio de Internet, como un servidor Web o FTP, use IPtables para controlar el acceso a éstos. Especifique un servicio particular usando las opciones de puerto de origen (**--sport**) o destino (**--dport**) con el puerto usando el servicio. IPtables permite usar nombres para puertos, como **www** para un puerto de servidor Web. Los nombres de servicios y puertos que utilizan se muestran en una lista en el archivo **/etc/services**, que correlaciona puertos con servicios particulares. En el caso de un servidor de nombre de dominio, el puerto sería **domain**. También utilice el número de puerto, si quiere, poniendo dos puntos antes de él. En el siguiente ejemplo, se aceptan todos los mensajes al servidor Web ubicado en 192.168.0.43:

```
iptables -A INPUT -d 192.168.0.43 --dport www -j ACCEPT
```

También puede usar referencias a puertos para proteger ciertos servicios y negar otros. Este método suele utilizarse si está diseñando una firewall más abierta a Internet, permitiendo a los usuarios tener mayor uso de conexiones de Internet. Ciertos servicios que sabe son dañinos, como Telnet y NTP, pueden negarse selectivamente. Por ejemplo, para negar cualquier tipo de operación Telnet en su firewall, rechace todos los paquetes procedentes del puerto Telnet, 23. Para proteger operaciones NFS, niegue el acceso al puerto utilizado por el portmapper, 111. Utilice número o nombre de puerto.

```
# deny outside access to portmapper port on firewall.
iptables -A arriving -j DROP -p tcp -i eth0 --dport 111
# deny outside access to telnet port on firewall.
iptables -A arriving -j DROP -p tcp -i eth0 --dport telnet
```

La lista de reglas se verá así:

```
DROP      tcp  --  0.0.0.0/0    0.0.0.0/0    tcp dpt:111
DROP      tcp  --  0.0.0.0/0    0.0.0.0/0    tcp dpt:23
```

Un problema de seguridad relacionado con puertos es el acceso a su servidor X en los puertos XFree86 que van de 6000 a 6009. En una firewall relativamente abierta, estos puertos pueden usarse de manera ilegal para acceder a su sistema a través de su servidor X. Un rango de puertos se especifica con dos puntos, como en 6000:6009. También puede utilizar x11 para el primer puerto, x11:6009. Las sesiones en el servidor X se aseguran al utilizar SSH, que suelen acceder al servidor X en el puerto 6010.

```
iptables -A arriving -j DROP -p tcp -i eth0 --dport 6000:6009
```

Aquí se muestran los puertos revisados y sus etiquetas:

Servicio	Número de puerto	Etiqueta de puerto
Auth	113	auth
Finger	79	finger
FTP	21	ftp
NTP	123	ntp
Portmapper	111	sunrpc
Telnet	23	telnet
Servidor Web	80	www
XFree86	6000:6009	x11:6009

Estados de paquete: rastreo de conexiones

Una de las extensiones más útiles es state, que detecta con facilidad información de rastreo para un paquete. El rastreo de conexiones mantiene información acerca de una conexión como origen, destino y puerto. Proporciona medios efectivos para determinar qué paquetes pertenecen a una conexión establecida o relacionada. Para utilizar el rastreo de conexiones, especifique el módulo state con **-m state**. Luego use la opción **--state**. Aquí especifique cualquiera de los siguientes estados:

Estado	Descripción
NEW	Paquete que crea una conexión nueva
ESTABLISHED	Paquete que pertenece a una conexión existente
RELATED	Paquete relacionado con una conexión, pero no es parte de ella, como un error ICMP o paquete que establece una conexión de datos FTP
INVALID	Paquete que, por alguna razón, no se identifica
RELATED+REPLY	Paquete relacionado con una conexión establecida pero no es parte de una directamente

Si está diseñando una firewall que intente proteger su red local de cualquier intento de penetración desde una red externa, tal vez quiera restringir los paquetes entrantes. Simplemente negar el acceso a todos los paquetes es imposible, porque los usuarios conectados a servidores externos (digamos, Internet) deben recibir información de éstos. En cambio, puede negar el acceso a un tipo determinado de paquete utilizado para iniciar una conexión. La idea es que un atacante debe iniciar una conexión desde el exterior. Los encabezados de este tipo de paquetes tienen su conjunto de bits SYN, FIN y ACK vacíos. El estado NEW del módulo state coincide con cualquier paquete SYN. Al especificar un destino DROP para tales paquetes, niega el acceso a cualquier paquete parte de un intento de hacer conexiones con su sistema. Cualquiera persona que intente conectarse a su sistema desde el exterior será incapaz de hacerlo. Los usuarios de su sistema local que hayan iniciado conexiones con host externos, todavía pueden comunicarse con éstos. En el siguiente ejemplo se rechaza cualquier paquete que busque crear una conexión nueva en la interfaz **eth0**, aunque se aceptarán en cualquier otra interfaz:

```
iptables -A INPUT -m state --state NEW -i eth0 -j DROP
```

El operador **!** se utiliza en el dispositivo **eth0** combinado con un destino ACCEPT para crear una regla que aceptará cualquier paquete nuevo, excepto los del dispositivo **eth0**. Si éste es el único conectado a Internet, todavía bloqueará de manera efectiva el acceso externo. Al mismo tiempo, una operación de ingreso para otros dispositivos como su localhost es libre de establecer conexiones. En general supone que una regla más posterior, como una directiva de cadena rechazará los paquetes restantes.

```
iptables -A INPUT -m state --state NEW ! -i eth0 -j ACCEPT
```

En el siguiente ejemplo se aceptará cualquier paquete que sea parte de una conexión establecida o que esté relacionado con esa conexión en la interfaz **eth0**:

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

SUGERENCIA Utilice la herramienta iptstate para desplegar la tabla state actual.

Rastreo de conexión especializada: ftp, irc, Amanda, tftp

Para rastrear ciertos tipos de paquetes, IPtables utiliza módulos de rastreo de conexión especializados. Se trata de módulos opcionales que debe cargar de manera manual. Para rastrear una conexión FTP pasiva, tendrá que cargar el módulo `ip_conntrack_ftp`. Para agregar soporte a la tabla NAT, también tendrá que cargar el módulo `ip_nat_ftp`. En el caso de conexiones IRC, use `ip_conntrack_irc` e `ip_nat_irc`. Existen módulos correspondientes para Amanda (el servidor de respaldo) y TFTP (TrivialFTP).

Si está escribiendo sus propias secuencias de comandos IPtables, deberá agregar los comandos **modprobe** para cargar los módulos.

```
modprobe ip_conntrack ip_conntrack_ftp ip_nat_ftp
modprobe ip_conntrack_amanda ip_nat_amanda
```

Traducción de direcciones de red (NAT)

La traducción de direcciones de red (NAT, Network Address Translation) es el proceso en que un sistema cambiará el destino u origen de los paquetes conforme atraviesan el sistema. Un paquete atravesará varios sistemas vinculados en una red antes de llegar a su destino final. Generalmente, sólo pasará el paquete. Sin embargo, si uno de estos sistemas realiza NAT en un paquete, cambiará origen o destino. Un paquete enviado a un destino particular tiene sus direcciones de destino cambiadas. Para que esto funcione, el sistema también necesita recordar tales cambios para que origen y destino de cualquier paquete de respuesta se modifiquen cuando sean enviados de regreso a las direcciones originales del paquete al que está respondiendo.

NAT a menudo se utiliza para ofrecer acceso a sistemas conectados a Internet a través de sólo una dirección IP. Tal es el caso de características de red como enmascaramiento de IP, soporte a varios servidores y proxy transparente. Con el enmascaramiento de IP, las operaciones de NAT cambiarán destino y origen de un paquete atravesando una firewall y una puerta de enlace vinculando a Internet con equipos de una red local. La puerta de enlace tiene una sola dirección IP que otras computadoras locales utilizan mediante operaciones NAT. Si tiene varios servidores pero sólo una dirección IP, use operaciones NAT para enviar paquetes a servidores alternos. También puede usar operaciones NAT para que su dirección IP haga referencia a una aplicación de servidor particular como un servidor Web (proxy transparente). Las tablas NAT no se implementan para ip6tables.

Adición de reglas de NAT

Las reglas de selección de paquetes para operaciones NAT se agregan en la tabla NAT administrada por el comando **iptables**. Para agregar reglas a la tabla NAT, debe especificar la tabla NAT con la opción **-t**. Aunque para agregar una regla a la tabla NAT, deberá especificar la tabla NAT con la opción **-t nat** como se muestra aquí:

```
iptables -t nat
```

Con la opción **-L**, puede desplegar una lista de reglas agregadas a la tabla NAT:

```
iptables -t nat -L -n
```

La adición de la opción **-n** mostrará una lista de direcciones IP y puertos en forma numérica. Esto hará más rápido el proceso de la lista, porque IPtables no intentará hacer una búsqueda DNS para determinar el nombre de host de la dirección IP.

Destinos y cadenas de NAT

Además, existen dos tipos de operaciones NAT: NAT de origen, especificada como destino de SNAT, y NAT de destino, especificada como destino de DNAT. El destino de SNAT se usa para reglas modificando direcciones de origen y destino DNAT, para las que alteran las direcciones de destino.

El kernel utiliza tres cadenas en la tabla NAT para operaciones NAT: PREROUTING, POSTROUTING y OUTPUT. PREROUTING se maneja para reglas NAT de destino (DNAT, Destiny NAT), paquetes que llegan. POSTROUTING se utiliza para reglas de NAT de origen (SNAT, Source NAT), para paquetes que salen. OUTPUT se utiliza para reglas de NAT de destino, en el caso de paquetes generados localmente.

Como en el caso del filtrado de paquetes, puede especificar las direcciones de origen (**-s**) y destino (**-d**), además de los dispositivos de entrada (**-i**) y salida (**-o**). La opción **-j** especificará un destino como MASQUERADE. El enmascaramiento de IP se implementa al agregar la regla MASQUERADE a la cadena POSTROUTING:

```
# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Para cambiar la dirección de origen de un paquete saliendo de su sistema, use la regla POSTROUTING con el destino de SNAT. Para el destino de SNAT, use la opción **--to-source** para especificar la dirección de origen:

```
# iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source 192.168.0.4
```

Para cambiar una dirección de destino de paquetes llegando a su sistema, use la regla PREROUTING con el destino DNAT y la opción **--to-destination**:

```
# iptables -t nat -A POSTROUTING -i eth0 \
-j DNAT --to-destination 192.168.0.3
```

La especificación de un puerto permite cambiar destinos para paquetes llegando a un puerto determinado. En efecto, esto permite implementar el reenvío de puerto. En el siguiente ejemplo, cada paquete llegando al puerto 80 (el de servicio Web) se redirige a 10.0.0.3, en este caso un servidor Web ejecutándose en el sistema.

```
# iptables -t nat -A POSTROUTING -i eth0 -dport 80 \
-j DNAT --to-destination 10.0.0.3
```

Con los destinos TOS y MARK, puede “planchar” el paquete para controlar su enrutamiento o prioridad. Un destino TOS configura el tipo de servicio para un paquete, lo que puede establecer la prioridad mediante criterios como servicio normal, minimizar costo y maximizar rendimiento, entre otros.

A continuación se muestran los destinos válidos sólo para la tabla NAT:

SNAT	Modifica direcciones de origen, se utiliza la opción <code>--to-source</code> para especificar nuevas direcciones de origen.
DNAT	Modifica direcciones de destino, se utiliza la opción <code>--to-destination</code> para especificar nuevas direcciones de destino.
REDIRECT	Redirige un paquete.
MASQUERADE	Enmascaramiento IP
MIRROR	Invierte el origen y el destino y se envía de regreso al emisor.
MARK	Modifica el campo Mark para controlar el enrutamiento de mensajes.

Redireccionamiento NAT: proxies transparentes

Las tablas NAT se utilizan para implementar cualquier tipo de redireccionamiento de paquete, un proceso transparente para el usuario. El redireccionamiento suele utilizarse para implementar un proxy transparente. El redireccionamiento de paquetes se lleva a cabo con el destino REDIRECT. Con proxies transparentes, los paquetes recibidos se dirigen automáticamente a un servidor proxy. Por ejemplo, los paquetes llegando al puerto de servicio Web, 80, se redirigen al puerto de servicio proxy Squid, en general, 3128. Esto incluye un comando para redirigir un paquete, al utilizar el destino REDIRECT en la cadena PREROUTING:

```
iptables -t nat -A PREROUTING -i eth1 --dport 80 -j REDIRECT --to-port 3128
```

Planchado de paquetes: la tabla mangle

La tabla de *planchado de paquetes* se utiliza realmente para modificar información de paquete. Las reglas aplicadas específicamente a esta tabla, con frecuencia están diseñadas para controlar el comportamiento mundano de los paquetes, como enrutamiento, tamaño de conexión y prioridad. Las reglas realmente modifican un paquete, en vez de sólo redirigirlo o detenerlo, se utiliza sólo en la tabla mangle. Por ejemplo, el destino TPS se utiliza directamente en la tabla mangle para cambiar el campo Type of Service, con el fin de modificar la prioridad de un paquete. Un destino TCPMSS se establece para modificar el tamaño permitido de paquetes para una conexión. El destino ECN le permite trabajar alrededor de hoyos negros de ECN, mientras el destino DSCP permitirá cambiar bits de DSCP. Varias extensiones como ROUTE cambiarán un paquete, en este caso, reescribiendo su destino en vez de sólo redirigirlo.

La tabla mangle se indica con la opción `-t mangle`. Use el siguiente comando para ver qué cadenas se muestran en la lista de su tabla mangle:

```
iptables -t mangle -L
```

Aquí se muestran varios destinos de tablas mangle:

TOS	Modifica el campo Type of Service para administrar la prioridad de paquete.
TCPMSS	Modifica el tamaño permitido de los paquetes para una conexión, permitiendo transmisiones más grandes.
ECN	Elimina información de hoyos negros de ECN
DSCP	Cambia bits DSCP
ROUTE	Extensión TARGET para modificar información de destino en el paquete.

NOTA El paquete *IPtables* está diseñado para ser extensible, lo que permite la fácil adición de destinos personalizados. Esto incluye la aplicación de parches al kernel y su reconstrucción. Visite netfilter.org para conocer más detalles, junto con una lista de destinos extendidos.

Secuencias de comandos de IPtables

Aunque puede insertar reglas de IPtables desde la línea de comandos de shell, cuando apaga su sistema, estos comandos se perderán. Lo más probable es que necesite colocar sus reglas IPtables en una secuencia de comandos que se ejecuta directamente. De esta forma puede editar y administrar un conjunto complejo de reglas, agregando comentarios y administrando sus órdenes.

Un ejemplo de secuencia de comandos de IPtables: IPv4

Ahora tiene suficiente información para crear una secuencia de comandos IPtables simple, con protección básica para un solo sistema conectado a Internet. La siguiente secuencia de comandos, **mifiltro**, proporciona un proceso de filtrado de IPtables para proteger una red local y un sitio Web de ataques externos. En este ejemplo se utilizan IPtables y direccionamiento de IPv4. Para el direccionamiento de IPv6 utilizaría *ip6tables*, cuyos comandos correspondientes, excepto las reglas NAT, se implementan como reglas *mangle*.

La secuencia de comandos configura una firewall simple para una red privada (consulte HOWTO de IPtables para revisar un ejemplo más complejo). Si tiene una red local, adapte la secuencia de comandos a ésta. En esta configuración, se bloquen todos los accesos remotos iniciados desde el exterior, pero se permiten comunicaciones de dos vías para conexiones que los usuarios de la red hacen con sistemas externos. En este ejemplo, la firewall del sistema funciona como puerta de enlace para redes privadas cuya dirección de red es 192.168.0.0 (véase la figura 20-1). La dirección

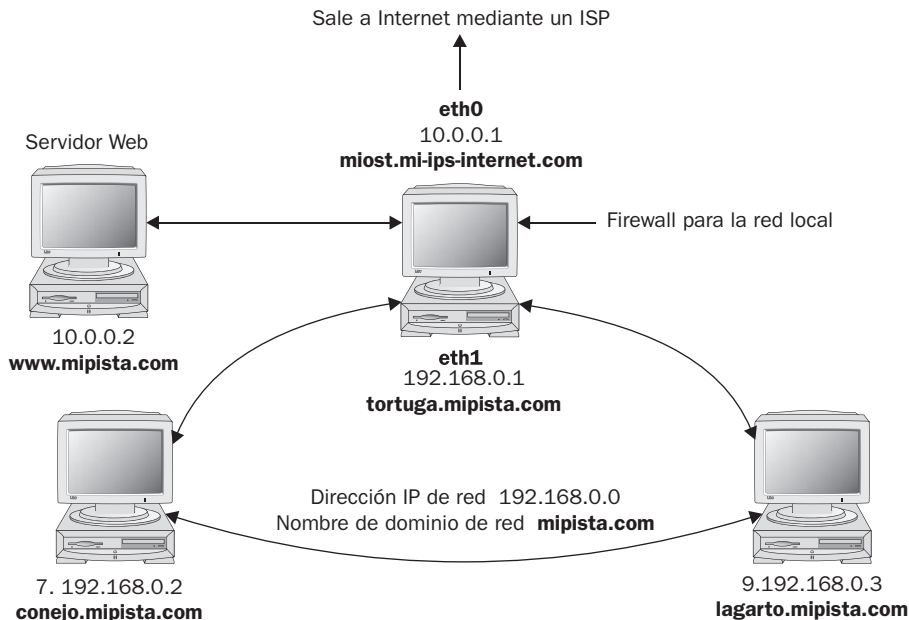


FIGURA 20-1 Una red con una firewall

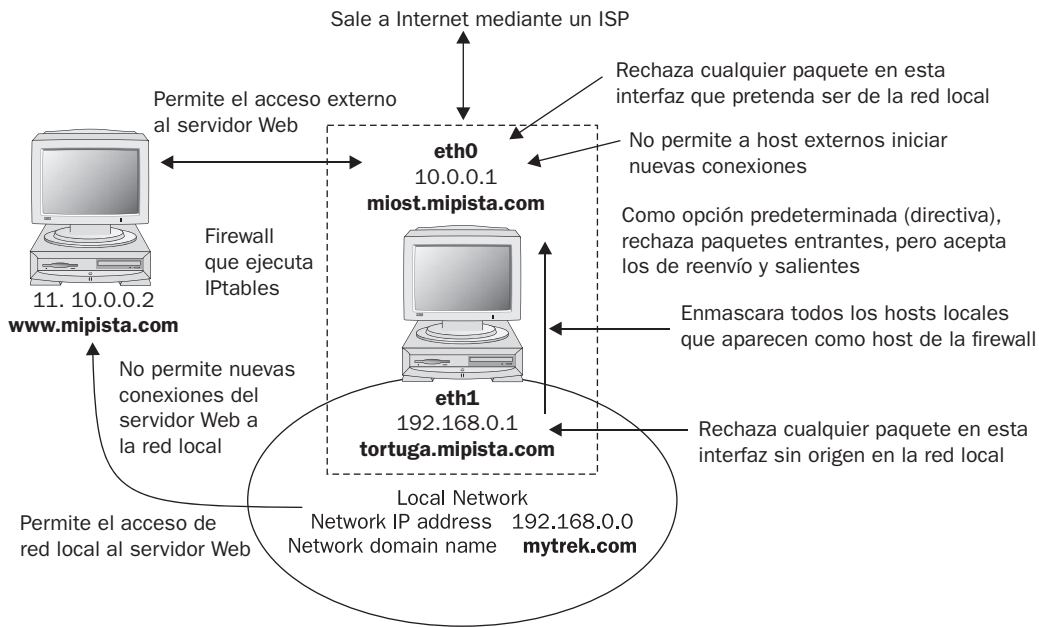


FIGURA 20-2 Las reglas de la Firewall aplicadas a un ejemplo de red local

de Internet es, para este ejemplo, 10.0.0.1. El sistema tiene dos dispositivos Ethernet: uno para la red privada (**eth1**) y otro para Internet (**eth0**). La puerta de enlace de la firewall del sistema también soporta un servidor Web en la dirección 10.0.0.2. Las entradas en este ejemplo, que son demasiado grandes como para caber en una línea, se continúan en una segunda línea, con la nueva línea citada con una diagonal invertida.

Las reglas básicas, como se aplican a diferentes partes de la red, se ilustran en la figura 20-2.

```
mifiltro
# La dirección IP del sistema de puerta de enlace de la firewall es 10.0.0.1
# empleando
# el dispositivo de Ethernet eth0.
# La dirección de la red privada es 192.168.0.0 empleando el dispositivo de
# Ethernet eth1
# La dirección del sitio Web es 10.0.0.2
# se deshabilita el reenvío IP
echo 0 > /proa/sys/net/ipv4/ip_forward
# Limpia las reglas de cadena
iptables -F INPUT
iptables -F OUTPUT
iptables -F FORWARD
# establece las reglas (directivas) predeterminadas.
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD ACCEPT
```

```
# Engaño de IP, se rechaza cualquier paquete de la red interna
#que tenga una dirección de origen externa.
iptables -A INPUT -j LOG -i eth1 \! -s 192.168.0.0/24
iptables -A INPUT -j DROP -i eth1 \! -s 192.168.0.0/24
iptables -A FORWARD -j DROP -i eth1 \! -s 192.168.0.0/24
# Engaño de IP, se rechaza cualquier paquete externo (que no sea de eth1)
#que tenga la dirección de origen de la red internar
iptables -A INPUT -j DROP \! -i eth1 -s 192.168.0.0/24
iptables -A FORWARD -j DROP \! -i eth1 -s 192.168.0.0/24
# Engaño de IP, se rechaza cualquier paquete del exterior con dirección localhost
# (los paquetes no están en la interfaz local (en eth0 o eth1)
# que tengan la dirección de origen localhost)
iptables -A INPUT -j DROP -i \! lo -s 127.0.0.0/255.0.0.0
iptables -A FORWARD -j DROP -i \! lo -s 192.168.0.0/255.0.0.0

# permite todos los mensajes entrantes para usuarios del sistema de su firewall
iptables -A INPUT -j ACCEPT -i lo

# permite la comunicación con el servidor web (dirección 10.0.0.2), port www
iptables -A INPUT -j ACCEPT -p tcp -i eth0 --dport www -s 10.0.0.2
# Permite conexiones establecidas de servidores Web con la red interna
iptables -A INPUT -m state --state ESTABLISHED,RELATED -i eth0 -p tcp
--sport www -s 10.0.0.2 -d 192.168.0.0/24 -j ACCEPT
# Evita nuevas conexiones de los servidores Web a la red interna
iptables -A OUTPUT -m --state NEW -o eth0 -p tcp --sport
www -d 192.168.0.1.0/24 -j DROP

# permite comunicaciones externas y relacionadas con su sistema
# permite comunicación externa con la firewall, excepto para paquetes ICMP
iptables -A INPUT -m state --state ESTABLISHED,RELATED -i eth0 -p \! icmp -j
ACCEPT
# evita conexiones iniciadas en el exterior
iptables -A INPUT -m state --state NEW -i eth0 -j DROP
iptables -A FORWARD -m state --state NEW -i eth0 -j DROP
# permite toda la comunicación con la firewall en eth1 desde la red local
iptables -A INPUT -j ACCEPT -p all -i eth1 -s 192.168.0.0/24
# Configura el enmascaramiento para para permitir que las máquinas internas
accedan a la red externa
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
# Acepta mensajes de ping de ICMP y de destino inalcanzable
# Otros serán rechazados por la directiva INPUT y OUTPUT DROP
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type echo-reply
-d 10.0.0.1
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type echo-request
-d 10.0.0.1
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type destination-
unreachable -d 10.0.0.1
# Habilita el reenvío IP.
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Al principio, en la secuencia de comandos se limpian sus IPtables actuales con la opción de limpieza (**-F**) y después se establecen las directivas (destinos predeterminados) para reglas no definidas por los usuarios. El reenvío de IP también se debe desactivar mientras las reglas de cadena se establecen

```
echo 0 > /proc/sys/net/ipv4/ip_forward
```

Directiva de rechazo

En primer lugar, se establece una directiva DROP para cadenas IP integradas INPUT y FORWARD. Esto significa que si un paquete no cumple con un criterio en alguna de las reglas para dejarlo pasar, se rechazará.

Después se rechazan los ataques de engaño de IP y cualquier intento desde el exterior de iniciar conexiones (paquetes SYN). Los intentos de conexión desde el exterior también se registran. Esta es una configuración muy básica que se refina de manera sencilla a sus propias necesidades al agregar reglas de IPtables.

```
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD ACCEPT
```

Engaño de IP

Una forma de proteger la red privada del engaño de IP de cualquier paquete, consiste en revisar cualquier dirección externa en el dispositivo Ethernet dedicado a la red privada. En este ejemplo, cualquier paquete en el dispositivo **eth1** (dedicado a la red privada) cuya dirección de origen no es la de la red privada (! -s 192.168.0.0) se niega. Además, revise si cualquier paquete procedente del exterior está designando la red privada como su origen. En este ejemplo, se rechaza cualquier paquete con dirección de origen de la red privada en cualquier dispositivo Ethernet que no sea el de la red privada (**eth1**). La misma estrategia aplica para el host local.

```
# Engaño de IP, se rechaza cualquier paquete de la red interna
# con una dirección de origen externa.
iptables -A INPUT -j LOG -i eth1 \! -s 192.168.0.0/24
iptables -A INPUT -j DROP -i eth1 \! -s 192.168.0.0/24
iptables -A FORWARD -j DROP -i eth1 \! -s 192.168.0.0/24
# Engaño de IP, se rechaza cualquier paquete externo (que no sea de eth1)
# que tenga la dirección de origen de la red interna
iptables -A INPUT -j DROP \! -i eth1 -s 192.168.0.0/24
iptables -A FORWARD -j DROP \! -i eth1 -s 192.168.0.0/24
# Engaño de IP, se rechaza cualquier paquete del exterior con dirección localhost
# (los paquetes no están en la interfaz local (en eth0 o eth1)
# que tengan la dirección de origen localhost)
iptables -A INPUT -j DROP -i \! lo -s 127.0.0.0/255.0.0.0
iptables -A FORWARD -j DROP -i \! lo -s 192.168.0.0/255.0.0.0
```

Después, debe configurar reglas para permitir que pasen todos los paquetes enviados y recibidos en su sistema (host local).

```
iptables -A INPUT -j ACCEPT -i lo
```

Acceso a servidor

En el caso del servidor Web, tal vez quiera permitir el acceso a usuarios externos, pero bloquear el acceso a cualquiera intentado iniciar una conexión desde el servidor Web en la red privada. En el siguiente ejemplo, se aceptan todos los mensajes al servidor Web, pero éste no inicia contacto con la red privada. Esto evita que cualquiera entre en la red local a través del servidor Web, abierto al acceso externo. Las conexiones establecidas se permiten, lo que facilita la red privada use el servidor Web.

```
# permite la comunicación con el servidor web (dirección 10.0.0.2), port www
iptables -A INPUT -j ACCEPT -p tcp -i eth0 --dport www -s 10.0.0.2
# Permite conexiones establecidas de servidores Web con la red interna
iptables -A INPUT -m state --state ESTABLISHED,RELATED -i eth0 \
    -p tcp --sport www -s 10.0.0.2 -d 192.168.0.0/24 -j ACCEPT
# Evita nuevas conexiones de los servidores Web a la red interna
iptables -A OUTPUT -m --state NEW -o eth0 -p tcp \
    --sport www -d 192.168.0.1/24 -j DROP
```

Acceso externo a la firewall

Para abrir el acceso a redes externas en la firewall, debe permitir la entrada de todos los paquetes, excepto los ICMP. Estos se manejan después. La firewall está especificada con el dispositivo de firewall, **eth0**. En primer lugar, su firewall debe permitir conexiones establecidas y relacionadas para seguir adelante, como se muestra aquí. Luego debe bloquear el acceso externo, descrito más adelante.

```
# permite la comunicación externa con la firewall,
# excepto para los paquetes ICMP
iptables -A INPUT -m state --state ESTABLISHED,RELATED \
    -i eth0 -p ! icmp -j ACCEPT
```

Bloqueo del acceso iniciado en el exterior

Para evitar que usuarios del exterior inicien cualquier acceso a su sistema, cree una regla para bloquear el acceso por paquetes SYN del exterior, usando la opción **state** con **NEW**. Rechace cualquier conexión nueva en la conexión **eth0** (suponiendo que sólo **eth0** está conectado a Internet o una red externa).

```
# evita las conexiones iniciadas en el exterior
iptables -A INPUT -m state --state NEW -i eth0 -j DROP
iptables -A FORWARD -m state --state NEW -i eth0 -j DROP
```

Acceso a la red local

Para permitir la interacción de la red interna con la firewall, permita la entrada de todos los paquetes en la conexión Ethernet interna, **eth1**. Las direcciones de red internas se designan como origen de la entrada.

```
iptables -A INPUT -j ACCEPT -p all -i eth1 -s 192.168.0.0/24
```

Enmascaramiento de redes locales

Para implementar el enmascaramiento, donde los sistemas de la red privada utilizan direcciones de Internet en la puerta de enlace para conectarse a host de Internet, cree una regla POSTROUTING de tabla NAT (**-t nat**) con un destino MASQUERADE.

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Control de paquetes ICMP

Además, para ping y paquetes ICMP alcanzando el destino, inserte reglas INPUT con la firewall como destino. Para habilitar operaciones de ping, use los tipos ICMP echo-reply y echo-requests, así como para destinos inalcanzables, utilice el tipo destination-unreachable.

```
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type \
    echo-reply -d 10.0.0.1
```

```
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type \
    echo-request -d 10.0.0.1
iptables -A INPUT -j ACCEPT -p icmp -i eth0 --icmp-type \
    destination-unreachable -d 10.0.0.1
```

Al final, el reenvío de IP se activa de nuevo.

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Listas de reglas

Una lista de opciones **iptables** muestra diferentes reglas para cada opción, ilustradas aquí:

```
# iptables -L
Chain INPUT (policy DROP)
target    prot opt source                destination
LOG       all  --  !192.168.0.0/24         anywhere        LOG level warning
DROP      all  --  !192.168.0.0/24         anywhere
DROP      all  --  192.168.0.0/24          anywhere
DROP      all  --  127.0.0.0/8             anywhere
ACCEPT    all  --  anywhere               anywhere
ACCEPT    tcp  --  10.0.0.2                anywhere        tcp dpt:http
ACCEPT    tcp  --  10.0.0.2                192.168.0.0/24  state RELATED,ESTABLISHED
                                     tcp spt:http
ACCEPT    !icmp --  anywhere               anywhere        state RELATED,ESTABLISHED
DROP      all  --  anywhere               anywhere        state NEW
ACCEPT    all  --  192.168.0.0/24          anywhere
ACCEPT    icmp --  anywhere               10.0.0.1        icmp echo-reply
ACCEPT    icmp --  anywhere               10.0.0.1        icmp echo-request
ACCEPT    icmp --  anywhere               10.0.0.1        icmp destination-unreachable
Chain FORWARD (policy ACCEPT)
target    prot opt source                destination
DROP      all  --  !192.168.0.0/24         anywhere
DROP      all  --  192.168.0.0/24          anywhere
DROP      all  --  127.0.0.0/8             anywhere
DROP      all  --  anywhere               anywhere        state NEW

Chain OUTPUT (policy ACCEPT)
target    prot opt source                destination
DROP      tcp  --  anywhere               192.168.0.0/24  state NEW tcp spt:http

# iptables -t nat -L
Chain PREROUTING (policy ACCEPT)
target    prot opt source                destination
Chain POSTROUTING (policy ACCEPT)
target    prot opt source                destination
MASQUERADE all  --  anywhere               anywhere
Chain OUTPUT (policy ACCEPT)
target    prot opt source                destination
```

Reglas definidas por el usuario

En el caso de reglas más complejas, tal vez quiera crear su propia cadena para reducir la repetición. Un método común consiste en definir una cadena de usuario para las cadenas INPUT y FORWARD, para que no repetir las operaciones de DROP para cada una. En cambio, sólo tiene una cadena de

usuario donde las cadenas FORWARD e INPUT se alimenten para operaciones DROP. Tenga en cuenta que las cadenas FORWARD e INPUT pueden tener reglas separadas además de las que comparten. En el siguiente ejemplo, se crea una cadena definida por el usuario llamada entrante. Esta cadena se define con la opción **-N** en la parte superior de la secuencia de comandos.

```
iptables -N entrante
```

Debe definir una cadena de usuario antes de usarla como destino en otras reglas, de modo que primero debe definir y agregar todas las reglas para esa cadena y después emplearla como destino. Primero se define la cadena de entrada y agregan sus reglas. Después, al final del archivo, se utiliza como destino para las cadenas INPUT y FORWARD. La cadena INPUT muestra una lista de reglas para aceptar paquetes, mientras la cadena FORWARD tiene una directiva ACCEPT que los aceptará como opción predeterminada

```
iptables -N entrante
iptables -F entrante
# Engaño de IP, se rechaza cualquier paquete de la red interna
#que tenga una dirección de origen externa.
iptables -A entrante -j LOG -i eth1 \! -s 192.168.0.0/24
iptables -A entrante -j DROP -i eth1 \! -s 192.168.0.0/24
iptables -A entrante -j DROP \! -i eth1 -s 192.168.0.0/24
.....
# entradas al final de la secuencia de comandos
iptables -A INPUT -j entrante
iptables -A FORWARD -j entrante
```

Aquí se muestra un listado de las reglas correspondientes:

```
Chain INPUT (policy DROP)
target prot opt source destination
entrante all -- 0.0.0.0/0 0.0.0.0/0
Chain FORWARD (policy ACCEPT)
target prot opt source destination
entrante all -- 0.0.0.0/0 0.0.0.0/0
Chain entrante (2 references)
target prot opt source destination
LOG all -- 192.168.0.0/24 0.0.0.0/0 LOG flags 0 level 4
DROP all -- 192.168.0.0/24 0.0.0.0/0
DROP all -- 192.168.0.0/24 0.0.0.0/0
```

En el caso de reglas donde las cadenas pueden diferir, todavía deberá insertar reglas separadas. En la secuencia de comandos **mifiltro**, la cadena FORWARD tiene una directiva ACCEPT, que permite el paso a través de la firewall a todos los paquetes reenviados a la red local. Si la cadena FORWARD tiene una directiva DROP, como la cadena INPUT, tal vez deba definir reglas separadas bajo las que la cadena FORWARD puede aceptar paquetes. En este ejemplo, las cadenas FORWARD e INPUT tienen diferentes reglas para aceptar paquetes en el dispositivo **eth1**. La regla INPUT es más restrictiva. Para habilitar la recepción de la red local a paquetes reenviados a través de la firewall, habilite el reenvío en su dispositivo empleando una regla FORWARD separada, como se muestra aquí:

```
iptables -A FORWARD -j ACCEPT -p all -i eth1
```


La cadena INPUT aceptará paquetes sólo de la red local.

```
iptables -A INPUT -j ACCEPT -p all -i eth1 -s 192.168.0.0/24
```

Configuración de LAN simple

Para crear una secuencia de comandos soportando una red de área local (LAN) simple, sin servicio de Internet como servidores Web, simplemente no incluya reglas soportando esos servicios. Todavía necesitará reglas FORWARD y POSTROUTING para conectar sus hosts locales a Internet, además de reglas para regir la interacción entre los host y la firewall. Para modificar la secuencia de comandos de ejemplo con el fin de soportar una LAN simple sin el servidor Web, sólo elimine las tres reglas gobernando el servidor Web. Deje todo lo demás igual.

Configuración LAN con servicios de Internet en el sistema de la firewall

A menudo, el mismo sistema funcionando como firewall también se usa para ejecutar servidores de Internet, como los servidores FTP y Web. En este caso, las reglas de la firewall aplican a puertos empleados para tales servicios. La secuencia de comandos de ejemplo trata con un servidor Web ejecutado en un sistema host separado. En cambio, si un servidor Web estuviera ejecutándose en un sistema de firewall, aplicaría las reglas de la firewall del servidor Web al puerto usando éste. En general, el puerto utilizado para un servidor Web es 80. En el siguiente ejemplo, las reglas IPTables para el servidor Web han aplicado al puerto www, puerto 80, en el sistema de firewall. La modificación sólo requiere eliminar referencias de direcciones host del servidor Web antiguo, 10.0.0.2.

```
# permite la comunicación con el servidor Web, puerto www (puerto 80)
iptables -A INPUT -j ACCEPT -p tcp -i eth0 --dport www
# Permite establecer conexiones entre los servidores Web y la red interna
iptables -A INPUT -m state --state ESTABLISHED, RELATED -i eth0 \
    -p tcp --sport www -d 192.168. 0.0/24 -j ACCEPT
# Evita nuevas conexiones de servidores Web con la red interna
iptables -A OUTPUT -m state --state NEW -o eth0 -p tcp \
    --sport www -d 192.168.0.1.0/24 -j DROP
```

Entradas similares pueden configurarse para un servidor FTP. En caso de ejecutar varios servicios de Internet, puede utilizar una regla definida por usuario para ejecutar las mismas reglas en cada servicio, en lugar de repetir tres reglas separadas por servicio. Trabajando en la secuencia de comandos de ejemplo, utilizaría dos reglas definidas, una para INPUT y otra para OUTPUT, para controlar paquetes entrantes y salientes para los servicios.

```
iptables -N serviciodeentrada
iptables -N serviciodesalida
iptables -F serviciodeentrada
iptables -F serviciodesalida
# permite comunicación con el servicio
iptables -A serviciodeentrada -j ACCEPT -p tcp -i eth0
# Permite conexiones establecidas del servicio con la red interna
iptables -A serviciodeentrada -m state --state ESTABLISHED, RELATED -i eth0 \
    -p tcp -d 192.168. 0.0/24 -j ACCEPT
# Evita nuevas conexiones de los servidores Web con la red interna
iptables -A serviciodesalida -m state --state NEW -o eth0 -p tcp \
    -d 192.168.0.1.0/24 -j DROP
.....
```

```
# Ejecuta reglas para el servidor Web, puerto www (puerto 80)
iptables -A INPUT --dport www -j serviciodeentrada
iptables -A INPUT --dport www -j serviciodesalida
# Ejecuta reglas para el servidor FTP, puerto ftp (puerto 21)
iptables -A OUTPUT --dport ftp -j serviciodeentrada
iptables -A OUTPUT --dport ftp -j serviciodesalida
```

Enmascaramiento de IP

En sistemas Linux, puede configurar una red en la que sólo tiene una conexión a Internet, usada por varios sistemas de su red. De esta forma, al utilizar sólo una dirección IP, varios sistemas diferentes se conectan a Internet. A este método se le denomina *enmascaramiento de IP*, donde un sistema se enmascara como otro sistema, usando la dirección IP de este último. En ese tipo de red, un sistema se conecta a Internet con su propia dirección IP, mientras otros sistemas se conectan en una LAN a ese sistema. Cuando un sistema local quiere acceder a la red, se enmascara como sistema conectado a Internet, tomando prestada su dirección IP.

El enmascaramiento IP se implementa en Linux usando la herramienta firewalling de IPtables. En efecto, usted configura una firewall, que después ajusta para enmascaramiento de IP. Actualmente, el enmascaramiento de IP soporta todos los servicios de red comunes (como hace firewalling de IPtables): exploración Web, Telnet y ping. Otros servicios, como IRC, FTP y RealAudio, requieren el uso de ciertos módulos. Cualquier servicio al que quiera sistemas locales accedan, también debe estar en el sistema de la firewall, porque los servicios de ese sistema manejan solicitudes y respuestas.

Encontrará más información acerca del enmascaramiento de IP en el sitio Web de IP Masquerade Resource en ipmasq.webhop.net. En particular, el mini HOWTO del enmascaramiento de IP proporciona una guía detallada, paso a paso, para configurar el enmascaramiento IP en su sistema. Es necesario que el kernel soporte esta función antes de usarla. Si su kernel no lo hace, tal vez tenga que reconstruirlo, incluido el soporte a enmascaramiento de IP o usar módulos de carga para agregarlo. Consulte el mini HOWTO del enmascaramiento de IP para conocer más información.

Con el enmascaramiento de IP, tal como se implementa en sistemas Linux, la máquina con la dirección de Internet es también firewall y puerta de enlace, para la LAN de máquinas usando la dirección de Internet de la firewall para conectarse a Internet. A las firewalls que también implementan enmascaramiento de IP, en ocasiones se les conoce como *puertas MASQ*. Con el enmascaramiento de IP, el sistema conectado a Internet (la firewall) escucha peticiones de Internet de hosts en su LAN. Cuando recibe una, reemplaza la dirección IP del host local solicitante con la dirección IP de Internet de la firewall y luego pasa la solicitud a Internet, como si fuera suya. Después, las respuestas desde Internet se envían al sistema de firewall. Las respuestas recibidas por la firewall son dirigidas a la firewall, empleando su dirección de Internet. Entonces la firewall determina el sistema al que están respondiendo sus solicitudes. Después se quita su dirección IP y envía la respuesta al localhost a través de la LAN. La conexión es transparente desde la perspectiva de las máquinas locales. Éstas parecen estar conectadas directamente a Internet.

Enmascaramiento de redes locales

El enmascaramiento de IP a menudo se usa para permitir que las máquinas de una red privada accedan a Internet. Pueden ser máquinas de una red casera o LAN pequeña, como negocios pequeños. Tal vez ese tipo de red sólo tenga una máquina con acceso a Internet y, como tal, sólo una dirección de Internet. La red privada local tendría direcciones IP seleccionadas desde ubicaciones de red privada (10., 172.16 o 192.168). De manera ideal, la firewall tiene dos tarjetas Ethernet: una

para la interfaz a LAN (por ejemplo; **eth1**) y otra para interfaz a Internet, como **eth0** (para ISP por marcado telefónico, sería **ppp0** para el módem). A la tarjeta para conexión a Internet (**eth0**) se le asignaría la dirección IP de Internet. La interfaz Ethernet para la red local (**eth1**, en este ejemplo) es la interfaz Ethernet de la firewall. Su LAN privada tendría una dirección red como 192.168.0. A su interfaz de firewall de Ethernet (**eth1**) se le asignaría la dirección IP 192.168.0.1. En efecto, la interfaz de la firewall permite a la firewall operar como puerta de enlace de la red local. Entonces la firewall se configura para enmascarar cualquier paquete entrante desde la red privada. Su LAN necesita servidor propio de nombres de dominio, identificando las máquinas en red, incluida su firewall. Cada máquina local necesita tener una firewall especificada como su puerta de enlace. No intente usar alias de IP para asignar firewall y direcciones IP de Internet a la misma interfaz física. Use interfaces separadas para ambas, como dos tarjetas Ethernet o una tarjeta Ethernet y un modem (**ppp0**).

Enmascaramiento de reglas NAT

En Netfilter, el enmascaramiento de IP es una operación de NAT y no está integrado con el filtrado de paquetes en IP Chains. Los comandos del enmascaramiento de IP se colocan en una tabla NAT y tratan de manera separada, desde los comandos de filtrado de paquetes. Utilice IPtables para colocar una regla de enmascaramiento en la tabla NAT. Primero haga referencia a la tabla NAT con la opción **-t nat**. Después agregue una regla a la cadena POSTROUTING con la opción **-o**, para especificar dispositivo de salida y opción **-j** con el comando MASQUERADE.

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Reenvío de IP

El siguiente paso es activar el reenvío IP, ya sea de manera manual o configurando la variable **net.ipv4.ip_forward** en el archivo **/etc/sysctl.conf** y ejecutar **sysctl** con la opción **-p**. El reenvío IP estará desactivado, como opción predeterminada. Para IPv6, utilice **net.ipv6.conf.all.forwarding**. Las entradas **/etc/sysctl.conf** se muestran aquí:

```
net.ipv4.ip_forward = 1
net.ipv6.conf.all.forwarding = 1
```

Ejecute entonces **sysctl** con la opción **-p**.

```
sysctl -p
```

Puede cambiar directamente los archivos de reenvío correspondientes con un comando **echo**, como aquí se muestra:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Para IPv6, use el archivo de reenvío en el directorio **/proc/sys/net/ipv6** correspondiente, **/conf/all/forwarding**.

```
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
```

Enmascaramiento de hosts seleccionados

En lugar de enmascarar todos los localhosts como única dirección IP del host de firewall y puerta de enlace, utilice la tabla NAT para reescribir las direcciones de unos cuantos host seleccionados. A menudo, este método aplica a configuraciones donde quiere que varios hosts locales aparezcan

como servidores de Internet. Al utilizar destinos DNAT y SNAT, dirige los paquetes a hosts locales específicos. Emplee las reglas en las cadenas PREROUTING y POSTROUTING para dirigir entrada y salida de paquetes.

Por ejemplo, el servidor Web descrito en el ejemplo anterior podría configurarse como localhost, que el destino DNAT podría redirigir cualquier paquete recibido originalmente por 10.0.0.2. Digamos que el servidor Web se configuró en 192.168.0.5. Aparecería como si tuviera la dirección 10.0.0.2 en Internet. La tabla NAT reescribiría los paquetes enviados a 10.0.0.2 y dirigiría a 192.168.0.5. Usaría la cadena PREROUTING con la opción **-d** para administrar paquetes entrantes y POSTROUTING con la opción **-s** para paquetes salientes.

```
iptables -t nat -A PREROUTING -d 10.0.0.2 \
    --to-destination 192.168.0.5 -j DNAT
iptables -t nat -A POSTROUTING -s 192.168.0.5 \
    --to-source 10.0.0.2 -j SNAT
```

SUGERENCIA Tenga en cuenta que con IPtables, el enmascaramiento no se combina con la cadena FORWARD, como hacía con IP Chains. De modo que, si especifica una directiva DROP para la cadena FORWARD, también tendrá que habilitar específicamente la operación FORWARD, para la red que se está enmascarando. También necesitaría una regla POSTROUTING y otra FORWARD.

VI

PARTE

Internet y servicios de red

CAPÍTULO 21

Administración de servicios

CAPÍTULO 22

Servidor FTP

CAPÍTULO 23

Servidores Web

CAPÍTULO 24

Servidores proxy

CAPÍTULO 25

Servidores de correo

CAPÍTULO 26

Servidores de impresión,
noticias, búsqueda y bases
de datos

Administración de servicios

Un solo sistema de Linux puede proporcionar varios tipos de servicios diferentes, que van desde seguridad hasta administración e incluyen servicios de Internet obvios como sitios Web y FTP, correo electrónico e impresión. Las herramientas de seguridad como SSH y Kerberos se ejecutan en calidad de servicios, junto con herramientas de red administrativas como DHCP y LDAP. La interfaz de conexión de red es por sí sola un servicio que puede reiniciar cuando quiera. Cada servicio opera como daemon, en ejecución continua, buscando peticiones para sus servicios particulares. En el caso de un servidor Web, las peticiones vendrán de usuarios remotos. Tiene por opción activar o desactivar servicios al iniciar o terminar sus daemons.

El proceso de iniciar o terminar un servicio se administra mediante secuencias de comandos de servicios, descritos con más detalle en este capítulo. Aplica a todos los servicios, incluidos los analizados en las partes del libro “Seguridad”, “Administración de sistema” y “Administración de red”. Se cubren en esta sección del libro porque es probable que los utilice para iniciar o detener servicios de Internet, como servidores Web y servidores de correo.

Archivos de sistema: `/etc/rc.d`

Cada vez que inicia su sistema, éste lee una serie de comandos de inicio de archivos de inicialización de sistema ubicados en el directorio `/etc/rc.d`. Estos archivos de inicialización se organizan de acuerdo con diferentes tareas. Algunas localizadas en el propio directorio `/etc/rc.d`, mientras otras se ubican en un subdirectorio llamado `init.d`. No debe cambiar estos archivos. La organización de archivos de inicialización de sistema varía entre distribuciones de Linux. Algunos de los que se encuentran en `/etc/rc.d/` se muestran en la tabla 21-1.

`rc.sysinit` y `rc.local`

El archivo `/etc/rc.d/rc.sysinit` almacena comandos para inicializar su sistema, incluido montaje y desmontaje de sistemas de archivos. El archivo `/etc/rc.d/rc.sysinit` es el último archivo de inicialización que se ejecuta. Coloque sus propios comandos aquí. Cuando apaga su sistema, éste llama al archivo `halt`, conteniendo comandos de apagado. Luego llama a los archivos de `init.d` para apagar daemons y desmontar los sistemas de archivos. `halt` se ubica en el directorio `init.d`.

Archivos y directorios	Descripción
<code>/etc/sysconfig</code>	Directorio que almacena archivos de configuración y directorios.
<code>/etc/rc.d</code>	Directorio que almacena archivos de inicio y apagado de sistema.
<code>/etc/rc.d/rc.sysinit</code>	Archivos de inicialización para su sistema.
<code>/etc/init.d/rc.local</code>	Archivo de inicialización para sus propios comandos; edite de manera libre este archivo al agregar sus propios comandos de inicio; es el último archivo de inicio que se ejecuta.
<code>/etc/init.d</code>	Directorio almacenando secuencias de comandos de red para iniciar conexiones de red.
<code>/etc/rc.d/rcnum.d</code>	Directorio para diferentes niveles de ejecución, donde <i>num</i> es el nivel de ejecución. Los directorios almacenan vínculos para secuencias de comandos en el directorio <code>/etc/init.d</code> .
<code>/etc/init.d</code>	Directorio que almacena secuencias de comandos de servicio de sistema (véase la Tabla 21-2).
<code>/etc/init.d/halt</code>	Operaciones realizadas cada vez que apaga su sistema, como desmontar sistemas de archivos; se le llama rc.halt en otras distribuciones.

TABLA 21-1 Archivos de sistema y directorios

/etc/init.d

El directorio `/etc/init.d` está diseñado principalmente para almacenar secuencias de comandos que inicien y apaguen diferentes daemons especializados, como red e impresora, fuente y servidores Web. Estos archivos realizan una doble tarea: iniciar un daemon cuando el sistema se enciende y cerrar un daemon cuando el sistema se apaga. Los archivos **init.d** están diseñados de tal forma que es muy fácil escribir secuencias de comandos para iniciar y apagar aplicaciones especializadas. Usan funciones definidas en el archivo **functions**. Muchos de estos archivos se configuran automáticamente. No necesita cambiarlos. Si lo hace, primero asegúrese de conocer cómo funcionan.

Cuando su sistema inicia, también varios programas y se ejecutan automáticamente de manera continua para proporcionar servicios, como un servidor de impresora o sitio Web. Dependiendo del tipo de servicios que quiera en su sistema, agregue o elimine el elemento en una lista de servicios que inician automáticamente. Por ejemplo, el servidor Web se ejecuta automáticamente cuando su sistema inicia. Si no es host de un sitio Web, no lo necesita. Evite que el servicio inicie, eliminando una tarea adicional que el sistema no necesita realizar, liberando recursos y posiblemente reduciendo posibles hoyos de seguridad. Varios de los servidores y daemons realizan tareas necesarias. El servidor **sendmail** permite enviar mensajes a través de redes y el servidor **cupsd** realiza operaciones de impresión.

Con el fin de configurar un servicio para iniciar automáticamente con el sistema, utilice las herramientas de configuración de inicio de su distribución. Red Hat y distribuciones similares como SUSE y Fedora, emplean **chkconfig**, mientras Debian y sus distribuciones similares como Ubuntu, recurren a herramientas **rrconf** o **sysv-rc-conf**. El comando **chkconfig** emplea las opciones **on** y **off** para seleccionar y deseleccionar servicios que habrán de iniciarse junto con el sistema (consulte páginas posteriores de este capítulo).

`chkconfig httpd on`

En el caso de distribuciones de Linux permitiendo secuencias de comandos de SysV Init, use el comando **service** para iniciar y detener servicios de manera manual en cualquier momento. Con el comando **service**, se incluyen los argumentos **stop** para detenerlo, **start** para iniciarlo y **restart** para reiniciarlo.

```
service httpd start
```

NOTA En Debian y distribuciones similares, tal vez deba instalar manualmente la herramienta **service**. Los nombres para secuencias de comandos de servicios pueden diferir, como **apache2** en vez de **httpd** para el servidor Web.

SysV Init: secuencias de comandos **init.d**

Los daemons de servidor de inicio y apagado se administran con secuencias de comandos especiales ubicadas en el directorio **/etc/init.d**. Estas secuencias a menudo tienen el mismo nombre que el programa del servicio. Por ejemplo, para el programa de servidor Web **/usr/sbin/httpd**, la secuencia de comandos correspondiente se llama **/etc/init.d/httpd**. Esta secuencia inicia y detiene el servidor Web. A este método para uso de secuencias de comandos de servicios **init.d** de inicio de servidores se llama *SysV Init*, por el método utilizado en el System V de Unix. Algunas de las secuencias de comandos de uso común se muestran en la tabla 21-2.

Las secuencias de comandos del directorio **/etc/init.d** se ejecutan automáticamente cuando inicia su sistema. Sin embargo, tenga cuidado cuando acceda a estas secuencias de comandos. Inician programas esenciales, como su interfaz de red y daemon de impresora. Se accede a estas secuencias de comandos **init** desde vínculos en subdirectorios configurados para cada nivel de ejecución posible. El directorio **/etc/rc.d** almacena un conjunto de subdirectorios cuyos nombres tienen el formato **rcn.d**, donde *n* es un número aludiendo el nivel de ejecución (también existen vínculos en el directorio **/etc**, directamente en los subdirectorios de nivel de ejecución **/etc/rc.d**). Las tareas de nivel de ejecución diferirán de acuerdo con la distribución, aunque la mayor parte de éstas utilizan designaciones de Red Hat o Debian (consulte la tabla 21-3).

La secuencia de comandos **rc** detecta el nivel de ejecución en que se inició el sistema y después ejecuta sólo las secuencias de comandos de servicios especificados en el subdirectorio, para ese nivel de ejecución. Cuando inicia su sistema, **rc** ejecuta las secuencias de comandos de servicio designadas en el directorio de inicio predeterminado, como **rc5.d** (inicio de sesión gráfico para Fedora y SUSE) o **rc1.d** (inicio de sesión gráfico para Debian y Ubuntu). Algunas distribuciones tendrán inicio de sesión de línea de comandos, como **rc3.d** en Fedora, Red Hat y SUSE. Los directorios **rcn.d** almacenan vínculos simbólicos con ciertas secuencias de comandos de servicios, en el directorio **/etc/init.d**. Por tanto, en realidad se llama a la secuencia de comandos del directorio **/etc/init.d** mediante un vínculo simbólico en un directorio **rcn.d**. El vínculo simbólico para la secuencia de comandos **/etc/init.d/httpd**, en el directorio **rc3.d** en Fedora es **S85httpd**. El prefijo S del vínculo viene de “startup” (arranque); por tanto, el vínculo llama la correspondiente secuencia de comandos **init.d**, con la opción **start**. El número indica el orden en que se ejecutan las secuencias de comandos de servicios; los números menores se ejecutan primero. **S85httpd** invoca **/etc/init.d/httpd** con la opción **start**. Si cambia el nombre del vínculo para empezar con *K*, la secuencia de comandos se invoca con la opción **stop**, deteniéndolo. Estos vínculos se usan en directorios de niveles de ejecución 0 y 6, **rc6.d** y **rc0.d**. El nivel de ejecución 0 suspende el sistema y 6 lo reinicia. Utilice el comando **runlevel** para ver en qué nivel de ejecución opera (consulte el capítulo 27, para conocer más detalles acerca de los niveles de ejecución). Una lista de niveles de ejecución se muestra en la tabla 21-3.

Secuencia de comandos de servicio	Descripción
network	Operaciones para iniciar y terminar sus conexiones de red
xinetd	Operaciones para iniciar y terminar el daemon xinetd
autofs	Montaje automático del sistema de archivos (consulte el capítulo 29)
cups	El daemon de impresora CUPS (consulte el capítulo 25)
cpuspeed	Servicio para administrar la velocidad de la CPU (Cool y Quiet de Athlon)
dhcpcd	Daemon de protocolo de configuración de host dinámico (consulte el capítulo 36)
httpd	Servidor Web Apache (apache2 en Debian, capítulo 23)
innd	Servicio de noticias de Internet (consulte el capítulo 26)
ipsec	Servicio VPN seguro IPsec (consulte el capítulo 18)
iptables	Controla el daemon IPtables
ip6tables	IPtables spara el protocolo IP versión 6 (consulte el capítulo 20)
krb5kdc	Servidor kdc de Kerberos (consulte el capítulo 19)
ldap	Servicio LDAP (consulte el capítulo 28)
nfs	Sistema de archivos de red (consulte el capítulo 37)
postfix	Servidor de correo Postfix (consulte el capítulo 25)
sendmail	Daemon MTA de Sendmail (consulte el capítulo 25)
smb	Samba para host Windows (samba en Debian)
squid	Servidor proxy-caché Squid (consulte el capítulo 24)
sshd	Daemon de Secure Shell (consulte el capítulo 19)
syslog	Daemon de inicio de sesión de sistema (consulte el capítulo 27)
vsftpd	Servidor FTP muy seguro (consulte el capítulo 22)
yplibind	Servicio de información de red (NIS, Network Information Service) (consulte el capítulo 37)

TABLA 21-2 Selección de secuencias de comandos de servicios en /etc/init.d

Servicios de inicio: independiente y xinetd

Un *servicio* es un daemon ejecutado junto con otros programas, buscando continuamente una solicitud de sus servicios, ya sea de otros usuarios en su sistema o usuarios remotos, conectados a su sistema mediante una red. Cuando un servidor recibe una solicitud de un usuario, inicia una *sesión* para proporcionar sus servicios. Por ejemplo, si los usuarios quieren descargar un archivo de su sistema, utilizan su propio cliente FTP para conectarse a su servidor FTP e iniciar una sesión. En la sesión, acceden a archivos de su sistema y los descargan. Es necesario que su servidor esté en ejecución para que un usuario acceda a sus servicios. Por ejemplo, si configura un sitio Web en su sistema con archivos HTML, debe tener el programa de servidor Web **httpd** en ejecución, antes de que los usuarios puedan acceder a su sitio Web y desplegar esos archivos.

Nivel de ejecución	Directorio rc.d	Descripción
Red Hat		Red Hat, Fedora, SUSE y distribuciones asociadas
0	rc0.d	Suspende (apaga) el sistema
1	rc1.d	Modo de un solo usuario (sin red, capacidades limitadas)
2	rc2.d	Modo multiusuario, sin soporte NFS (capacidades limitadas)
3	rc3.d	Modo multiusuario (modo operacional completo, sin inicio de sesión gráfico, interfaz de línea de comandos de forma predeterminada)
4	rc4.d	Definido por el usuario
5	rc5.d	Modo multiusuario con inicio de sesión gráfico (modo operacional completo con inicio de sesión gráfico e interfaz gráfica como opción predeterminada)
6	rc6.d	Reinicia el sistema
S	rcS.d	Modo de un solo usuario
Debian		Debian, Ubuntu y distribuciones asociadas
0	rc0.d	Suspende (apaga) el sistema
1	rc1.d	Modo de un solo usuario (capacidades limitadas)
2	rc2.d	Modo multiusuario con inicio de sesión gráfico (modo operacional completo, X server iniciado automáticamente)
3	rc3.d	Definido por el usuario
4	rc4.d	Definido por el usuario
5	rc5.d	Definido por el usuario
6	rc6.d	Reinicia el sistema
S	rcS.d	Modo de un solo usuario

TABLA 21-3 Niveles de ejecución de sistema para distribuciones Red Hat y Debian

Inicio directo de servicios

Cuenta con varias formas de iniciar un servidor. Una es hacerlo manualmente, desde la línea de comandos, al ingresar el nombre del programa de servidor y sus argumentos. Cuando oprime ENTER, el servidor inicia, aunque reaparezca su indicador de línea de comandos. El servidor se ejecuta de manera concurrente, mientras realiza otras tareas. Para ver si su servidor está en ejecución, emplee el comando **service** con la opción **status**.

```
# service httpd status
```

Como opción, utilice el comando **ps** con la opción **-aux**, para mostrar una lista de todos los procesos en ejecución. Debe ver un proceso para el programa de servidor iniciado. Para refinar la lista, use la operación **grep**, con un patrón para el nombre de servidor requerido. El segundo comando muestra una lista de procesos para el servidor Web.

```
# ps -aux
# ps -aux | grep 'httpd'
```

Con la misma facilidad, puede revisar el proceso **httpd** en el Monitor del sistema de GNOME.

Inicio y detención de servicios con secuencias de comandos de servicio

En distribuciones soportando secuencias de comandos de SysV Init, use las de servicio para iniciar o detener su servidor de forma manual. Estas secuencias de comandos se ubican en el directorio `/etc/init.d` y tienen los mismos nombres que los programas de servidor. Por ejemplo, la secuencia de comandos `/etc/init.d/httpd`, con la opción **start**, inicia el servidor Web. Si usa esta secuencia de comandos con la opción **stop**, lo detiene. En lugar de un nombre de ruta completo para la secuencia de comandos, utilice el comando **service** y el nombre de secuencia. Los siguientes comandos son equivalentes:

```
/etc/init.d/httpd stop
service httpd stop
```

Inicio automático de servicios

En lugar de ejecutar manualmente todos los programas de servidor cada vez que inicia su sistema, haga que éste inicie los servidores automáticamente. Esto se hace de dos formas, dependiendo de la manera en que quiera usar un servidor. Puede tener un servidor ejecutándose continuamente desde el momento que inicie su sistema hasta su apagado o hacer que el servidor inicie, sólo cuando recibe una solicitud de servicios por parte de un usuario. Si un servidor se utiliza con frecuencia, tal vez quiera se ejecute todo el tiempo. Si se usa en raras ocasiones, tal vez sólo quiera iniciar cuando recibe una solicitud. Por ejemplo, si es host de un sitio Web, su servidor Web recibe peticiones todo el tiempo de usuarios remotos en Internet. Sin embargo, en el caso de un sitio FTP, tal vez reciba peticiones poco frecuentes, por lo que quizás quiera el servidor FTP inicie sólo cuando recibe peticiones. Por supuesto, ciertos sitios FTP reciben peticiones frecuentes, que necesitan un servidor FTP se ejecute continuamente.

Servidores independientes

A un servidor con inicio automático y en ejecución de manera continua, se le conoce como servidor *independiente*. El procedimiento SysV Init se utiliza para iniciar servidores automáticamente siempre que inicie su sistema. Este procedimiento emplea secuencias de comandos de servicio, para servidores ubicados en el directorio `/etc/init.d`. Casi todos los sistemas Linux configuran el servidor Web para iniciar automáticamente y ejecutarse de manera continua, como opción predeterminada. Una secuencia de comandos para éste, llamada **httpd**, se encuentra en el directorio `/etc/init.d`.

Servidores xinetd

Para que un servidor sólo inicie cuando reciba una solicitud de este servicio, configúrelo con el daemon **xinetd**. Si agrega, cambia o elimina entradas de servidor en archivos `/etc/xinetd`, debe reiniciar el daemon **xinetd** para que estos cambios surtan efecto. En distribuciones soportando secuencias de comandos de SysV Init, reinicie el daemon mediante la secuencia de comandos `/etc/init.d/xinetd`, con el argumento **restart**, como se muestra aquí:

```
# service xinetd restart
```

También se utiliza la secuencia de comandos **xinetd** para iniciar y detener el daemon **xinetd**. Al detenerse, apaga realmente todos los servidores administrando el daemon **xinetd** (que se muestran en la lista del archivo `/etc/xinetd.conf` o el directorio `xinetd.d`)

```
# service xinetd stop
# service xinetd start
```

También puede reiniciar directamente **xinetd** al detener su proceso. Para esto, utilice el comando **killall** con la señal **-HUP** y el nombre **xinetd**.

```
# killall -HUP xinetd
```

Administración de servicios: **chkconfig**, **services-admin**, **rrconf**, **sysv-rc-conf** y **update-rc.d**

Aunque no existe herramienta independiente de distribución para administrar servidores, casi todas las distribuciones usan las herramientas **chkconfig**, **services-admin** (GNOME), **rrconf** (Debian), **sysv-rc-dconf** o **update-rc.d**. La herramienta **chkconfig** fue desarrollada por Red Hat y se utiliza en el mismo, Fedora, SUSE y distribuciones similares, mientras **rrconf** y **update-rc.d** fueron desarrolladas por Debian y se utilizan en Debian, Ubuntu y distribuciones similares. La herramienta **services-admin** es genérica y puede usarse en todas las distribuciones.

Las herramientas proporcionan interfaces simples usadas para seleccionar servidores que quiera se inicien y como quiera que se ejecuten. Utilice estas herramientas para controlar cualquier daemon que quiera iniciar, incluidos servicios de sistema como **cron**, el servidor de impresora, servidores de archivo remotos para Samba y NFS, servidores de autenticación Kerberos y, por supuesto, servidores de Internet para FTP o HTTP. A estos daemons se les conoce como *servicios* y debe considerar estas herramientas como administradores de esos servicios. Cualquiera de estos servicios se configura para iniciar o detenerse en niveles de ejecución diferentes.

Si agrega un nuevo servicio, **chkconfig**, **services-admin**, **rrconf**, **sysv-rc-dconf** o **update-rc.d** lo administrarán. Como se describe en la siguiente sección, los servicios inician en niveles de ejecución específicos, usando vínculos de servicio en varios directorios de nivel de ejecución. Estos vínculos se conectan con secuencias de comandos de servicio en el directorio **init.d**. Los directorios de nivel de ejecución se numeran del 0 al 6, en el directorio **/etc/rc.d**, como **/etc/rc.d/rc3.d** para el nivel de ejecución 3 y **/etc/rc.d/rc5.d** para el 5. Al eliminar un servicio del nivel de ejecución, sólo cambia sus vínculos en el directorio **rc.d** del nivel de ejecución correspondiente. No toca la secuencia de comandos de servicio en el directorio **init.d**.

chkconfig

Tiene la opción de especificar el servicio que quiere iniciar y el nivel en que quiere inicie con el comando **chkconfig**. A diferencia de otras herramientas de administración de servicio, **chkconfig** funciona bien en servicios independientes y **xinetd**. Aunque los servicios independientes se utilizan en cualquier nivel de ejecución, también puede activar o desactivar servicios **xinetd** para niveles de ejecución en que se ejecuta **xinetd**. En la tabla 21-4 se muestra una lista con diferentes opciones de **chkconfig**.

Lista de servicios con **chkconfig**

Para ver una lista de servicios, utilice la opción **--list**. Aquí se muestra un ejemplo de servicios administrados por **chkconfig**. El estado on u off del servicio, se muestra en cada nivel de ejecución, además de servicios **xinetd** y sus estados:

```
chkconfig --list
dhcpcd  0:off 1:off 2:off 3:off 4:off 5:off 6:off
httpd   0:off 1:off 2:off 3:off 4:off 5:off 6:off
named   0:off 1:off 2:off 3:off 4:off 5:off 6:off
lpd     0:off 1:off 2:on  3:on  4:on  5:on  6:off
```

Opción	Descripción
<code>--level nivel de ejecución</code>	Especifica un nivel de ejecución para iniciar, apagar o reiniciar un servicio.
<code>--list servicio</code>	Muestra una lista de información de inicio para servicios en diferentes niveles de ejecución. Los servicios xinetd sólo son on u off . Sin argumentos, todos los servicios se muestran en la lista, incluidos los servicios xinetd .
<code>--add servicio</code>	Agrega un servicio, creando vínculos en los niveles de ejecución específicos de forma predeterminada (o todos los niveles de ejecución, si no están especificados).
<code>--del servicio</code>	Elimina todos los vínculos para el servicio (inicio y apagado) en todos los directorios de nivel de ejecución.
<code>servicio on</code>	Activa un servicio, creando un vínculo de servicio en los directorios de nivel de ejecución especificados o predeterminados.
<code>servicio off</code>	Desactiva un servicio, creando vínculos de apagado en directorios especificados o predeterminados.
<code>servicio reset</code>	Reinicia la información de inicio de servicio, creando vínculos predeterminados de acuerdo con lo especificado en la entrada chkconfig de la secuencia de comandos de servicio init.d del servicio.

TABLA 21-4 Opciones para **chkconfig**

```
nfs      0:off 1:off 2:off 3:off 4:off 5:off 6:off
crond    0:off 1:off 2:off 3:off 4:off 5:off 6:off
xinetd   0:off 1:off 2:on  3:on  4:on  5:on  6:off
xinetd   0:off 1:off 2:off 3:on  4:on  5:on  6:off
xinetd based services:
    time:      off
    finger:    off
    pop3s:     off
    swat:      on
```

Inicio y detención de servicios con **chkconfig**

Utilice la opción **on** para que un servicio inicie en niveles de ejecución específicos y la opción **off** para deshabilitarlos. Especifique el nivel de ejecución para afectar la opción `--level`. Si no especifica un nivel, **chkconfig** usará cualquier información predeterminada de **chkconfig** en una secuencia de comandos de servicio **init.d** del servicio. Las distribuciones suelen instalar servicios con información predeterminada de **chkconfig** ya inserta (si falta, **chkconfig** utiliza los niveles de ejecución 3, 4 y 5). En el siguiente ejemplo se inicia el servidor Web (**httpd**) en un nivel de ejecución 5:

```
chkconfig --level 5 httpd on
```

La opción **off** configura un servicio para apagar, si el sistema entra en un nivel de ejecución especificado. En el siguiente ejemplo se apaga el servidor Web en caso de entrar en el nivel de ejecución 3. Si el servicio no se ejecuta, permanece apagado:

```
chkconfig --level 3 http off
```

La opción **reset** restaura un servicio a las opciones predeterminadas en su **chkconfig**, como especificó en la secuencia de comandos de servicio **init.d** del servicio:

```
chkconfig httpd reset
```

Para ver sólo la información de inicio de un servicio, use únicamente el nombre del servicio con la opción **--list**:

```
chkconfig --list httpd
httpd    0: off  1:off 2:off 3:on  4:off 5:on  6:off
```

Habilitación y deshabilitación de servicios xinetd con chkconfig

El comando **chkconfig** también habilita y deshabilita servicios **xinetd**. Sólo inserte el servicio **xinetd** con la opción **on** u **off**. El servicio iniciará o apagará y editará como corresponde a la línea de desactivación en la secuencia de comandos de configuración **xinetd** del directorio **/etc/xinetd.d**. Por ejemplo, para iniciar SWAT, el servidor de configuración Samba, que se ejecuta en **xinetd**, sólo inserte

```
chkconfig swat on
chkconfig --list swat
swat          on
```

El archivo de configuración **swat** para **xinetd**, **/etc/xinetd.d/swat**, tendrá su línea de desactivación en **no**, como se muestra aquí:

```
disable=no
```

Si quiere apagar el servidor SWAT, utilice la opción **off**. Esto cambiará la línea de desactivación en **/etc/xinetd.d/swat** a “**disable:yes**”.

```
chkconfig swat off
```

El mismo procedimiento funciona con otros servicios **xinetd** como el servidor POP3 y **finger**.

Eliminación y adición de servicios con chkconfig

Si quiere eliminar por completo un servicio del proceso de inicio y apagado en todos los niveles de ejecución, utilice la opción **--del**. Esto elimina todos los vínculos de inicio y apagado en los directorios de nivel de ejecución.

```
chkconfig --del httpd
```

También puede agregar servicios a la administración **chkconfig** con la opción **--add**; **chkconfig** creará vínculos de inicio con el nuevo servicio en directorios de inicio apropiados, **/etc/rc.d/rcn.d**. Si eliminó antes todos los vínculos para un servicio, restáurelos con la opción **add**.

```
chkconfig --add httpd
```

Configuración de servicios xinetd para uso con chkconfig

La información de nivel de ejecución predeterminada debe colocarse en secuencias de comandos de servicio manejadas por **chkconfig**. Edite estas secuencias para cambiar la información predeterminada, si así lo desea. Esta información se inserta como una línea comenzando con un signo **#**, seguido por la palabra clave **chkconfig** y dos puntos. Luego elabore una lista de niveles de ejecución predeterminados en que debe iniciar el servicio, junto con prioridades de inicio y

detención. La siguiente entrada muestra una lista de niveles de ejecución 3 y 5, con una prioridad de 85 y una detención de 15. Consulte la sección “Etiquetas de secuencias de comandos de servicio”, para conocer más información:

```
# chkconfig: 35 85 15
```

Por tanto, cuando un usuario activa el servicio **httpd**, sin opción de nivel específica, **chkconfig** iniciará **httpd** en un nivel de ejecución 3 y 5.

```
chkconfig http on
```

Cómo trabaja chkconfig

La herramienta **chkconfig** funciona creando vínculos de inicio y apagado en directorios de nivel de ejecución apropiados en el directorio **/etc/rc.d/rc5.d**. Por ejemplo, cuando **chkconfig** agrega el servicio **httpd** en el nivel de ejecución 5, crea un vínculo en el directorio **/etc/rc.d/rc5.d** con la secuencia de comandos de servicio **httpd** en el directorio **/etc/init.d**. Cuando se desactiva el servicio Web desde el nivel de ejecución 3, se crea un vínculo de apagado en el directorio **/etc/rc.d/rc3.d** para usar la secuencia de comandos **httpd** en el directorio **/etc/init.d**, con el fin de asegurar que el servicio Web no inicie. En el siguiente ejemplo, el usuario activa el servicio Web (**httpd**) en el nivel de ejecución 3, creando el vínculo de inicio en **rc5.d**, **S85httpd** y después desactiva el servicio Web en el nivel de ejecución 3, creando un vínculo de apagado en **rc3.d**, **K15httpd**.

```
chkconfig --level 5 httpd on
ls /etc/rc.d/rc5.d/*httpd
    /etc/rc.d/rc5.d/S85httpd
chkconfig --level 3 httpd off
ls /etc/rc.d/rc3.d/*httpd
    /etc/rc.d/rc3.d/K15httpd
```

rcconf, services-admin, sys-rc-conf y update-rc.d

En Debian y sistemas similares, si desea activar o desactivar servicios para diferentes niveles de ejecución, como hace **chkconfig**, debe usar **rcconf** o **sysv-rc-conf**. Ambas herramientas se ejecutan desde una ventana terminal en línea de comandos. Ambas proporcionan una interfaz basada en cursor empleando las teclas de flechas y la barra espaciadora para activar o desactivar servicios. La herramienta **rcconf** es más limitada; activa y desactiva servicios para todos los niveles de ejecución predeterminados, mientras **sysv-rc-conf** es más refinada, permitiéndole seleccionar niveles de ejecución específicos.

La herramienta **services-admin** de GNOME es tan limitada como **rcconf**, que sólo activa y desactiva servicios, sin especificar nivel de ejecución. Proporciona una GUI en GNOME, a la que puede accederse desde el menú Sistema | Administración, en la entrada Servicios. Cada servicio tiene una casilla de verificación, mismas que tienen una marca de verificación para iniciar al momento del arranque; las que no tienen la marca, no se iniciarán. Para activar o desactivar un servicio, desplácese hasta su entrada y marque la casilla de verificación después de la entrada, si está vacía. Para desactivar un servicio, marque su casilla de verificación nuevamente.

NOTA *Boot Up Manager (bum) proporciona una interfaz de escritorio simple para activar o desactivar servicios. Sus características son parecidas a **rrconf**.*

La herramienta **sysv-rc-conf** despliega una pantalla basada en cursor, donde puede revisar servicios que se ejecutarán o detendrán y nivel de ejecución. Estos niveles se mostrarán en una lista

de 0 a 6 y S. Utilice las teclas de flecha para colocarlo en la celda de su servicio y nivel de ejecución. Después utilice la barra espaciadora para activar o desactivar un servicio.

La herramienta **update-rc.d** es una herramienta de nivel bajo que instala y elimina vínculos de nivel de ejecución. Suele utilizarse cuando instala paquetes de servicio para crear vínculos de nivel de ejecución predeterminados. Utilícela para configurar sus propios niveles de ejecución para un servicio, pero requiere una comprensión detallada de configuración de los vínculos de niveles de ejecución para servicios.

La herramienta **update-rc.d** no afecta vínculos ya instalados. Sólo trabaja en vínculos que no están presentes en directorios de nivel de ejecución. En este sentido, no puede activar o desactivar un servicio directamente como haría con **sysv-rc-conf** y **chkconfig**. Para desactivar un servicio primero tendría que eliminar todos los vínculos de nivel de ejecución en todos los directorios **rcn.d**, mediante la opción **remove** y después agregar los que quiera, con las opciones **start** o **stop**. Esto complica mucho más la activación y desactivación de servicios con la herramienta **update-rc.d**.

Utilice las opciones **start** y **stop** con el nivel de ejecución, para configurar niveles de ejecución en que inicia o detiene un servicio. Necesitará proporcionar un número de vínculo para ordenar la secuencia en que se ejecutará. Inserte el nivel de ejecución seguido por un punto. Especifique más de uno. Lo siguiente iniciará el servidor Web en un nivel de ejecución 5. El número de orden utilizado para el vínculo es 91. El nombre de vínculo será **S91apache**.

```
update-rc.d apache start 91 5 .
```

El número de detención siempre es 100, menos el número de inicio. Así que para un servicio con número de inicio 91 sería 09.

```
update-rc.d apache stop 09 6 .
```

Las opciones **start** y **stop** se combinan.

```
update-rc.d apache 99 start . stop 09 6 .
```

Una opción **defaults** iniciará y detendrá el servicio en el nivel de ejecución predeterminado. Esta opción se usa para establecer de manera rápida vínculos de inicio y detención para todos los niveles de ejecución. Los vínculos de inicio se establecerán en niveles de ejecución 2, 3, 4 y 5. Las entradas de detención se configuran en los niveles 0, 1 y 6.

```
update-rc.d apache 99 start 2 3 4 5 . stop 09 0 1 6 .
```

Las opciones **multiuser** iniciarán las entradas 2, 3, 4, 5 y las detendrán en 1.

```
update-rc.d apache multiuser
```

Para eliminar un servicio utilice la opción correspondiente. Los vínculos no se eliminarán si la secuencia de comandos de servicio todavía está presente en el directorio **init.d**. Utilice la opción **-f** para forzar la eliminación de vínculos sin quitar la secuencia de comandos de servicio. Lo siguiente elimina todas las entradas de servicios Web de inicio y apagado de todos los niveles de ejecución.

```
update-rc.d -f apache remove
```

Para desactivar un servicio en un nivel de ejecución determinado, tendría que eliminar primero todos los vínculos y después agregar los deseados. Así que para desactivar el servidor Apache, en el nivel de ejecución 3, manteniendo activados los niveles de ejecución 2, 4 y 5 utilizaría los siguientes comandos.

```
update-rc.d -f apache remove
update-rc.d apache 99 start 2 4 5 . stop 09 0 1 3 6 .
```

Tenga en cuenta que la opción **remove** elimina todos los vínculos de detención al igual que los de inicio. Así debe reiniciar los vínculos para 0, 1 y 6.

***SUGERENCIA** En Debian y Ubuntu utilice **file-rc** en vez de **sysv-rc**. La herramienta **file-rc** utiliza un solo archivo de configuración, en lugar de vínculos en directorios de nivel de ejecución separados.*

Secuencias de comandos de servicio: /etc/init.d

Casi todo el software usando paquetes RPM instalará de manera automática cualquier secuencia de comandos de servicio y creará los vínculos necesarios en directorios **rcn.d**, donde *n* es el número de nivel de ejecución. Sin embargo, las secuencias de comandos de servicio se utilizan para cualquier programa que quiera ejecutar cuando inicia su sistema. Para que ese programa inicie automáticamente, cree primero una secuencia de comandos de servicio para éste en el directorio **/etc/init.d** y después cree vínculos simbólicos a esa secuencia de comandos, en los directorios **/etc/rc.d/rc3.d** y **/etc/rc.d/rc5.d**. También debe colocarse un vínculo de apagado (*K*) en el directorio **rc6.d**, utilizado para nivel de ejecución 6 (reinicio).

Funciones de secuencias de comandos de servicio

Una versión simplificada de la secuencia de comandos de servicio **httpd** se muestra en una sección posterior. Verá las diferentes opciones, mostradas en una lista, en el ejemplo **/etc/init.d/httpd**, bajo la instrucción **case: start, stop, status, restart y reload**. Si no proporciona una opción (*), se despliega la sintaxis utilizada de secuencia de comandos. **httpd** ejecuta primero una secuencia para definir funciones empleadas en estas secuencias de comandos de servicio. La función **daemon** con **httpd** realmente ejecuta el programa de servidor **/usr/sbin/httpd**.

```
echo -n "Iniciando httpd: "
daemon httpd
echo
touch /var/lock/subsys/httpd
```

La función **killproc** apaga el daemon. Luego se eliminan archivos de bloqueo e ID de proceso (**httpd.pid**):

```
killproc httpd
echo
rm -f /var/lock/subsys/httpd
rm -f /var/run/httpd.pid
```

Las funciones **daemon**, **killproc** y **status** son secuencias de comandos de shell definidas en la secuencia de comandos **functions**, ubicada en el directorio **inet.d**. La secuencia de comandos

Función de secuencia de comandos Init	Descripción
daemon [+/- niveledeprioridaddeservicio] programa [argumentos] [&]	Inicia un daemon, si no se está ejecutando.
killproc programa [señal]	Envía una señal al programa; como opción predeterminada envía un SIGTERM , y si el proceso no se detiene, envía un SIGKILL . También elimina cualquier archivo PID, si puede.
pidofproc programa	Utilizado por otra función, determina el PID de un programa.
status programa	Despliega información de estado.

TABLA 21-5 Funciones de secuencia de comandos Init

functions se ejecuta al principio de cada secuencia de comandos de servicio para activar estas funciones. Una lista de estas funciones se proporciona en la tabla 21-5.

. /etc/init.d/functions

Etiquetas de secuencia de comandos de servicio

Al principio de la secuencia de comandos de servicio se almacenan etiquetas utilizadas para configurar el servidor. Estas etiquetas, comenzando con un **#** inicial, se usan para proporcionar información de tiempo de ejecución acerca del servicio a su sistema. Las etiquetas se muestran en la tabla 21-6, junto con funciones de servicio. Inserte una etiqueta con un símbolo **#** al principio,

Etiqueta de secuencia de comando Init	Descripción
# chkconfig: listadeniveldeinicio prioridaddeinicio prioridaddefinal	Obligatoria. Especifica los niveles de inicio predeterminados para este servicio, además de prioridades de inicio y fin.
# description [In]: descripción del servicio	Obligatoria. La descripción del servicio, seguida con caracteres \. Utilice un # inicial para cualquier línea agregada. Con la opción <i>In</i> , especifique el lenguaje en que está escrita la descripción.
# autoreload: true	Opcional. Si esta línea existe, el daemon revisa sus archivos de configuración y vuelve a cargarlos automáticamente cuando cambian.
# processname: programa	Opcional, se permiten varias entradas. El nombre del programa o daemon iniciado en la secuencia de comandos.
# config: archivo-configuración	Opcional, se permiten varias entradas. Especifica un archivo de configuración utilizado por el servidor.
# pidfile: archivo-pid	Opcional, se permiten varias entradas. Especifica el archivo PID.
# probe: true	Opcional, se utiliza <i>en lugar</i> de las entradas autoreload , processname , config , y pidfile para probar e iniciar el servicio automáticamente.

TABLA 21-6 Etiquetas de secuencia de comandos de System V Init

nombre de la etiqueta con dos puntos y después los argumentos de la etiqueta. Por ejemplo, la etiqueta **processname** especifica el nombre del programa en ejecución, en este ejemplo **httpd**:

```
# processname: httpd
```

Si su secuencia de comandos inicia más de un daemon, debe tener una entrada **processname** para cada uno. Por ejemplo, el servicio Samba inicia los daemons **smbd** y **nmbd**.

```
# processname: smdb
```

```
# processname: smdb
```

El final de la sección de etiqueta se indica con una línea vacía. Después de esta línea, cualquiera comenzando con un **#** se trata como comentario. La línea **chkconfig** muestra listas de niveles de ejecución predeterminados que el servicio debe iniciar, junto con prioridades de inicio y detención. La siguiente entrada muestra una lista de niveles de ejecución 3, 4 y 5, con una prioridad de inicio 85 y una detención 15:

```
# chkconfig: 345 85 15
```

Para la descripción, debe insertar una explicación corta del servicio, usando el símbolo **** antes de una nueva línea para utilizar más de una línea.

```
# description: Servidor Web Apache
```

Con las etiquetas **config**, se especifican archivos de configuración que el servidor puede utilizar. En el caso del servidor web Apache, existen tres configuraciones:

```
# config: /etc/httpd/conf/access.conf
```

```
# config: /etc/httpd/conf/httpd.conf
```

```
# config: /etc/httpd/conf/srm.conf
```

La entrada **pidfile** indica el archivo donde se almacena el ID de proceso del servidor.

Ejemplo de secuencia de comandos de servicio

Como ejemplo, aquí se muestra una versión simplificada de una secuencia de comandos de servicio de un servidor Web, **/etc.init.d/httpd**. Casi todas las secuencias de comandos son más complicadas, sobre todo cuando determinan cualquier argumento o variable que necesite especificar un servidor cuando inicia. La secuencia de comandos tiene el mismo nombre que el daemon del servidor Web, **httpd**:

```
# !/bin/sh
```

```
#
```

```
# Secuencia de comandos de servicio para el servidor Web Apache
```

```
#
```

```
# chkconfig: 35 85 15
```

```
# description: Apache es un servidor World Wide Web. \
```

```
# Se utiliza para servir archivos HTML y CGI.
```

```
# processname: httpd
```

```
# pidfile: /var/run/httpd.pid
```

```
# config: /etc/httpd/conf/access.conf
```

```
# config: /etc/httpd/conf/httpd.conf
```

```
# config: /etc/httpd/conf/srm.conf
```

```
# Biblioteca de función de fuente
```

```
. /etc/init.d/functions

# Veamos cómo nos llamaron.
case "$1" in
  start)
    echo -n "Iniciando httpd: "
    daemon httpd
    echo
    touch /var/lock/subsys/httpd
    ;;
  stop)
    killproc httpd
    echo
    rm -f /var/lock/subsys/httpd
    rm -f /var/run/httpd.pid
    ;;
  status)
    status httpd
    ;;
  restart)
    $0 stop
    $0 start
    ;;
  reload)
    echo -n "Recargando httpd: "
    killproc httpd -HUP
    echo
    ;;
  *)
    echo "Uso: $0 {start|stop|restart}"
    exit 1
esac
exit
```

Instalación de secuencias de comandos de servicio

Las distribuciones que permiten secuencias de comandos de SysV Init, incluirán una secuencia de comandos de servicio en su paquete de servicio. Por ejemplo, un paquete de servidor de Internet incluye la secuencia de comandos del servicio para ese servidor. Al instalar paquetes RPM se instala la secuencia de comandos en el directorio **/etc/init.d** y se crean sus vínculos apropiados en los directorios de nivel de ejecución, como **/etc/rc.h/rc3.d**. Si, en cambio, decide crear el servidor a partir de sus archivos de código fuente, entonces instale manualmente la secuencia de comandos de servicio. Si no existe secuencia de comandos de servicio, primero haga una copia de la **httpd** (cambie su nombre) y después edite la copia para reemplazar todas las referencias a **httpd** con el nombre del programa daemon del servidor. Luego coloque la copia de la secuencia de comandos en el directorio **/etc/init.d** y cree un vínculo simbólico para éste en el directorio **/etc/rc.d/rc3.d**.

Daemon extendido de servicios de Internet (xinetd)

Si su sistema sólo recibe unas cuantas solicitudes de un servicio específico, no es necesario que el servidor de ese servicio se ejecute todo el tiempo. Sólo lo necesita cuando un usuario remoto accede a este servicio. El daemon extendido de servicios de Internet (**xinetd**) administra servicios de

Internet, invocándolos sólo cuando su sistema recibe la solicitud de sus servicios. **xinetd** revisa continuamente cualquier solicitud hecha por usuarios remotos de un servicio determinado de Internet; cuando recibe una solicitud, inicia el daemon del servidor apropiado.

El programa **xinetd** está diseñado para reemplazar a **inetd**, ofreciendo mejoras de seguridad, soporte a inicio de sesión e incluso notificaciones de usuario. Por ejemplo, con **xinetd** se envían noticias de aviso a usuarios, cuando no puedan acceder al servicio, indicándoles por qué. Las opciones de seguridad de **xinetd** se usan para prevenir ataques de negación de servicio, limitando conexiones simultáneas de host remotos o restringiendo la tasa de conexiones entrantes. **xinetd** también incorpora TCP, proporcionando seguridad TCP sin invocar el daemon **tcpd**. Además, no tiene que hacer que un servicio se muestre en la lista del archivo **/etc/services**. **xinetd** se configura para iniciar cualquier tipo de servidor de propósito especial.

Inicio y detención de servicios xinetd

En Red Hat y distribuciones asociadas, active o desactive servicios **xinetd** particulares con **chkconfig**, como se describió antes. Use las opciones **on** y **off** para habilitar o deshabilitar un servicio; **chkconfig** editará la opción deshabilitar el servicio, al cambiar su valor a “yes” para off y “no” para on. Por ejemplo, para habilitar el servidor SWAT, inserte

```
chkconfig swat on
```

En el caso de distribuciones soportando secuencias de comandos de Sys V Init, inicie, detenga o reinicie **xinetd** al usar su secuencia de comandos de servicio en el directorio **/etc/init.d**, como aquí se muestra:

```
# service xinetd stop
# service xinetd start
# service xinetd restart
```

Configuración de xinetd: xinetd.conf

El archivo **xinetd.conf** contiene configuraciones para su servidor **xinetd**, como inicio de sesión y atributos de seguridad (consulte la Tabla 21-7 más adelante, en este capítulo). Este archivo también contiene entradas de configuración de servidor o pueden estar colocadas en archivos de configuración separados en el directorio **/etc/xinetd.d**. El atributo **includedir** especifica este directorio:

```
includedir /etc/xinetd.d
```

Registro de servicios xinetd

Tiene la opción de agregar otros atributos como registro de información de conexiones y prioridad de servicio (**nice**). En el siguiente ejemplo, el atributo **log_on_success** registra duración (**DURATION**) e ID de usuario (**USERID**) para conexiones a un servicio, **log_on_failure** registra usuarios que fallaron en conectarse, **nice** establece una prioridad de servicio de 10.

```
log_on_success += DURATION USERID
log_on_failure += USERID
nice = 10
```

Los atributos definidos en el bloque de opciones predeterminadas establecen atributos globales, como actividad predeterminada de registro y restricciones de seguridad: **log_type** especifica adónde se enviará la información de registro, como un archivo específico (**FILE**) o registrador de sistema (**SYSLOG**),

log_on_success especifica la información que se registrará cuando las conexiones se crean y **log_on_failure** especifica la información que se registrará cuando fallen.

```
log_type = SYSLOG daemon info
log_on_failure = HOST
log_on_success = PID HOST EXIT
```

Seguridad de red xinetd

En el caso de restricciones de seguridad, use **only_from** para restringir el acceso a ciertos hosts remotos. El atributo **no_access** niega el acceso a los host listados, pero no los demás. Estos controles toman direcciones IP como valores. Haga una lista de direcciones IP individuales, un rango de direcciones IP o red, al usar la dirección de red. El atributo **instances** limita el número de procesos de servicios que pueden estar activos de inmediato para un servicio particular. En los siguientes ejemplos se restringe el acceso a la red local 192.168.1.0 y el host local, niega el acceso desde 192.168.1.15 y utiliza el atributo **instances**, para limitar el número de procesos de servidor de una sola vez a 60.

```
only_from = 192.168.1.0
only_from = 192.168.1.0
no_access = 192.168.1.15
instances = 60
```

El programa **xinetd** también ofrece servicios internos, incluidos **time**, **services**, **servers** y **xadmin**: **services** proporciona una lista de servicios activos y **servers** información acerca de servidores; **xadmin** proporciona soporte administrativo a **xinetd**.

Archivos de configuración de servicio xinetd: directorio /etc/xinetd.d

En vez de tener un archivo **xinetd.conf** grande para todos los servicios, las configuraciones de servicio se dividen en varios archivos de configuración, uno para cada servicio. El directorio se especifica en archivos **xinetd.conf** con una opción **includedir**. En el siguiente ejemplo, el directorio **xinetd.d** almacena archivos de configuración **xinetd** para servicios como SWAT. Este método tiene la ventaja de permitirle agregar servicios con sólo crear un archivo de configuración nuevo para éstos. Para modificar un servicio sólo requiere editar su archivo de configuración, no todo el archivo **xinetd.conf**.

Como ejemplo, aquí se muestra el archivo **swat** en el directorio **xinetd.d**. Observe que está deshabilitado, como opción predeterminada:

```
# default: off
# description: SWAT is the Samba Web Admin Tool.\
# Use swat to configure your Samba server. \
# To use SWAT, connect to port 901 with your \
# favorite web browser.
service swat
{
    port                = 901
    socket_type         = stream
    wait                = no
    only_from           = 127.0.0.1
    user                = root
    server               = /usr/sbin/swat
```



```

    log_on_failure += USERID
    disable        yes
}

```

Configuración de servicios: atributos **xinetd**

Las entradas en un archivo de servicios **xinetd** definen el servidor que se activará cuando lo solicite, junto con cualquier opción y advertencia de seguridad. Una entrada se integra por un bloque de atributos definidos para cualquier característica diferente, como nombre de un programa de servidor, protocolo utilizado y restricciones de seguridad. Cada bloque de un servicio de Internet, como un servidor, va precedido por una palabra clave **service** y el nombre con que quiere identificar el servicio. Un par de corchetes encierra el bloque de atributos. Cada entrada de atributo comienza con un nombre de atributo, seguido por un operador de asignación, como **=**, después el valor o valores asignados. Un bloque especial especificado por la palabra clave **default** contiene atributos predeterminados para servicios. La sintaxis se muestra aquí:

```

service <nombre_servicio>
{
<atributo> <operador_asignación> <valor> <valor> ...
...
}

```

Casi todos los atributos toman un sólo valor, por lo cual, se utiliza el operador de asignación estándar, **=**. Algunos atributos toman una lista de valores. Puede asignar valores con el operador **=**, pero también puede agregar o eliminar elementos desde estas listas con los operadores **+=** y **-=**. Utilice **+=** para agregar valores y **-=** para eliminarlos. A menudo se usan operadores **+=** y **-=** para agregar valores a atributos con un valor inicial asignado en el bloque predeterminado.

Los atributos se muestran en la tabla 21-7. Ciertos atributos se requieren para un servicio. Entre éstos se incluyen **socket_type** y **wait**. Para un servicio de Internet estándar, también necesita proporcionar **user** (ID de usuario para el servicio), **server** (nombre del programa de servidor) y **protocol** (protocolo utilizado por el servicio). Con **server_args**, también haga una lista de cualquier argumento que quiera pasar al programa de servidor (no incluye el nombre de servidor). Si **protocol** no está definido, se utiliza el protocolo predeterminado para el servicio.

Habilitación y deshabilitación de servicios de **xinetd**

Puede activar o desactivar servicios manualmente al editar su archivo de configuración **xinetd**. Los servicios se activan o desactivan con el atributo **disable** en su archivo de configuración. Para habilitar un servicio, asigne **no** al atributo **disable**, como se muestra aquí:

```
disable = no
```

Luego debe reiniciar **xinetd** para iniciar el servicio.

```
# /etc/init.d/xinetd restart
```

Para habilitar la administración mediante **chkconfig**, necesita colocar una entrada de comentario predeterminada y otra de descripción antes de cada segmento de servicio. Donde se usen archivos separados, la entrada se coloca en el encabezado de cada archivo. Muchas distribuciones ya proporcionan estos para los servicios instalados en sus distribuciones, como **tftp** y **SWAT**. Una entrada predeterminada puede ser **on** u **off**. Por ejemplo, aquí se muestra la opción **chkconfig** predeterminada y las entradas de descripción para el servicio **FTP**:

Atributo	Descripción
ids	Identifica un servicio. Como opción predeterminada, el ID de servicio es el mismo que el nombre del servicio.
type	Tipo de servicio: RPC , INTERNAL (proporcionado por xinetd), o xinetd (no se presenta en un archivo de sistema estándar).
flags	Entre las posibles marcas se incluyen REUSE , INTERCEPT , NORETRY , IDONLY , NAMEINARGS (permite el uso de tcpd), NODELAY y DISABLE (disable the service). See the xinetd.conf Man page for more details.
disable	Especifique yes para deshabilitar un servicio.
socket_type	Especifique stream para servicios basados en flujo, dgram para servicios basados en datagrama, raw para un servicio requiriendo acceso directo a IP y seqpacket para transmisión confiable de datagrama secuencial.
protocol	Especifica un protocolo para el servicio. El protocolo debe en etc/protocols . Si este atributo no se define, se utilizará el protocolo predeterminado empleado por el servicio.
wait	Especifica si el servicio es de un solo subproceso o varios (yes o no). Si es yes , el servicio es de un solo subproceso, que significa xinetd iniciará el servidor y dejará de administrar las peticiones para el servicio hasta que el servidor se detenga. Si es no , el servicio es de varios subprocesos y xinetd continuará administrando nuevas peticiones para éste.
user	Especifica el ID de usuario (UID, User ID) para el proceso de servidor. El nombre de usuario debe estar en /etc/passwd .
group	Especifica el GID para el proceso de servidor. El nombre de grupo debe estar en /etc/group .
instances	Especifica el número de procesos del servidor activos de manera simultánea para un servicio.
nice	Especifica la prioridad de servidor.
server	Especifica el programa para ejecutar este servicio.
server_args	Muestra una lista de argumentos pasados al servidor. No incluye nombre de servidor.
only_from	Controla los hosts remotos para los que está disponible un servicio particular. Su valor es una lista de direcciones IP. Sin valor, el servicio se niega a todos los host remotos.
no_access	Controla los hosts remotos para los que no está disponible un servicio particular.
access_times	Especifica los intervalos en que el servidor está disponible. Un intervalo tiene la forma hora:min-hora:min.
log_type	Especifica adónde se envía la salida de un registro de servicio, ya sea una utilidad syslog (SYSLOG) o un archivo (FILE).
log_on_success	Especifica la información que se registra cuando un servidor inicia o se detiene. La información que puede especificar incluye PID (ID de proceso del servidor), HOST (la dirección del host remoto), USERID (el usuario remoto), EXIT (estado de salida y señal de terminación) y DURATION (duración de una sesión de servicio).

TABLA 21-7 Atributos para xinetd

Atributo	Descripción
log_on_failure	Especifica la información registrada cuando un servidor no puede iniciar. La información que puede especificar incluye HOST (la dirección del host remoto), USERID (ID de usuario del usuario remoto), ATTEMPT (registra un intento fallido) y RECORD (registra información del host remoto para permitir el monitoreo de intentos para acceder al servidor).
rpc_version	Especifica la versión RPC de un servicio RCP.
rpc_number	Especifica el número de un servicio UNLISTED de RPC.
env	Define variables de entorno para un servicio.
passenv	La lista de variables de entorno del entorno xinetd que se pasará al servidor.
port	Especifica el puerto de servicio.
redirect	Permite que un servicio TCP se redirija a otro host.
bind	Permite que un servicio se una a una interfaz específica en una máquina.
interface	Sinónimo de bind .
banner	El nombre de un archivo desplegado para un host remoto cuando establezca una conexión con ese servicio.
banner_success	El nombre de un archivo que se despliega en el host remoto cuando obtiene una conexión a ese servicio.
banner_fail	El nombre de archivo que se despliega para en el host remoto cuando niega una conexión a ese servicio.
groups	Permite el acceso a grupos a los que tiene acceso el servicio (yes o no).
enabled	Especifica la lista de nombres de servicio que habrá de habilitarse.
include	Inserta el contenido de un archivo especificado como parte del archivo de configuración.
includedir	Toma un nombre de directorio en forma de includedir /etc/xinetd.d . Cada archivo dentro del directorio se leerá en secuencia como un archivo de configuración xinetd , combinándose para formar la configuración xinetd .

TABLA 21-7 Atributos de xinetd (continuación)

```
# default: off
# description: The tftp server serves files using the trivial file transfer \
#   protocol. The tftp protocol is often used to boot diskless \
#   workstations, download configuration files to network-aware printers, \
#   and to start the installation process for some operating systems.
```

Si quiere activar un servicio desactivado por opción predeterminada, establezca su atributo **disable** en **no** y reinicie **xinetd**. Aquí se muestra la entrada para el servidor FTP TFTP. Un comentario inicial dice que está en off de manera predeterminada, pero después el atributo **disable** lo activa:

```
service tftp
{
    socket_type      = dgram
    protocol        = udp
```

```
wait          = yes
user          = root
server        = /usr/sbin/in.tftpd
server_args   = -s /tftpboot
disable       = yes
per_source    = 11
cps           = 100 2
flags         = IPv4
}
```

NOTA También puede utilizar *xinetd* para implementar el reenvío al puerto SSH, su sistema debe utilizarse para entunelar conexiones entre hosts o servicios.

Envolturas TCP

Las envolturas TCP agregan otro nivel de seguridad a servidores administrados por **xinetd**. En efecto, el servidor se envuelve con un nivel de seguridad intermedio, que monitorea conexiones y controla el acceso. Se monitorea una conexión de servidor hecha a través de **xinetd**, al verificar las identidades de usuarios remotos y revisar para asegurarse que se hagan peticiones válidas. Las conexiones se registran con el daemon **syslogd** (consulte el capítulo 27) y pueden encontrarse en archivos **syslogd** como **/var/log/secures**. Con envolturas TCP, también se restringe el acceso por parte de usuarios remotos a su sistema. Las listas de host se almacenan en archivos **hosts.allow** y **hosts.deny**. Las entradas en estos archivos tienen formato **service:hostname:domain**. El dominio es opcional. Para el servicio, especifique un servicio particular, como FTP o inserte **ALL** para todos los servicios. Como nombre de host, especifique uno particular o use un comodín para relacionar varios hosts. Por ejemplo, **ALL** coincidirá con todos los host. En la tabla 21-8 se muestra una lista de comodines disponibles. En el siguiente ejemplo, la primera entrada permite acceder por parte de todos los hosts al servidor Web, **http**. La segunda entrada permite el acceso a todos los servicios para el host **pangol.train.com**. La tercera y cuarta entrada permiten acceso FTP a **conejo.pista.com** y **gorrion.com**:

```
http:ALL
ALL:pangol.train.com
ftp:conejo.pista.com
ftp:gorrion.com
```

Comodín	Descripción
ALL	Coincide con todos los host o servicios.
LOCAL	Coincide con cualquier host especificado sólo con un nombre de host, sin nombre de dominio. Utilizado para coincidir con todos los host en el dominio local.
UNKNOWN	
KNOWN	Coincide con cualquier usuario o host cuyo nombre o dirección sean desconocidos.
PARANOID	Coincide con cualquier host cuyo nombre de host no coincida con su dirección IP.
EXCEPT	Un operador que permite excepciones para coincidencias. Toma la forma de <i>lista1 EXCEPT lista2</i> , donde se excluyen los host que coinciden con los de la <i>lista1</i> que también coinciden con los de la <i>lista2</i> .

TABLA 21-8 Comodines de envoltura de TCP

El archivo **host.allow** almacena hosts a los que permite acceso. Si quiere permitir el acceso a todos, excepto unos cuantos hosts determinados, especifique **ALL** para un servicio en el archivo **hosts.allow** pero haga una lista de hosts a los que está negando el acceso en el archivo **hosts.deny**. Es más seguro usar direcciones IP en vez de nombres de host, porque estos últimos pueden comprometerse a través de registros DNS mediante ataques de engaño, cuando un atacante pretende ser otro host.

Cuando **xinetd** recibe una solicitud de servicio FTP, una envoltura monitorea la conexión e inicia el programa de servidor **in.ftpd**. Como opción predeterminada, se permiten todas las solicitudes. Para permitir específicamente todas las solicitudes del servicio FTP, inserte lo siguiente en su archivo **/etc/hosts.allow**. La entrada **ALL:ALL** abre su sistema a todos los hosts para todos los servicios:

```
ftp:ALL
```

SUGERENCIA Originalmente, las envolturas TCP eran manejadas con el daemon **tcpd**. Sin embargo, **xinetd** ha integrado desde entonces soporte a envolturas TCP en su propio programa. Invoque de manera explícita el daemon **tcpd** para administrar servicios, si lo desea. Las páginas *Man tcpd* (**man tcpd**) proporcionan información más detallada acerca de **tcpd**.

Servidores FTP

El protocolo de transferencia de archivos (FTP, File Transfer Protocol), está diseñado para transferir archivos grandes en una red desde un sistema a otro. Como casi todas las operaciones de Internet, FTP trabaja en un modelo cliente/servidor. Los programas cliente FTP permiten a los usuarios transferir archivos a un sistema remoto, ejecutando un programa de servidor FTP y desde éste. Cualquier sistema Linux opera como servidor FTP. Sólo debe ejecutarse como software de servidor (un daemon FTP con la configuración apropiada). Las transferencias se hacen entre cuentas de usuario en un sistema cliente y servidor. Un usuario de sistema remoto debe iniciar sesión en una cuenta de servidor y luego transferir archivos sólo a ese directorio de la cuenta y desde éste. Un tipo especial de cuenta de usuario, llamada *ftp*, permite a cualquier usuario iniciar sesión con un nombre de usuario “anonymous”. Esta cuenta tiene su propio conjunto de directorios y archivos considerados públicos, disponibles para cualquiera en la red que quiera descargarlos. Muchos sitios FTP en Internet son servidores FTP que ofrecen cuentas de usuario FTP, con inicios de sesión anónimos. Cualquier sistema Linux puede configurarse para soportar acceso anónimo FTP, convirtiéndolo en un sitio FTP de red. Tales sitios trabajan en intranet o Internet.

Servidores FTP

El *software de servidor FTP* consta de un daemon FTP y archivos de configuración. El *daemon* es un programa que revisa continuamente solicitudes FTP de usuarios remotos. Cuando recibe una solicitud, administra un inicio de sesión, configura la conexión para la cuenta de usuario pedida y ejecuta cualquier comando FTP enviada por el usuario remoto. Para acceso FTP anónimo, el daemon FTP permite iniciar sesión al usuario remoto en la cuenta FTP, empleando **anonymous** o **ftp** como nombre de usuario. Luego, el usuario accede los directorios y archivos configurados para la cuenta FTP. Sin embargo, como medida para mayor seguridad, el daemon cambia el directorio root de esa sesión, para ser el directorio de inicio de FTP. Esto oculta el resto del sistema al usuario remoto. Generalmente, cualquier usuario de un sistema puede desplazarse al directorio abierto para él. Un inicio de sesión de usuario con FTP anónimo, sólo puede ver el directorio de inicio de FTP y sus subdirectorios. El resto del sistema se esconde a ese usuario. Este efecto se logra con la operación **chroot** (analizada más adelante), que literalmente cambia el directorio root del sistema para ese usuario por el directorio de FTP. Como opción predeterminada, el servidor FTP también requiere de un usuario una shell válida. Revise una lista de shells válidas en el archivo */etc/shells*. Casi todos los daemons tienen opciones para desactivar esta característica.

Servidores disponibles

Hay varios servidores FTP disponibles para su uso en sistemas Linux (revise la tabla 22-1). Tres de los servidores más comunes son **vsftpd**, **pureftpd** y **proftpd**. Very Secure FTP Server ofrece un servidor FTP muy seguro y simple. El servidor Pure FTPD es un servidor FTP ligero, rápido y seguro, basado en Troll-FTPd. Documentación y fuentes más recientes están disponibles en **pureftpd.org**.

ProFTPD es un daemon FTP popular, basado en un diseño de servidor Web Apache. Presenta configuración simplificada y soporte a host FTP virtuales. El archivo comprimido de la versión más actualizada, junto con la documentación, están disponibles en el sitio Web de ProFTPD en **proftpd.org**. Otro daemon FTP, NcFTPd, es un producto comercial producido por los mismos programadores responsables del cliente FTP NcFTP. Éste es gratuito para uso académico y ofrece un cobro reducido para redes pequeñas. Revise **ncftpd.org** para conocer más información.

También hay varios servidores FTP basados en seguridad, incluidos SSLFTP y SSH **sftpd**, así como **gssftpd**. SSLFTP usa capas de conexión segura (SSL, Secure Sockets Layer), para cifrar y autenticar transmisiones, además de MD5 digest para revisar la integridad de archivos transmitidos. SSH **sftpd** es un servidor FTP, ahora parte del paquete Open SSH, con cifrado y autenticación SSH para establecer conexiones FTP seguras. El servidor **gssftpd** es parte del paquete Kerberos 5 y provee seguridad a nivel Kerberos para operaciones FTP.

Usuarios FTP

Usuarios normales con cuentas en un servidor FTP obtienen acceso FTP completo con sólo iniciar sesión en sus cuentas. Tales usuarios pueden acceder o transferir archivos directamente desde sus propias cuentas o cualquier directorio al que puedan entrar. También pueden crear usuarios, conocidos como invitados, con ingreso restringido a directorios FTP accesibles al público. Esto requiere la configuración de restricciones de usuario estándar, con el directorio público FTP como directorio de inicio. Los usuarios también pueden iniciar sesión como anónimos, permitiendo que cualquier usuario de la red o Internet tenga entrada a los archivos en un servidor FTP.

Servidor FTP	Siio
Very Secure FTP Server (vsftpd)	vsftpd.beasts.org
ProFTPD	proftpd.org
PureFTP	pureftpd.org
NcFTPd	ncftpd.org
SSH sftp	openssh.org
Washington University web server (WU-FTPD)	wu-ftp.org
Tux	Web server con Capacidad FTP
gssftpd	Servidor FTP Kerberos

TABLA 22-1 Servidores FTP

FTP anónimo: vsftpd

Un sitio FTP anónimo es, en esencia, un tipo especial de usuario en su sistema con directorios y archivos, accesibles al público en su directorio de inicio. Cualquier usuario puede iniciar sesión en esta cuenta y llegar a sus archivos. Debido a esto, debe cuidar las restricciones a un usuario FTP remoto, para que sólo acceda a archivos en el directorio FTP anónimo. Frecuentemente, los archivos del usuario se interconectan con toda la estructura de archivos de su sistema. Los usuarios normales tienen acceso de escritura, permitiéndoles crear o eliminar archivos y directorios. Los archivos y directorios FTP anónimos se configuran de tal manera que el resto del sistema de archivos se oculta y a los usuarios remotos sólo se les permite lectura. En ProFTPD, esto se logra mediante directivas de configuración, alojadas en su archivo de configuración. Un método más antiguo implementado por el paquete vsftpd, incluye la copia de ciertos archivos de configuración, comandos y bibliotecas del sistema, colocados en subdirectorios del directorio de inicio de FTP.

Un sitio FTP se compone por una cuenta de usuario FTP, directorio de inicio de FTP y copias de directorios de sistema, conteniendo la configuración elegida y archivos de soporte. Los daemons FTP más recientes, como ProFTPD, no necesitan directorios de sistema ni archivos de soporte. Casi todas las distribuciones ya configuran una cuenta de usuario FTP cuando se instala su sistema.

La cuenta de usuario FTP: anónimo

Para permitir el acceso FTP anónimo a otros usuarios de su sistema, debe tener una cuenta de usuario llamada *FTP*. Casi todas las distribuciones crean esta cuenta. Si su sistema no tiene esta cuenta, deberá crear una. Luego puede colocar restricciones en la cuenta FTP, para evitar que cualquier usuario FTP acceda a cualquier otra parte de su sistema. También debe modificar la entrada de esta cuenta en su archivo */etc/passwd*, para evitar el acceso de usuarios normales. A continuación se muestra la entrada que encontrará en su archivo */etc/passwd*, para configurar un inicio de sesión FTP como usuario anónimo:

```
ftp:x:14:50:FTP User:/var/ftp:
```

La **x** en el campo de contraseña bloquea la cuenta, que evita el acceso de cualquier usuario a ésta, con lo que obtendría control sobre sus archivos o entraría a otras partes de su sistema. El ID de usuario, 14, es un ID único. El campo de comentario es FTP User. El directorio de inicio de sesión es */var/ftp*. Cuando los usuarios FTP inician sesión en su sistema, se colocan en este directorio. Si no ha configurado un directorio de inicio, cree uno y después cambie su propiedad al usuario FTP con el comando **chown**.

Grupo FTP

El ID de grupo es el ID del grupo **ftp**, configurado sólo para usuarios FTP anónimos. Tiene la opción de configurar restricciones al grupo **ftp**, con lo que restringe cualquier usuario FTP anónimo. Aquí se muestra la entrada para el grupo **ftp** en el archivo */etc/group*. Si su sistema no tiene uno, debe agregarlo:

```
ftp::50:
```


Creación de nuevos usuarios FTP

Si está creando un host FTP virtual, necesitará generar un usuario FTP para cada uno, junto con sus directorios. Por ejemplo, para crear un servidor FTP para un host `host1-ftp`, cree un usuario `host1-ftp` con directorio propio.

```
useradd -d /var/host1-ftp host1-ftp
```

Esto crea un usuario como el descrito aquí:

```
host1-ftp:x:14:50:FTP User:/var/host1-ftp:
```

También necesita crear el directorio de inicio correspondiente, `/var/host1-ftp` en este ejemplo y configurar sus permisos para darle a los usuarios acceso restringido.

```
mkdir /var/host1-ftp
chmod 755 /var/host1-ftp
```

Además, necesita asegurarse de que el usuario `root` es dueño del directorio, no los nuevos usuarios FTP. Esto da control del directorio al usuario `root`, no a cualquier usuario iniciando sesión.

```
chown root.root /var/host1-ftp
```

Directorios de servidor FTP anónimos

Como ya se indicó, el directorio de inicio de FTP se llama `ftp` y se coloca en el directorio `/var`. Cuando los usuarios inician sesión de forma anónima, se colocan en este directorio. Parte importante de la protección de su sistema, consiste en evitar que usuarios remotos ejecuten cualquier comando o programa que no figure en los directorios restringidos. Por ejemplo, no dejaría que un usuario utilice su comando `ls` para mostrar una lista de nombres de archivo, porque `ls` se ubica en su directorio `/bin`. Al mismo tiempo, quiere permitir que el usuario FTP despliegue una lista de nombres con el comando `ls`. Los daemons FTP más recientes, como `vsftp` y `ProFTPD`, resuelven este problema creando acceso seguro a comandos y archivos del sistema necesarios, mientras restringen a los usuarios remotos para sólo ver los directorios del sitio FTP. En cualquier caso, asegúrese de que el directorio de inicio de FTP pertenece al usuario `root`, no al usuario FTP. Utilice el comando `ls -ld` para revisar a quién pertenece el directorio FTP.

```
ls -ld /var/ftp
```

Para cambiar la propiedad del directorio, utilice el comando `chown`, como se muestra en este ejemplo.

```
chown root.root /var/ftp
```

Otra solución más tradicional consiste en crear copias de ciertos directorios y archivos del sistema, necesarios para los usuarios remotos y colocarlos en el directorio `ftp`, donde los usuarios pueden alcanzarlos. Un directorio `bin` se coloca en el directorio `ftp` y los usuarios remotos se limitan a éste, en vez del directorio `bin` del sistema. Siempre que usan el comando `ls`, los usuarios remotos están utilizando el de `ftp/bin`, no el que se encuentra en `/bin`. Si, por alguna razón, configura los directorios FTP anónimos, debe utilizar el comando `chmod` para cambiar los permisos de acceso a directorios, para que los usuarios remotos no accedan al resto de su sistema. Cree un directorio `ftp` y utilice el comando `chmod` con el permiso 555 para desactivar el acceso de escritura: `chmod 555 ftp`.

Después, cree un nuevo directorio **bin** en el directorio **ftp**, haga una copia del comando **ls** y colóquelo en **/ftp/bin**. Haga esto para cualquier comando, si quiere que esté disponible para todos los usuarios FTP. Después cree un directorio **ftp/etc** para almacenar una copia de sus archivos **passwd** y **group**. De nuevo, la idea es evitar cualquier acceso a los archivos originales en el directorio **/etc** por parte de los usuarios FTP. Debe editarse el archivo **ftp/etc/passwd** para eliminar cualquier entrada de usuarios regulares en su sistema. Todas las demás entradas deben tener contraseña configurada en **x** para bloquear el acceso. En el caso del archivo **group**, elimine todos los grupos de usuario y asigne **x** a todas las contraseñas. Cree un directorio **ftp/lib** y después haga copias de las bibliotecas que necesite para ejecutar comandos colocados en el directorio **bin**.

Archivos FTP anónimos

Un directorio llamado **pub**, localizado en el directorio de inicio de FTP, suele almacenar archivos disponibles para descarga de usuarios FTP remotos. Cuando los usuarios FTP inician sesión, se colocan en el directorio de inicio de FTP (**/var/ftp**) y después pueden cambiar el directorio **pub** para acceder a esos archivos (**/var/ftp/pub**). En el directorio **pub**, agregue los directorios y archivos que desee. Incluso puede designar directorios para carga, permitiendo a los usuarios FTP transferir archivos a su sistema.

En cada subdirectorio configurado bajo el directorio **pub** para almacenar archivos FTP, debe crear un archivo **README** y otro **INDEX**, como cortesía para los usuarios FTP. El archivo **README** contiene una descripción breve del tipo de archivos almacenados en ese directorio. El archivo **INDEX** contiene una lista de archivos, así como una descripción de lo que almacena cada uno.

Uso de FTP con rsync

Muchos servidores FTP también permiten operaciones rsync, empleando **rsync** como daemon. Esto provee actualizaciones de archivos incrementales e inteligentes desde un servidor FTP. Puede actualizar varios archivos en un directorio o sólo un archivo, como una imagen ISO grande.

Acceso a sitios FTP con rsync

Para acceder al servidor FTP que se ejecuta en un servidor rsync, inserte el comando **rsync** y después el nombre de host; escriba un par de dos puntos y después la ruta del directorio al que quiere acceder o uno de los módulos del servidor FTP. En el siguiente ejemplo, el usuario actualiza un directorio **miproyecto** local, desde el sitio FTP **mipista.com**:

```
rsync ftp.mipista.com::/var/ftp/pub/miproyecto /home/miproyecto
```

Para conocer los directorios a los que rsync da soporte, revise sus módulos rsync en ese sitio. Éstos se definen en el archivo de configuración **/etc/rsyncd.conf** del sitio. Un *módulo* es sólo un directorio con todos sus subdirectorios. Para encontrar módulos disponibles, inserte el sitio FTP con un par de dos puntos.

```
rsync ftp.mipista.com::
ftp
```

Esto indica que el sitio **ftp.mipista.com** tiene un módulo FTP. Para mostrar una lista de archivos y directorios en ese módulo, utilice el comando **rsync** con la opción **-r**.

```
rsync -r ftp.mipista.com::ftp
```

Muchos sitios ejecutando el servidor rsync tendrán un protocolo rsync, que ya configurado para acceder al módulo rsync disponible (directorio). Por ejemplo, para el siguiente URL se usa **rsync** para acceder a la ubicación ibiblio para su distribución. En el caso de Fedora, el módulo se llama fedora-linux-core, que sigue después del nombre de host.

```
rsync://distro.ibiblio.org/fedora-linux-core/
```

Incluso puede emplear **rsync** para actualizar un sólo archivo, como una imagen ISO, que tal vez se haya modificado. En el siguiente ejemplo, se actualiza la imagen ISO del disco 1 de Fedora 7. La opción **--progress** mostrará el progreso de la descarga.

```
rsync -a --progress rsync://distro.ibiblio.org/fedora-linux-core/7/i386/iso/FC7-i386-disc1.iso
```

Configuración de un servidor rsync

Con el fin de configurar un servidor FTP para permitir que los clientes utilicen rsync en su sitio, primero necesita ejecutar rsync como servidor. Utilice **chkconfig** o **sysv-rc-conf** para activar el daemon rsync, comúnmente conocido como **rsyncd**. Esto ejecutará el daemon **rsync** mediante **xinetd**, al utilizar una secuencia de comandos **rsync** en **/etc/xinetd.d**, para activar y configurar parámetros.

```
chkconfig rsync on
```

Cuando ejecute como daemon, rsync leerá el archivo **/etc/rsyncd.conf** para sus opciones de configuración. Aquí tiene la opción de especificar los parámetros de FTP, como ubicación de archivos del sitio FTP. El archivo de configuración se segmenta en módulos, cada uno con opciones propias. Un módulo es una representación simbólica de un árbol exportado (un directorio y sus subdirectorios). El nombre del módulo se encierra entre llaves, como **[ftp]**, para un módulo FTP. Luego inserte opciones para ese módulo, como al usar la opción de ruta, para especificar la ubicación de directorios y archivos de su sitio FTP. Los ID de usuario y grupo se especifican con las opciones **uid** y **gid**. La opción predeterminada es nobody. Aquí se muestra un módulo FTP, ejemplificando el acceso anónimo:

```
[ftp]
    path = /var/ftp/pub
    comment = ftp site
```

En el caso de un acceso más restringido, agregue una opción **auth users**, para especificar usuarios autorizados; como opción predeterminada, rsync permitirá acceso anónimo a todos los usuarios. Es posible incrementar el control del acceso de rsync a áreas en el sitio FTP, mediante un archivo secrets, como **/etc/rsyncd.secrets**. Se trata de una lista separada por dos puntos, con nombres de usuario y contraseñas.

```
alicia:micontra3
larisa:sucont5
```

Un módulo correspondiente al área controlada se verá así:

```
[specialftp]
    path = /var/ftp/pub
    comment = ftp site
    auth users = alicia
    secrets file = /etc/rsyncd.secrets
```

Si se encuentra en su servidor FTP y quiere ver que módulos estarán disponibles, ejecute **rsync** con la opción **localhost** y nada tras los dos puntos.

```
$ rsync localhost::
ftp
specialftp
```

Los usuarios remotos pueden saber que módulos tiene usted al escribir su nombre de host y sólo un par de dos puntos.

```
rsync ftp.mipista.com::
```

Imagen de espejo de rsync

Algunos sitios permitirán utilizar rsync para realizar operaciones de espejo. Con rsync no tiene copia de todo el sitio, sino sólo de archivos que se han modificado. En el siguiente ejemplo se crea una imagen de espejo del sitio FTP mipista en el directorio **/var/ftp/mirror/mipista** en un sistema local:

```
rsync -a --delete ftp.mipista.com::ftp /var/ftp/mirror/mipista
```

La opción **-a** es el modo de archivero, que incluye otras opciones, como **-r** (repetitiva) para incluir todos los subdirectorios, **-t** para preservar fechas y horas de archivos, **-l** para recrear vínculos simbólicos y **-p** para preservar todos los permisos. Además, la opción **--delete** se agrega a archivos eliminados que no existen en el extremo del emisor, eliminando archivos obsoletos.

El servidor FTP muy seguro

Very Secure FTP Server (vsftpd) es pequeño, rápido, sencillo y seguro. Está diseñado para evitar la sobrecarga de aplicaciones grandes en el servidor FTP, como ProFTPD, conservando un nivel de seguridad muy alto. También administra gran cantidad de trabajo de carga, al administrar niveles de tráfico altos en un sitio FTP. Tal vez se trate de lo mejor para sitios donde muchos usuarios anónimos e invitados descargarán los mismos archivos. Reemplazó al servidor FTP de la Universidad de Washington, WU-FTPD, en muchas distribuciones.

Very Secure FTP Server tiene un diseño intrínseco, para suministrar toda la seguridad posible, aprovechando características de los sistemas operativos Unix y Linux. El servidor se separa en procesos privilegiados y no privilegiados. Los procesos no privilegiados reciben todas las solicitudes FTP, interpretándolas y después enviándolas a través de una conexión al proceso privilegiado, que luego filtra todas las solicitudes de forma segura. Ni siquiera los procesos privilegiados se ejecutan con las capacidades completas de root, al utilizar sólo los necesarios para realizar sus tareas. Además, Very Secure FTP Server utiliza su propia versión de comandos de directorio como **ls**, en vez de las versiones del sistema.

Ejecución de vsftpd

El daemon Very Secure FTP Server es denominado **vsftpd**. Se diseñó para ejecutarse como servidor independiente, que se inicia y detiene usando la secuencia de comandos de servidor **/etc/rc.d/init.d/vsftpd**. Para que el servidor inicie automáticamente, actívelo con el comando **chconfig** y el argumento **on**; los servicios **admin**, **rrconf** o **sysv-rc-conf** de GNOME para activar o desactivar el servicio vsftpd. Desactive el servicio para deshabilitar el servidor. Si permitió antes otro servidor FTP como ProFTPD, asegúrese de deshabilitarlo primero.

Como opción, implemente **vsftpd** para ejecutar **xinetd**, al ejecutar el servidor, sólo cuando un usuario hace una solicitud. El daemon **xinetd** ejecutará un archivo de secuencia de comandos de **xinetd** llamado **vsftpd**, ubicado en el directorio **/etc/xinetd.d**. Al principio, el servidor se desactivará.

Configuración de vsftpd

Configure **vsftpd** con un archivo de configuración, **vsftpd.conf**. Las opciones de configuración son simples y mantienen bajo un mínimo, haciéndolo menos flexible que ProFTPd, pero mucho más rápido (consulte la tabla 22-2). El archivo **vsftpd.conf** contiene un conjunto de directivas en que se asigna un valor a una opción (no existen espacios alrededor del signo =). Las opciones pueden ser las marcas on y off, asignadas a un valor **YES** o **NO**, características que toman valor numérico o algunas a las que se asigna una cadena. Un archivo **vsftpd.conf** predeterminado se instala en el directorio **/etc** o **/etc reference/etc/vsftpd**. Este archivo muestra una lista de opciones disponibles de uso común, con explicaciones detalladas para cada uno. Las que no se usan se convierten en comentarios con un caracter **#** antes. Los nombres de opción son muy comprensibles. Por ejemplo, **anon_upload_enable** permite a usuarios anónimos cargar archivos, mientras **anon_mkdir_write_enable** permite a los usuarios anónimos crear directorios. La página Man de **vsftpd.conf** muestra una lista de opciones, facilitando una explicación detallada de cada una.

Habilitación de acceso independiente

Para ejecutar **vsftpd** como servidor independiente, asigne **YES** a la opción **listen**. Esto instruye a **vsftpd** para escuchar continuamente solicitudes en su puerto asignado. Tiene la opción de especificar el puerto en que se escucha con la opción **listen_port**.

```
listen=YES
```

Habilitación de acceso de inicio de sesión

En el siguiente ejemplo, tomado del archivo **vsftpd.conf**, el FTP anónimo está habilitado al asignar el valor **YES** a la opción **anonymous_enable**. La opción **local_enable** permite a usuarios locales en su sistema utilizar el servidor FTP.

```
# Allow anonymous FTP?
anonymous_enable=YES
#
# Uncomment this to allow local users to log in.
local_enable=YES
```

Si quiere que los usuarios anónimos inicien sesión sin proporcionar una contraseña, asigne **YES** a **no_anon_password**.

Permisos de usuario local

Varios permisos de usuario controlan la manera en que los usuarios locales acceden archivos en el servidor. Si quiere permitir que los usuarios locales creen, cambien el nombre, además de eliminar archivos y directorios en su cuenta, debe permitir el acceso de escritura con la opción **write_enable**. De esta forma, también puede borrar cualquier archivo que carguen. Literalmente, la opción **write_enable** activa un rango de comandos para cambiar el sistema de archivos, incluidos crear, cambiar nombre, además de eliminar archivos y directorios. Con **user_config_dir** puede configurar usuarios específicos.

```
write_enable=YES
```

Opción	Descripción
listen	Establece el modo independiente.
listen_port	Especifica el puerto para el modo independiente.
anonymous_enable	Habilita el acceso de usuario anónimo.
local_enable	Habilita el acceso por parte de usuarios locales.
no_anon_password	Especifica si los usuarios anónimos deben presentar una contraseña.
anon_upload_enable	Habilita la carga de archivos para usuarios anónimos.
anon_mkdir_write_enable	Permite a los usuarios anónimos crear directorios.
anon_world_readable_only	Establece el tiempo límite de espera en segundos para sesiones inactivas.
idle_session_timeout	Establece el tiempo límite de espera en segundos para conexiones fallidas.
data_connection_timeouts	Set time limit in seconds for failed connections.
dirmessage_enable	Despliega mensajes de directorio.
ftpd_banner	Despliega un mensaje de inicio de sesión FTP.
xferlog_enable	Habilita el registro de transacciones de transmisión.
xferlog_file	Especifica el archivo de registro.
deny_email_enable	Habilita la negación de usuarios anónimos, cuyas direcciones de correo se especifican en vsftpd.banned .
userlist_enable	Niega el acceso a usuarios especificados en el archivo vsftp.user_list .
userlist_file	Niega o permite el acceso a usuarios, dependiendo de la configuración de userlist_deny .
userlist_deny	Cuando se establece en YES , userlist_file niega el acceso a usuarios en la lista. Cuando se establece en NO , userlist_file permite el acceso sólo a los usuarios en la lista.
chroot_list_enable	Restringe a los usuarios a sus directorios de inicio.
chroot_list_file	Permite a los usuarios acceder a directorios de inicio. A menos que chroot_local_user se establezca en YES , este archivo contiene una lista de usuarios a quienes no se permite acceso a sus directorios de inicio.
chroot_local_user	Permite el acceso a todos los usuarios a sus directorios de inicio.
user_config_dir	Directorio para la capacidad de configuración específica de usuario.
ls_recurse_enable	Permite el despliegue recursivo de listas.

TABLA 22-2 Opciones de configuración para vsftpd.conf

Tiene la opción de especificar con más detalle los permisos para archivos subidos, con la opción **local_umask** (022 es el parámetro predeterminado en **vsftpd.conf**, permitiendo al propietario lectura y escritura, pero sólo lectura a todos los demás usuarios, 644).

`local_umask=022`

Debido a que la carga de archivos ASCII conlleva riesgos de seguridad, se desactivan como opción predeterminada. Sin embargo, si está cargando archivos de texto grandes, tal vez quiera habilitarlos en casos especiales. Utilice **ascii_upload_enable** para permitir la carga de archivos ASCII.

Permisos de usuario anónimo

También puede permitir que usuarios anónimos carguen y eliminen archivos, además de crear y eliminar directorios. La carga de archivos por parte de usuarios anónimos se habilita con la opción **anon_upload_enable**. Para permitir que usuarios anónimos también cambien el nombre de archivo o los eliminen, configure la opción **anon_other_write_enable**.

```
anon_upload_enable=YES
anon_other_write_enable=YES
anon_mkdir_write_enable=YES
```

La opción **anon_world_readable_only** hará que los archivos cargados sean sólo de lectura (para descargar), restringiendo el acceso de escritura al usuario que los creó. Sólo el usuario que los cargó puede eliminarlos.

Todos los archivos cargados pertenecen al usuario FTP anónimo. Puede tener archivos pertenecientes a otro usuario, aumentando la posible seguridad. En efecto, el usuario real, es dueño de los archivos cargados, se oculta para usuarios anónimos. Para habilitar esta opción, use **chown_uploads** y especifique el nuevo usuario con **chown_username**. Nunca haga que el usuario sea administrativo, como **root**.

```
chown_uploads=YES
chown_username=misarchivosftp
```

Se deben otorgar permisos de escritura al propio directorio de carga, por parte de otros usuarios.

```
chmod 777 /var/ftp/upload
```

Puede controlar el tipo de acceso que los usuarios tienen para los archivos con la opción **anon_mask**, configurando los permisos de escritura y lectura predeterminados para subir archivos. La opción predeterminada es 077, otorgando permisos de escritura y lectura sólo al dueño (600). Para permitir el acceso de lectura a todos los usuarios, asigne 022 a umask, donde 2 desactiva el permiso de escritura, pero establece el de lectura (644). El valor 000 permite lectura y escritura a todos los usuarios.

Límites de tiempo de conexión

Para adquirir un control más eficiente de la carga de trabajo en un servidor, establezca límites de tiempos a usuarios inactivos y transmisiones fallidas. La opción **idle_session_timeout** desconectará a los usuarios inactivos tras un tiempo específico, mientras **data_connection_timeouts** desconectará todas las conexiones de datos fallidas. Aquí se muestran las opciones predeterminadas:

```
idle_session_timeout=600
data_connection_timeout=120
```

Mensajes

La opción **dirmessage_enable** permite que un mensaje se almacene en el archivo **.message** del directorio, que habrá de desplegarse cuando un usuario acceda a ese directorio. La opción **ftpd_**

banner permite configurar sus propios mensajes de inicio de sesión FTP. Aquí se muestra la opción predeterminada:

```
ftpd_banner=Bienvenido al servicio FTP bla.
```

Registro

Un conjunto de opciones **xferlog** controla los registros. Puede habilitar los registros, al igual que especificar formato y ubicación del archivo.

```
xferlog_enable=YES
```

Utilice la opción **xferlog_file** para especificar el archivo de registro que prefiere utilizar. Aquí se muestra la opción predeterminada:

```
xferlog_file=/var/log/vsftpd.log
```

Controles de acceso vsftpd

Ciertas opciones controlan el acceso al sitio FTP. Como ya se observó, la opción **anonymous_enable** habilita el acceso a usuarios anónimos y **local_enable** permite a los usuarios locales iniciar sesión en sus cuentas. (Si existe `/etc/vsftpd`, no se utiliza un prefijo `vsftpd.` en los archivos.)

Negación de acceso

La opción **deny_email_enable** niega el acceso a usuarios anónimos, así como la opción de archivo **banned_email** designa un archivo (habitualmente, **vsftpd.banned**), para almacenar direcciones de correo electrónico de esos usuarios. El archivo **vsftpd.ftputers** muestra una lista de usuarios que no tendrán nunca acceso al sitio FTP. Suele tratarse de usuarios de sistema, como **root**, **mail** y **nobody**. Consulte la tabla 22-3 para conocer una lista de archivos `vsftpd`.

Acceso de usuario

La opción **userlist_enable** controla el acceso de los usuarios, negando el acceso a quienes aparecen en la lista del archivo designado por la opción **userlist_file** (generalmente **vsftpd.user_list**). Si, en cambio, quiere restringir el acceso únicamente a ciertos usuarios seleccionados,

Archivo	Descripción
vsftpd.ftputers	Muestra una lista de usuarios a los que siempre se niega el acceso
vsftpd.user_list	Se niega el acceso a usuarios especificados (se les permite si userlist_deny es NO)
vsftpd.chroot_list	Muestra una lista de usuarios locales a los que se permite el acceso (se niega el acceso si chroot_local_user está activo)
/etc/vsftpd/vsftpd.conf	Archivo de configuración vsftpd (o <code>/etc/vsftpd/vsftpd.conf</code>)
/etc/pam.d/vsftpd	Secuencia de comandos vsftpd de PAM
/etc/rc.d/init.d/vsftpd	Secuencia de comandos de servidor vsftpd de servicio, independiente
/etc/xinetd.d/vsftpd	Secuencia de comandos vsftpd de xinetd

TABLA 22-3 Archivos para vsftpd

cambie el significado y utilice el archivo **vsftpd.user_list**, para indicar que sólo esos usuarios tienen acceso, en vez de negárseles el acceso. Para esto, configure la opción **userlist_deny** en **NO** (como opción predeterminada es **YES**). Sólo a los usuarios en la lista del archivo **vsftpd.user_list** se les permitirá el acceso al sitio FTP.

Restricciones de usuario

La opción **chroot_list_enable** controla el acceso de usuarios locales, permitiéndoles acceder sólo a directorios de inicio, mientras restringen el acceso al sistema. La opción **chroot_list_file** designa un archivo (frecuentemente, **vsftpd.chroot**), que muestra las listas de los usuarios a quienes se otorga el acceso. Permite al acceso a todos los usuarios locales con la opción **chroot_local_user**. Si esta opción está habilitada, entonces el archivo designado por **chroot_list_file** tendrá un significado inverso, presentando listas de archivos a los que no se permite el acceso. En el siguiente ejemplo, el acceso de usuarios locales se limita a los que se muestran en la lista **vsftpd.chroot**:

```
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd.chroot_list
```

Autenticación de usuario

El servidor **vsftpd** usa el servicio PAM, para autenticar usuarios locales accediendo a sus cuentas de manera remota, a través de FTP. En el archivo **vsftpd.conf**, la secuencia de comandos PAM que se utiliza para el servidor, se especifica con la opción **pam_service_name**.

```
pam_service_name=vsftpd
```

En el directorio **/etc/pam.d**, encontrará un archivo PAM llamado **vsftpd**, con entradas para controlar el acceso al servidor **vsftpd**. PAM está configurado para autenticar usuarios con cuentas válidas, además de negar el acceso a usuarios en el archivo **/etc/vsftpd.ftpusers**. Aquí se muestra el archivo **/etc/pam.d/vsftpd** predeterminado:

```
##PAM-1.0
auth required pam_listfile.so item=user sense=deny
                        file=/etc/vsftpd.ftpusers onerr=succeed
auth required pam_stack.so service=system-auth
auth required pam_shells.so
account required pam_stack.so service=system-auth
session required pam_stack.so service=system-auth
```

Acceso a comandos

El uso de comandos está muy restringido por **vsftpd**. Casi no se permiten las opciones del comando **ls**, para mostrar listas de archivos. Sólo se admite la operación de búsqueda de archivos con asterisco. Para habilitar el despliegue de listas recursivas de archivos en subdirectorios, debe habilitar el uso de la opción **-R**, al configurar la opción **ls_recurse_enable** en **YES**. Algunos clientes, como **ncftp**, supondrán que la opción recursiva está permitida.

Host virtuales vsftpd

Aunque la capacidad no está integrada de manera inherente en **vsftpd**, puede configurar y establecer el servidor **vsftpd** para permitir hosts virtuales. Un *host virtual* es un sólo servidor FTP, operando como si tuviera dos o más direcciones IP. Entonces se utilizan varias direcciones IP para acceder al mismo servidor. Luego, el servidor utilizará un directorio de usuario FTP y archivos separados para cada host. Con **vsftpd**, esto incluye la creación manual de usuarios y directorios FTP

separados para cada host virtual, junto con archivos de configuración **vsftpd**, separados para cada host virtual en el directorio **/etc/vsftpd**. **vsftpd** se configura para ejecutarse como servicio independiente. Su secuencia de comandos de inicio **/etc/rc.d/init.d/vsftpd** buscará y leerá automáticamente cualquier archivo de configuración similar en la lista del directorio **/etc/vsftpd**.

Si, por otra parte, quiere ejecutar **vsftpd** como servicio **xinetd**, debe crear una secuencia de comandos de servicio de **xinetd**, separada para cada host en el directorio **/etc/xinetd.d**. En realidad, tiene varios servicios **vsftpd**, ejecutándose en paralelo para cada host virtual separado. En el siguiente ejemplo se usan dos direcciones IP para un servidor FTP:

1. Cree un usuario FTP para cada host. Cree directorios para cada host (puede usar el ya establecido para uno de los usuarios). Por ejemplo, para el primer host virtual puede usar **FTP-host1**. Asegúrese de establecer propiedad y permisos indicados del root.

```
useradd -d /var/ftp-host1 FTP-host1
chown root.root /var/ftp-host1
umask 022
mkdir /var/ftp-host1/pub
```

2. Configure dos secuencias de comandos de servicio **vsftpd**, en el directorio **/etc/xinetd.d**. El directorio **vsftpd**, en **/usr/share/doc**, tiene la secuencia de comandos de ejemplo **xinetd**, **vsftpd.xinetd**. Dentro de cada una, inserte un comando **bind** para especificar la dirección IP a la que responderá el servidor.

```
bind 192.168.0.34
```

3. En la misma secuencia de comandos, inserte una entrada **server_args**, especificando el nombre del archivo de configuración que habrá de utilizarse.

```
server_args = vsftpd-host1.conf
```

4. En el directorio **/etc/vsftpd**, cree archivos de configuración separados para cada host virtual. Dentro de cada uno, indique el usuario FTP que creó para cada uno, con la entrada **ftp_username**.

```
ftp_username = FTP-host1
```

Usuarios virtuales vsftpd

Los usuarios virtuales se implementan empleando PAM para autenticar usuarios autorizados. En efecto, está permitiendo el acceso a ciertos usuarios, cuando en realidad no tiene que configurar cuentas para éstos en el sistema del servidor FTP. Primero, cree un archivo de base de datos de inicio de sesión PAM, que habrá de usarse junto con un archivo PAM del directorio **/etc/pam.d**, que accederá a la base de datos. En seguida, cree un usuario FTP virtual, junto con los directorios correspondientes a los que accederán los usuarios virtuales (consulte la documentación de **vsftpd** en **vsftpd.beasts.org**, para conocer información más detallada). Después, en el archivo **vsftpd.conf**, deshabilite el FTP anónimo:

```
anonymous_enable=NO
local_enable=YES
```

y después habilite el acceso de invitado:

```
guest_enable=YES
guest_username=virtual
```

Daemon FTP profesional: ProFTPD

ProFTPD se basa en el mismo diseño que el servidor Web Apache, implementando una estructura de configuración simplificada, que soporta características tan flexibles como alojamiento virtual. ProFTPD es un proyecto de fuente abierta disponible bajo GNU GPL. Puede descargar la versión actual desde su sitio Web en proftpd.org. También encontrará documentación detallada, incluyendo preguntas más frecuentes, manuales y configuraciones de ejemplo. Revise el sitio para conocer nuevos lanzamientos y actualizaciones.

ProFTPD se diseñó para ser extensible, al soportar módulos dinámicos personalizados o provistos por desarrolladores terceros. En la actualidad, existen módulos para suministrar autenticación LDAP y SQL, junto con gran número de módulos mejorados para características como cuotas de usuario, formatos de registro y restricciones de tiempo.

Instalación e inicio

Si instala ProFTPD con los paquetes de distribución, las entradas de configuración requeridas se incluyen en su archivo **proftpd.conf**. Si instala desde un código fuente compilado, tal vez deba modificar las entradas en el archivo **proftpd.conf** incorporado. Asegúrese de que existen usuario y grupo FTP especificados en el archivo **proftpd.conf**.

Para establecer los niveles de ejecución en que iniciará automáticamente, utilice **chkconfig** o **sysv-rc-conf**. El siguiente comando establece que **proftpd** se ejecutará automáticamente en los niveles de ejecución 3 y 5:

```
chkconfig --level 35 proftpd on
```

Para activar el servicio en niveles predeterminados, use **rcconf** o **services-admin**.

Autenticación

La autenticación de usuarios en ProTPD se implementa en PAM, LDAP o SQL. Para LDAP o SQL, necesita cargar el módulo ProFTPD correspondiente. La autenticación PAM es la predeterminada (consulte el capítulo 30, para conocer más información acerca de LDAP y PAM). Asegúrese de haber instalado el archivo **/etc/pam.d/ftp**, para integrar soporte a FTP de PAM. ProFTPD también soporta contraseñas estándar y shadow. La autenticación PAM predeterminada se desactiva configurando la directiva **AuthPAMAuthoritative** en off, permitiendo el uso de otros métodos de autenticación como LDAP. Si su servidor está más oculto en la red local, mediante técnicas de enmascaramiento de IP, utilice la directiva **MasqueradeAddress**, para especificar su nombre de host público, permitiendo que el nombre de host permanezca oculto.

proftpd.conf y .ftpassess

ProFTPD sólo utiliza un archivo de configuración, llamado **proftpd.conf**, localizado en el directorio **/etc**. Las entradas de configuración toman la forma de directivas. Este formato se modela intencionalmente con directivas de configuración de Apache. Con las directivas, puede ingresar información de configuración básica, como nombre de servidor, así como realizar operaciones más complejas, implementar host FTP virtuales, entre ellas. El diseño es suficientemente flexible para permitirle definir características de configuración para directorios, usuarios o grupos particulares.

Para configurar un directorio determinado, utilice el archivo **.ftpassess**, con las opciones de configuración colocadas dentro de ese directorio. Estas opciones de **.ftpassess** tienen precedencia sobre las del directorio **proftpd.conf**. Los archivos **.ftpassess** están diseñados para operar como archivos **.htaccess** en el servidor Web Apache, que configura directorios de sitio Web particulares.

Encontrará una lista completa de parámetros de configuración de ProFTPD en el sitio Web de ProFTPD (proftpd.org) y la documentación de ProFTPD instalada en `/usr/doc`, como parte del paquete de software ProFTPD. Cuando cree una nueva configuración, deberá copiar el archivo de configuración **proftpd.conf** y modificarlo. Luego puede probar su sintaxis usando el comando **proftpd** con la opción **-c** y el nombre del archivo.

```
proftpd -c nuevoarchivo.conf
```

Existen diferentes tipos de directivas. Muchos valores establecidos, como **MaxClients**, para determinar el número máximo de clientes o **NameServer**, definiendo el nombre del servidor FTP. Otros crean bloques que almacenan directivas aplicables a componentes de servidor FTP. Las directivas de bloque se insertan en pares: una directiva de inicio y otra directiva de terminación. Esta última define el final del bloque y está formada por el propio nombre, empezando con una diagonal. Las directivas de bloque toman un argumento, especificando el objeto particular al que se aplicarán las directivas. Para la directiva de bloque **Directory**, debe especificar un nombre de directorio al que se aplicará. La directiva de bloque **<Directory midir>** crea un bloque cuyas directivas dentro de éste aplican al directorio *midir*. El bloque termina con una directiva **</Directory>**. **<Anonymous dir-ftp>** configura el servicio anónimo para su servidor FTP. Necesita especificar el directorio en su sistema, para ser utilizado por su servicio FTP anónimo, como `/var/ftp`. El bloque se termina con la directiva **</Anonymous>**. La directiva de bloque **<VirtualHost direccionhost>** se usa para configurar un servidor FTP virtual específico y debe incluir la dirección IP o nombre de dominio empleado por ese servidor. **</VirtualHost>** es la directiva de terminación. Cualquier directiva colocada en este bloque se aplica a un servidor FTP virtual. La directiva **<Limit permiso>** especifica el tipo de acceso que quiere limitar. Toma como su argumento una de las palabras clave indicando el tipo de permiso que se controlará: **WRITE** para acceso de escritura, **READ** para acceso de lectura, **STOR** para acceso de transferencia (carga de archivos) y **LOGIN** para controlar el inicio de sesión de usuario.

Aquí se muestra un ejemplo del archivo **proftpd.conf** estándar, como parte del paquete de software ProFTPD. Establece un solo servidor y un solo inicio de sesión anónimo. Observe que el **ServerType** predeterminado es **standalone**. Si quiere utilizar **xinetd** para ejecutar su servidor, debe cambiar su entrada a **inetd**. Ejemplos detallados de archivos **proftpd.conf**, mostrando varios FTP anónimos y configuraciones de host virtual, se encuentran en la documentación de ProFTPD, ubicada en `/usr/share/doc`, además del sitio Web ProFTPD, en **proftpd.org**.

```
ServerName          "ProFTPD default Installation"
ServerType          standalone
DefaultServer       on
Port                21
Umask               022
MaxInstances        30
User                nobody
Group               nobody
<Directory /*>
    AllowOverwrite   on
</Directory>
# A Basic anonymous configuration, with one incoming directory.
<Anonymous ~ftp>
    User             ftp
    Group            ftp
    RequireValidShell off
```

```

MaxClients          10
UserAlias            anonymous ftp
DisplayLogin         bienvenida.msg
DisplayFirstChdir    .message
# Limit WRITE everywhere in the anonymous chroot except incoming
<Directory *>
    <Limit WRITE>
        DenyAll
    </Limit>
</Directory>
<Directory incoming>
    <Limit WRITE>
        AllowAll
    </Limit>
    <Limit READ>
        DenyAll
    </Limit>
</Directory>
</Anonymous>

```

Acceso anónimo

Se maneja la directiva de configuración **Anonymous**, para crear un bloque de configuración anónimo en que puede colocar directivas, para configurar su servicio FTP anónimo. La directiva incluye el directorio de su sistema usado para el servicio FTP anónimo. El daemon ProFTPD ejecuta una operación **chroot** en su directorio, haciéndolo el directorio raíz para el usuario remoto que accede al servicio. Como opción predeterminada, se permiten inicios de sesión anónimos, esperando que los usuarios inserten su dirección de correo electrónico como contraseña. Tiene la opción de modificar una configuración anónima para construir servicios anónimos más controlados, como inicios de sesión de invitado y contraseñas obligatorias.

NOTA En el caso de ProFTPD, su directorio FTP anónimo no requiere archivo de sistema. Antes de que ProFTPD ejecute una operación **chroot**, ocultando el resto del sistema al directorio, accede a cualquier archivo de sistema necesario, fuera del directorio, manteniéndolo abierto.

En el siguiente ejemplo se muestra una configuración FTP anónima estándar. La directiva **Anonymous** especifica que **/var/ftp** es el directorio de inicio de FTP anónimo. La directiva **User** especifica el usuario con que se ejecutará el daemon FTP Anonymous, asimismo **Group** indica su grupo. En ambos casos, se utiliza FTP, el nombre de usuario estándar, en casi todos los sistemas para FTP anónimo. Una directiva **Directory**, con el carácter para búsqueda de archivos *****, define un bloque Directory aplicado a todos los directorios y archivos de **/var/ftp**. El símbolo ***** busca coincidencias en todos los archivos y directorios. En la directiva **Directory** se encuentra una directiva **Limit**, usada para colocar controles en un directorio. La directiva toma varios argumentos, incluidos **READ**, para acceso de lectura y **WRITE**, para acceso de escritura. En este ejemplo, la directiva **Limit** define restricciones en las capacidades de escritura de los usuarios. Dentro de la directiva **Limit**, la directiva **DenyAll** niega el permiso de escritura, evitando que los usuarios creen o eliminen archivos y al darles, efectivamente, sólo acceso de lectura. Una segunda directiva **Directory** crea una excepción a esta regla para el directorio de entrada. Un directorio de entrada suele establecerse en sitios FTP, para permitir que los usuarios carguen archivos. Para este directorio, la primera directiva **Limit** evita el acceso **READ** y **WRITE** a usuarios con su directiva

DenyAll, impidiendo que los usuarios eliminen o lean archivos aquí. Sin embargo, la segunda directiva, **Limit**, permite que los usuarios carguen archivos al habilitar únicamente las transferencias (**STOR**) con la directiva **AllowAll**.

Una directiva importante para configuraciones FTP anónimas es **RequireValidShell**. Por opción predeterminada, el daemon FTP primero revisa si el usuario remoto intenta iniciar sesión con una shell válida, como BASH shell o C shell. El daemon FTP obtiene una lista de shells válidas del archivo **/etc/shells**. Si el usuario remoto no tiene shell válida, se niega la conexión. Desactive la revisión al utilizar la directiva **RequireValidShell** y la opción **off**. El usuario remoto entonces inicia sesión usando cualquier tipo de shell.

```
<Anonymous /var/ftp>
  User ftp
  Group ftp
  UserAlias anonymous ftp
  RequireValidShell off
<Directory *>
  <Limit WRITE>
    DenyAll
  </Limit>
</Directory>
# El único comando permitido a la entrada es STOR
# (transfiere archivos del cliente al servidor)
<Directory incoming>
  <Limit READ WRITE>
    DenyAll
  </Limit>
  <Limit STOR>
    AllowAll
  </Limit>
</Directory>
</Anonymous>
```

Recuerde que FTP se creó originalmente para permitir a un usuario remoto conectarse a una cuenta de su propio sistema. Los usuarios inician sesión en diferentes cuentas de su sistema, mediante el servicio FTP. Los usuarios anónimos están restringidos a la cuenta de usuario anónimo. Sin embargo, puede crear otros usuarios y directorios de inicio, funcionando también como cuentas FTP anónimas, con las mismas restricciones. A éstas se les conoce como *cuentas de invitado*. Se requiere que los usuarios remotos sepan el nombre de usuario y, frecuentemente, la contraseña. Una vez conectados, sólo tienen acceso de lectura a los archivos de esa cuenta; el resto del sistema de archivos se oculta. En efecto, está creando un sitio FTP anónimo aparte, en la misma ubicación con acceso más restringido.

Para crear una cuenta de invitado, primero debe crear un usuario y su directorio de inicio. Luego debe crear un bloque **Anonymous** en el archivo **proftpd.conf** para esa cuenta. La directiva **Anonymous** incluye el directorio de inicio del usuario invitado que creó. Especifique su directorio con **~** para la ruta y nombre del directorio, que suele ser el mismo que el nombre de usuario. Dentro del bloque **Anonymous**, utilice las directivas **User** y **Group**, para especificar usuario y nombre de grupo para la cuenta de usuario. Establezca la directiva **AnonRequirePassword** en **on**, si quiere que usuarios remotos provean una contraseña. Una directiva **UserAlias** define los alias para el nombre de usuario. Un usuario remoto utiliza el alias o nombre de usuario original para iniciar sesión. Luego inserte las directivas restantes para controlar el acceso a archivos y directorios, en el

directorio de inicio de la cuenta. Aquí se muestra un ejemplo de directivas iniciales. La directiva **User** especifica el usuario **miproyecto**. El directorio de inicio es **~miproyecto**, que suele evaluarse como **/var/miproyecto**. La directiva **UserAlias** concede a los usuarios remotos iniciar sesión, ya sea con el nombre **miproyecto** o **mipostre**.

```
<Anonymous ~miproyecto>
  User miproyecto
  Group other
  UserAlias mipostre miproyecto
  AnonRequirePassword on
</Directory *>
```

Con la misma facilidad puede crear una cuenta que no requiera contraseña, permitiendo a los usuarios insertar, en cambio, su dirección de correo electrónico. En el siguiente ejemplo se configura un usuario anónimo llamado **misimagenes**. No requiere contraseña ni shell válida. Aún es necesario que el usuario remoto sepa el nombre de usuario, en este caso **misimagenes**.

```
<Anonymous /var/misimagenes>
  AnonRequirePassword off
  User misimagenes
  Group nobody
  RequireValidShell off
</Directory *>
```

En el siguiente ejemplo se suministra un tipo más genérico de inicio de sesión de invitado. El nombre de usuario es **guest**, con el directorio de inicio ubicado en **~guest**. Es obligatorio que los usuarios remotos conozcan la contraseña para la cuenta de invitado. La primera directiva **Limit** permite a los usuarios iniciar sesión. La segunda directiva **Limit**, ofrece acceso de escritura de usuarios en una red específica, como indica la dirección IP de red, negando el acceso de escritura a cualquier otro.

```
<Anonymous ~guest>
  User          guest
  Group         nobody
  AnonRequirePassword      on
  <Limit LOGIN>
    AllowAll
  </Limit>
  # Niega el acceso de escritura a todos menos los host confiables.
  <Limit WRITE>
    Order          allow,deny
    Allow          from 10.0.0.
    Deny           from all
  </Limit>
</Anonymous>
```

Servidores FTP virtuales

El daemon ProFTPD administra más de un sitio FTP a la vez. Al utilizar la directiva **VirtualHost**, en el archivo **proftpd.conf**, se crea un conjunto independiente de directivas configurando un servidor FTP separado. La directiva **VirtualHost** suele utilizarse para configurar servidores virtuales, como sitios FTP. Configure su sistema para soportar más de una dirección IP.

Las direcciones IP adicionales se usan para servidores virtuales, no para máquinas independientes. Utilice estas direcciones IP adicionales para configurar un servidor FTP virtual, que le da otro sitio FTP en el mismo sistema. Este servidor agregado usará la dirección IP adicional como propia. Los usuarios remotos tendrán acceso a ella mediante esa dirección IP, en lugar de la dirección IP principal del sistema. Debido a que dicho servidor FTP no se ejecuta de manera independiente en una máquina separada, sino la misma, se le conoce como *servidor FTP virtual* o *host virtual*. Esta característica le permite ejecutar, en la misma máquina, lo que a otros aparece como varios servidores FTP diferentes. Cuando un usuario remoto utiliza la dirección IP del servidor FTP virtual para acceder a éste, el daemon ProFTPD detecta esa solicitud y opera como el servidor FTP de ese sitio. ProFTPD maneja gran cantidad de sitios virtuales FTP, al mismo tiempo, en una sola máquina.

NOTA *Dadas sus capacidades de configuración, también puede adecuar cualquiera de los sitios FTP virtuales a roles específicos, como un sitio de invitado, un sitio anónimo para un grupo particular o un sitio anónimo para un usuario particular.*

Para configurar un servidor FTP virtual, se inserta una directiva **<VirtualHost>** para éste en un archivo **proftpd.conf**. Esta entrada comienza con la directiva **VirtualHost**, la dirección IP y termina con una directiva de **VirtualHost** final, **</VirtualHost>**. Cualquier directiva colocada dentro de éstas, se aplica al host virtual. En el caso de sitios anónimos o de invitado, agregue las directivas **Anonymous** y **Guest**. Incluso puede agregar directivas **Directory** para directorios específicos. Con la directiva **Port**, en una configuración independiente, puede crear un host virtual operando en el mismo sistema, pero conectada en un puerto diferente.

```
<VirtualHost 10.0.0.1>
ServerName "Mi servidor FTP virtual"
</VirtualHost>
```

Las configuraciones para **xinetd** e independientes, manejan los hosts virtuales de forma diferente. El proceso **xinetd** detecta una solicitud para un host virtual y la pasa al daemon FTP. Entonces, el daemon FTP examina direcciones y puertos especificados en la solicitud, procesando la solicitud para el host virtual apropiado. En la configuración independiente, el daemon FTP escucha continuamente solicitudes en todos los puertos especificados y genera procesos secundarios para manejar los de diferentes hosts virtuales, mientras llegan. En la configuración independiente, ProFTPD permite gran número de host virtuales simultáneamente.

En el siguiente ejemplo se muestra una configuración de ejemplo de un host FTP virtual. La directiva **VirtualHost** utiliza direcciones de nombre de dominio para sus argumentos. Cuando se utiliza una dirección de nombre de dominio, debe asociarse con una dirección IP en el servidor de nombre de dominio de la red. La dirección IP, a cambio, debe referirse a la máquina en que se está ejecutando el daemon ProFTPD. En el servidor FTP virtual **ftp.misimagenes.com**, se configura una cuenta de invitado anónimo llamada **imagenesderob**, necesitando una contraseña para iniciar sesión. También se configura una cuenta FTP anónima que utiliza el directorio de inicio **/varftp/virtual/pics**.

```
<VirtualHost ftp.misimagenes.com>

ServerName "Servidor FTP misimagenes"
MaxClients 10
MaxLoginAttempts 1
```



```
DeferWelcome          on
<Anonymous ~imagenesderob>
  User                imagenesderob
  Group               imagenesderob
  AnonRequirePassword on
<Anonymous /var/ftp/virtual/pics>
  User                ftp
  Group               ftp
  UserAlias           anonymous ftp
</Anonymous>
</VirtualHost>
```

Servidores Web

Las distribuciones Linux ofrecen varios servidores Web para su sistema. El servidor Web principal es Apache, que casi se ha vuelto el estándar de servidor Web en distribuciones Linux. Es un sistema muy poderoso, estable y fácil de configurar. Otros servidores Web también están disponibles, como Tux, más pequeño, pero muy rápido y eficiente para manejar datos Web que no cambian. Las distribuciones de Linux cuentan con varias configuraciones predeterminadas de servidores Web, que permite utilizarlas apenas se instalan.

De manera gratuita Apache soporta la capa de conector seguro (SSL, Secure Socket Layer) al utilizar OpenSSL. También existen productos de criptografía privados, por lo que sólo se pagan los costos de licencia. En vez de obtener la licencia directamente, sólo compre una versión comercial de Apache incluyendo licencias como Stronghold y Raven (**covalent.net**). Al principio, este tipo de restricción aplicaba para el uso de tecnologías RSA, sólo en Estados Unidos, donde se patentó alguna vez. La patente RSA ha expirado desde entonces y ahora está disponible para utilizarse en productos distribuidos como OpenSSL.

Tux

Tux, el acelerador de contenido Red Hat, es un servidor Web de contenido estático diseñado para ejecutarse con gran rapidez en el kernel de Linux. En efecto, se ejecuta en el espacio del kernel, haciendo los tiempos de respuesta más rápidos que los de servidores Web estándar, en el espacio del usuario, como Apache. Debido a que es un servidor que funciona en el espacio del kernel, se coordina con un servidor de espacio del usuario, como Apache, para ofrecer contenido dinámico, como programas CGI. Tux incluso puede hacer uso de la caché para almacenar contenido dinámico generado previamente, al utilizarlo como si fuera estático. La capacidad para coordinarse con un servidor Web en el espacio del usuario, permite usar Tux como servidor Web primario. Cualquier cosa que Tux no pueda manejar, pasará al servidor Web de espacio del usuario.

NOTA Tux se distribuye de manera gratuita bajo la licencia pública GNU y se incluye con muchas distribuciones.

El archivo de configuración de Tux se ubica en **/proc/sys/net/tux**. Aquí puede introducir parámetros como **serverport**, **max_doc_size** y **logfile** (revise el manual de referencia de Tux en **redhat.com/docs/manuals/tux** para conocer una lista detallada). Las opciones predeterminadas

ya están ingresadas; **serverport**, **clientport** y **documentroot** son parámetros requeridos que deben establecerse. **serverport** es el puerto que usará Tux (80 si es el servidor Web primario). **clientport** es el puerto usado por el servidor Web de espacio de usuario con el que se coordina Tux, como Apache. **documentroot** especifica el directorio root para sus documentos Web (**/var/www/html** en Red Hat y Fedora).

De manera ideal, Tux se ejecuta como servidor Web primario y Apache como secundario. Si desea configurar Apache para ejecutarse con Tux, la entrada de puerto en el archivo **httpd.conf** Apache necesita cambiarse de 80 a 8080.

Port 8080

Varios parámetros, como **DOCROOT**, se especifican como argumentos para este comando de Tux. Insértelos en el archivo **/etc/sysconfig/tux**.

NOTA También puede ejecutar Tux como servidor FTP. En el directorio **/proc/sys/net/tux**, cambie el contenido del archivo **serverport** a 21, **application_protocol** a 1 y **nonagle** a 0 y después reinicie Tux. Utilice el comando **generatetuxlist** en el directorio raíz del documento, para generar listas de directorio FTP.

Servidores Web alternos

Entre otros servidores Web disponibles para Linux se incluyen Stronghold Enterprise Server y el servidor Apache-SSL. Aquí se proporciona una lista:

- Apache-SSL (**Apache-ssl.org**) es un servidor Web de cifrado basado en Apache y OpenSSL (**openssl.org**).
- lighthttpd (**lighthttpd.net/**) es un servidor Web pequeño y muy rápido.
- El servidor Web de Sun Java System (**sun.com**) presenta soporte y seguridad al desarrollo de Java.
- El servidor de aplicaciones Zope (**zope.org**) es un servidor Web de fuente abierta, con características como seguridad integrada, administración y desarrollo basado en Web, además de una interfaz de base de datos. Fue desarrollado por Zope Corporation, que también desarrolló el lenguaje de programación Python.
- Stronghold Enterprise Server (**redhat.com/software/stronghold**) es una versión comercial del servidor Web Apache presentando seguridad y herramientas de administración mejoradas.
- Netscape Enterprise Server (**enterprise.netscape.com**), parte de las soluciones de seguridad de Netscape, presenta estándares abiertos con gran desempeño.
- También puede utilizar el servidor Web de NCSA original, aunque ya no está bajo desarrollo ni tiene soporte (**hoohoo.ncsa.uiuc.edu**).

Servidor Web Apache

El servidor Web Apache es un servidor HTTP (Web) gratuito, con características completas, desarrollado y mantenido por Apache Server Project. El objetivo del proyecto es proporcionar un servidor Web confiable, eficiente y fácilmente expansible; el código fuente abierto gratuito se pone a

disposición del usuario, a través de su propia licencia Apache Software License. El software de servidor incluye varios daemon de servidor, archivos de configuración, herramientas de administración y documentación. Apache Server Project es mantenido por un grupo central de programadores voluntarios y tiene soporte para gran cantidad de contribuyentes de todo el mundo. Apache Server Project es uno de varios proyectos con soporte de la Apache Software Foundation (antes conocida como Apache Group). Esta organización sin fines de lucro proporciona soporte financiero, legal y organizacional a varios proyectos del software Apache Open Source, incluidos Apache HTTPD Server, Java Apache, Jakarta y XML-Apache. El sitio Web de Apache Software Foundation es **Apache.org**. En la tabla 23-1 se muestra una lista de varios sitios Web relacionados con Apache.

Apache se basó originalmente en el servidor Web NCSA, desarrollado por el National Center for Supercomputing Applications, de la Universidad de Illinois en Urbana-Champaign. Desde entonces, Apache ha surgido como un servidor por derecho propio y se ha vuelto uno de los servidores Web más populares en uso. Aunque originalmente se desarrolló para sistemas Unix y Linux, Apache se ha vuelto una aplicación de varias plataformas con versiones para Windows y OS/2. Apache ofrece soporte y documentación en línea a su servidor Web en **httpd.apache.org**. Un manual basado en HTML también se brinda con la instalación del servidor. Utilice la Herramienta de configuración de Apache como ayuda para configurar su servidor Apache de manera sencilla. Opera en cualquier administrador de ventana X Windows System, incluidos GNOME y KDE. Además, puede usar la herramienta de configuración Comanche. Webmin conf también proporciona soporte para configuración de Apache.

Java: Apache Jakarta Project

Apache Jakarta Project permite el desarrollo de software de Open Source Java; su sitio Web se ubica en **jakarta.apache.org**. Actualmente, Jakarta soporta varios proyectos, incluidas bibliotecas, herramientas, marcos conceptuales, motores y aplicaciones de servidor. Tomcat es una implementación de fuente abierta de Java Servlet con especificaciones JavaServer Pages. Tomcat está diseñado para utilizarse en servidores Apache. JMeter es una herramienta de escritorio de Java para probar el rendimiento de recursos de servidor, como servlets y secuencias de comandos CGI. Velocity es un motor de plantillas para acceso sencillo a objetos de Java. Watchdog es una

Sitio Web	Descripción
apache.org	Apache Software Foundation
httpd.apache.org	Apache HTTP Server Project
jakarta.apache.org	Jakarta Apache Project
apache-gui.com/	Apache GUI Project
comanche.org	Comanche (Administrador de configuración para Apache)
apache-ssl.org	Servidor SSL Apache
openssl.org	Proyecto OpenSSL (capa de conector seguro)
modssl.org	El proyecto SSL module (mod_ssl) para agregar cifrado SSL a un servidor Web Apache
php.net	PHP Hypertext Preprocessor, lenguaje de programación incrustado de páginas Web.

TABLA 23-1 Sitios Web relacionados con Apache

herramienta revisando la compatibilidad de contenedores de servlet. Struts, Cactus y Tapestry son marcos conceptuales de Java, métodos establecidos para desarrollar aplicaciones Web de Java.

Instalaciones de Apache en Linux

Su distribución de Linux suele ofrecer la opción de instalar el servidor Web Apache durante la instalación inicial del sistema Linux. Todos los directorios y archivos de configuración necesarios se generan automáticamente. Entonces, siempre que ejecute Linux, su sistema ya será un sitio Web totalmente funcional. Cada vez que inicia su sistema, el servidor Web también inicia, ejecutándose continuamente. En casi todas las distribuciones, el directorio reservado para sus archivos de datos del sitio Web es `/var/www/html`. Coloque sus páginas Web en este directorio o cualquier subdirectorio. Su sistema ya estará configurado para operar como servidor Web. Todo lo que necesita es realizar cualquier configuración de servidor de red necesaria y después designar que archivos y directorios estén abiertos para el usuario remoto. No necesita hacer más. Una vez que su sitio Web está conectado a una red, los usuarios remotos accederán a éste.

En general, el servidor Web establece su sitio Web en el directorio `/var/www`. También varios directorios para administrar el sitio. El directorio `/var/www/cgi-bin` almacena las secuencias de comandos CGI y `/var/www/html/manual` almacena el manual de Apache en formato HTML. Puede utilizar su explorador para examinarlo. Sus páginas Web se colocan en el directorio `/var/www/html`. Coloque ahí la página de inicio de su sitio Web. Sus archivos de configuración se ubican en un directorio diferente, `/etc/httpd/conf`. En la tabla 23-2 se muestra una lista de varios directorios y archivos de configuración del servidor Web Apache.

Directorios de Sitios Web	Descripción
<code>/var/www/html</code>	Archivos HTML del sitio Web
<code>/var/www/cgi-bin</code>	Archivos de programa CGI
<code>/var/www/html/manual</code>	Manual del servidor Web Apache
Archivos de configuration	
<code>.htaccess</code>	Archivos de configuración basados en directorio; un archivo <code>.htaccess</code> almacena directivas para controlar el acceso a archivos dentro del directorio en que está localizado
<code>/etc/httpd/conf</code>	Directorio de archivos de configuración del servidor Web Apache
<code>/etc/httpd/conf/httpd.conf</code>	Archivo de configuración del servidor Web Apache
<code>/etc/httpd/conf.d</code>	Directorio que almacena archivos de configuración de módulo, como <code>ssl.conf</code> para SSL y <code>php.conf</code> para PHP
Application and Module Files	
<code>/usr/sbin</code>	Ubicación del archivo de programación y utilerías del servidor Web Apache
<code>/usr/share/doc/</code>	Documentación del servidor Web Apache
<code>/var/log/http</code>	Ubicación de los archivos de registro de Apache
<code>/etc/httpd/modules</code>	Directorio para almacena módulos de Apache
<code>/etc/httpd/run</code>	Directorio que almacena ID de procesos de Apache

TABLA 23-2 Archivos y directorios del servidor Web Apache (instalación RPM)

Módulos de multiprocesamiento de Apache: MPM

Apache ahora utiliza una nueva arquitectura con módulos de multiprocesamiento (MPM, MultiProcessing Modules), diseñados para personalizar Apache para diferentes sistemas operativos, además de manejar ciertas operaciones de multiprocesamiento. Para el MPM principal, un sistema Linux utiliza el MPM prefork o worker, mientras Windows utiliza el MPM mpm_winnt. prefork es un módulo MPM estándar diseñado para ser compatible con sistemas Unix y Linux más antiguos, sobre todo los que no soportan subprocessos. Puede configurar los parámetros de carga de trabajo para ambos en el archivo de configuración de Apache, `/etc/httpd/conf/httpd.conf`.

Muchas directivas que una vez residieron en el núcleo de Apache ahora se colocan en módulos y MPM respectivos. Con su diseño modular, se han eliminado varias directivas, como `ServerType`. Los archivos de configuración para tales módulos se ubican en el directorio `/etc/httpd/conf.d`.

Inicio y detención del servidor Web

En casi todos los sistemas, Apache se instala como servidor independiente, que se ejecuta continuamente. Con las secuencias de comandos `init`, su sistema inicia automáticamente el daemon del servidor Web, invocándolo siempre que inicia su sistema.

En Red Hat, Fedora, SUSE y distribuciones similares, se utiliza el comando `chkconfig`, para configurar los niveles de ejecución en que iniciará el servidor `httpd`, al crear vínculos con directorios de nivel de ejecución apropiados. El siguiente comando configurará el servidor Web (`httpd`) para iniciar en los niveles de ejecución 3 y 5:

```
chkconfig --level 35 httpd on
```

En Debian, Ubuntu y distribuciones similares use las herramientas `rrconf` o `sysv-rc-conf`. En distribuciones usando GNOME, utilice la herramienta `services-admin`.

NOTA Una secuencia de comandos de servicio para el servidor Web llamada `httpd` (apache2 en Debian) se encuentra en el directorio `/etc/rc.d/init.d`. Utilice el comando `service` para iniciar y detener manualmente el servidor `httpd`: `service httpd start`.

Apache también proporciona una herramienta de control llamada `apachectl` (Apache control) para administrar su servidor Web. Con `apachectl`, se inicia, detiene y reinicia el servidor desde la línea de comandos. El comando `apachectl` toma varios argumentos: **start** para iniciar el servidor, **stop** para detenerlo, **restart** para apagar y reiniciar el servidor y **graceful** para apagar y reiniciar de manera elegante. Además, se usa `apachectl` para revisar la sintaxis de sus archivos de configuración con el argumento **config**. También utiliza `apachectl` como archivo de servicio de sistema, para su servidor en el directorio `/etc/rc.d`.

Puede llamar al daemon directamente usando su nombre de ruta completo. Este daemon tiene varias opciones. La opción **-d** le permite especificar un directorio para el programa `httpd`, si es diferente del directorio predeterminado. Con la opción **-f**, se especifica un archivo de configuración diferente desde `httpd.conf`. La opción **-v** despliega la versión.

```
/usr/sbin/httpd -v
```

Para revisar su servidor Web, inicie su explorador Web e inserte la dirección de nombre de dominio de Internet de su sistema. Para el sistema `tortuga.mipista.com`, el usuario inserta `http://tortuga.mipista.com`. Esto deberá desplegar la página de inicio colocada en su directorio raíz de Web. Una forma simple de hacer esto es utilizar Lynx, el explorador Web de línea de comandos.

Inicie Lynx y escriba **g** para abrir una línea donde se pueda escribir el URL de su propio sistema. Lynx despliega la página de inicio de su sitio Web. Primero, asegúrese de colocar un archivo **index.html** en el directorio **/var/www/html**.

Una vez tenga su servidor en ejecución, revise su rendimiento con la herramienta de medición comparativa **ab**, también ofrecida con Apache: **ab** muestra cuántas solicitudes puede manejar su servidor al mismo tiempo. Entre las opciones se incluyen **-v**, que permite controlar el nivel de detalle desplegado; **-n**, especifica el número de solicitudes que habrán de manejarse (la opción predeterminada es 1); y **-t**, especifica un límite de tiempo.

*NOTA En la actualidad, no hay soporte para ejecutar Apache bajo **xinetd**. En Apache 2, ese soporte se determina al seleccionar un módulo MPM diseñado para ejecutarse en **xinetd**.*

Archivos de configuración de Apache

Las directivas de configuración se colocan en el archivo de configuración **httpd.conf**. Una versión documentada del archivo de configuración **httpd.conf** se instala automáticamente en **/etc/httpd/conf**. Se recomienda consultar este archivo en su sistema. Contiene descripciones detalladas de entradas predeterminadas para directivas Apache.

Es posible sobrescribir en cada directorio cualquiera de las directivas en los archivos de configuración principales, a través del archivo **.htaccess**, ubicado en un directorio. Aunque se diseñó originalmente sólo para directivas de acceso, el archivo **.htaccess** también almacena cualquier directiva de recursos, permitiéndole personalizar la manera en que las páginas Web se despliegan en un directorio particular. Configure el acceso a archivos **.htaccess** en el archivo **httpd.conf**.

Además, muchos de los módulos proporcionados para Apache tienen sus propios archivos de configuración. Estos se colocan en el directorio **/etc/httpd/conf.d**.

Configuración y directivas de Apache

Las operaciones de configuración de Apache toman la forma de directivas insertadas en archivos de configuración de Apache. Con estas directivas, puede insertar información de configuración básica, como el nombre de su servidor o realizar operaciones mucho más complejas, como implementar hosts virtuales. El diseño es suficientemente flexible para permitirle definir características de configuración para directorios particulares y diferentes hosts virtuales. Apache tiene directivas diferentes que realizan operaciones tan diversas como controlar el acceso a directorio, asignando formatos de iconos de archivos y crear archivos de registro. Casi todas las directivas establecen valores como **DirectoryRoot**, para almacenar el directorio raíz para páginas Web del servidor o **Port**, que almacena el puerto del sistema en que el servidor escucha las solicitudes. Aquí se muestra la sintaxis para una directiva simple:

```
directive opcion opcion ...
```

Ciertas directivas crean bloques permitiendo almacenar directivas aplicables a componentes de servidor específicos (también conocidas como directivas de sección). Por ejemplo, la directiva **Directory** se utiliza para definir un bloque en el que se colocan directivas sólo aplicables a un directorio particular. Las directivas de bloque se insertan en pares: una directiva de inicio y otra de terminación. Ésta última define el final del bloque y está integrada por el mismo nombre, empezando con una diagonal. Las directivas de bloque toman un argumento especificando el objeto particular al que aplicarán las directivas. Para la directiva de bloque **Directory**, debe especificar un

nombre de directorio al que se aplicará. La directiva de bloque **<Directory *midir*>** crea un bloque cuyas directivas dentro de éste aplican al directorio *midir*. El bloque termina por una directiva **</Directory>**. La directiva de bloque **<VirtualHost *direcciónhost*>** se usa para configurar un servidor Web virtual específico y debe incluir la dirección IP o nombre de dominio utilizado para ese servidor. **</VirtualHost>** es la directiva de terminación. Cualquier directiva colocada en este bloque aplicará a un servidor Web virtual. La directiva **<Limit *metodo*>** especifica el tipo de método de acceso que quiere limitar, como GET o POST. Las directivas de control de acceso ubicadas en el bloque muestran una lista de controles colocadas en esos métodos. La sintaxis para una directiva de bloque es la siguiente:

```
<block-directive opcion ... >
  directive opcion ...
  directive opcion ...
</block-directive>
```

Generalmente, las directivas se colocan en uno de los archivos de configuración principales. Las directivas de directorio encontradas en esos archivos se utilizan para configurar un directorio particular. Sin embargo, Apache también usa archivos de configuración basados en directorio. Cualquier directorio puede tener su propio archivo **.htaccess** almacenando directivas para configurar sólo ese directorio. Si su sitio tiene muchos directorios o cualquiera de los directorios requiere configuraciones especiales, puede colocar sus directivas de configuración en sus archivos **.htaccess**, en vez de llenar el archivo de configuración principal con directivas **Directory**, para cada uno. Puede controlar cuáles directivas de un archivo **.htaccess** tendrán precedencia sobre las que se encuentran en los archivos de configuración. Si su sitio permite directorios controlados por el usuario o cliente, tal vez quiera monitorear de manera cuidadosa el uso de archivos o deshabilitar **.htaccess** en éstos. (Es posible que las directivas de un archivo **.htaccess**, sobrescriban las que se encuentran en archivos de configuración estándar, a menos que se deshabiliten con directivas **AllowOverride**.)

Configuración global

La configuración de Apache estándar tiene tres secciones, correspondientes a configuraciones globales, configuraciones de servidor y host virtual. Las configuraciones globales controlan operaciones básicas y el rendimiento del servidor Web. Aquí se establece la configuración de ubicaciones, archivos de ID de proceso, tiempos, configuraciones para el módulo MPM utilizado y los módulos de Apache que se cargan.

La directiva **ServerTokens** evita el develamiento de cualquier módulo opcional que su servidor esté utilizando. La directiva **ServerRoot** especifica dónde se almacenan sus archivos de configuración, error y registro del servidor Web. Se trata de **/etc/httpd**, que también incluye archivos de error y registro, además de módulos de servidor. El directorio raíz del servidor se utiliza entonces como prefijo para otras entradas de directorio.

```
ServerRoot /etc/httpd
```

El archivo de ID de procesos (PID) del servidor, suele ser **/etc/httpd/run/httpd.pid**, como se establece con **PidFile**.

```
PidFile run/httpd.pid
```

Los tiempos de conexión y petición se manejan con directivas **Timeout**, **KeepAlive**, **MaxKeepAlive** y **KeepAliveTimeout**. **Timeout** es el tiempo en segundos que el servidor Web espera para terminar el envío o recepción de una solicitud. **KeepAlive** permite conexiones

persistentes y solicitudes de un cliente en la misma conexión. Esto se desactiva como opción predeterminada.

KeepAliveRequest establece el número máximo de solicitudes en una conexión persistente. **KeepAliveTimeout** es el tiempo que una conexión determinada con un cliente se mantiene abierta para recibir más solicitudes de ese cliente.

La directiva **Listen** unirá el servidor a una dirección IP o puerto específico. Como opción predeterminada este puerto es 80.

```
Listen 80
```

Módulos

Gran parte del poder y flexibilidad del servidor Web Apache proviene de su uso de módulos para extender sus capacidades. Apache se implementa con un conjunto central de directivas. Es posible crear módulos almacenando definiciones de otras directivas. Éstos se cargan en Apache, permitiéndole utilizar dichas directivas en su servidor. Un conjunto estándar de módulos incluido con la distribución de Apache, aunque puede descargar otros e incluso crear sus propios módulos. Por ejemplo, el módulo `mod_autoindex` almacena directivas para indizar directorios automáticamente (como se describe en la siguiente sección). El módulo `mod_mime` almacena el tipo MIME y directivas de manejador. Los módulos se cargan con la directiva **LoadModule**. Encontrará directivas **LoadModules** en el archivo de configuración **httpd.conf**, para casi todos los módulos estándar.

```
LoadModule mime_module modules/mod_mime.so
```

LoadModule toma como argumentos nombre del módulo y ubicación. Los módulos se almacenan en el directorio **/etc/httpd/modules**, al que se hace referencia aquí, con el prefijo **modules/**.

Los archivos de configuración de diferentes módulos están localizados en el directorio **/etc/httpd/conf.d**. También se cargan usando la directiva **Include**. La siguiente opción inserta todos los archivos de configuración (que tienen una extensión **.conf**) en el directorio **/etc/httpd/conf.d**.

```
Include conf .d/*.conf
```

La aplicación `apxs` incluida en el paquete Apache, se usa para construir módulos de extensión de Apache. Con la aplicación `apxs`, se compila el código fuente del módulo Apache en C y se crean objetos compartidos dinámicamente, cargables con la directiva **LoadModule**. La aplicación `apxs` requiere que el módulo `mod_so` sea parte de su aplicación Apache. Incluye amplias opciones, como **-n** para especificar el nombre del módulo, **-a** para agregar una entrada a éste en el archivo **httpd.conf** e **-i** para instalar el módulo en su servidor Web.

Encontrará una lista completa de directivas de configuración Web de Apache en el sitio Web de Apache, **httpd.apache.org** y el manual ubicado en su directorio raíz de su sitio Web. En muchos sistemas, éste se localiza en el subdirectorio `manual`, del directorio predeterminado del sitio Web, configurado por la distribución (**/var/www/manual**).

Configuración de MPM

Las opciones de configuración de los módulos `prefork` y `worker` de MPM, permiten personalizar su servidor Web Apache de acuerdo con demandas de carga de trabajo. Las entradas predeterminadas ya estarán configuradas para un servidor Web estándar, operando bajo una carga ligera. Puede modificar estas configuraciones para diferentes demandas.

Dos módulos MPM comunes en sistemas Unix y Linux son `prefork` y `worker`. El módulo `prefork` soporta un subproceso por proceso, lo que mantiene compatibilidad con otros sistemas y módulos más antiguos. El módulo `worker` soporta varios subprocesos para cada proceso, colocando una

carga mucho menor sobre los recursos del sistema. Comparten varias de las mismas directivas, como **StartServer** y **MaxRequestPerChild**.

Apache ejecuta un sólo proceso principal, con todos los procesos secundarios necesarios para manejar solicitudes. La configuración de los módulos MPM se concentra en el número de procesos disponibles. El módulo **prefork** mostrará una lista de números de servidor, a medida que inicia cada proceso por servidor; el módulo **worker** controlará los subprocesos, porque usa subprocesos para cada proceso. La directiva **StartServer** presenta la lista de números de procesos del servidor que iniciarán ambos módulos. En general, serán mayores para el módulo **prefork** que para el **worker**.

En el módulo **prefork** necesita establecer configuraciones mínimas y máximas para servidores sobrantes. **MaxClients** establece el número máximo de servidores que pueden iniciarse y **ServerLimit** establece el número de servicios permitidos. **MaxRequestsPerChild** establece el número máximo de solicitudes permitidas a los usuarios.

En el módulo **worker**, **MaxClients** también establece el número máximo de subprocesos de clientes y **ThreadsPerChild** establece el número de subprocesos para cada servidor. **MaxRequestsPerChild** limita el número máximo de solicitudes por servidor. También se configuran los límites de subprocesos sobrantes.

Las directivas sirven como una especie de acelerador para el acceso al servidor Web, controlando procesos para mantener disponibles y limitar los recursos que se usarán. En la configuración **prefork**, el número de **StartServer** se establece en 8, el mínimo de servidores sobrantes en 5 y el máximo en 20. Esto significa que al principio se iniciarán 8 procesos por servidor (mismos que esperarán solicitudes), junto con 5 procesos sobrantes. Cuando ya no se usen los procesos de servidor, se terminarán hasta que el número de estos procesos sea menor a 20. El número máximo de procesos de servidor que iniciará es 256. El número máximo de conexiones por proceso de servidor se establece en 4000.

En el MPM **worker**, al principio sólo se inician 2 procesos de servidor. Se configura un mínimo de 25 subprocesos sobrantes y un máximo de 75. El número máximo de subprocesos se establece en 150, con 25 subprocesos por proceso secundario.

Configuración de servidor

Ciertas directivas se usan para configurar operaciones generales de su servidor. Estas directivas se colocan en medio del archivo de configuración **httpd.conf**, directamente bajo la sección etiquetada **Server Settings**. Algunas directivas requieren nombres de ruta, mientras otras sólo necesitan activarse o desactivarse con las palabras clave **on** y **off**. El archivo **httpd.conf** predeterminado ya contiene estas directivas. Algunas se comentan con un símbolo **#** al principio. Muchas de estas entradas se anteceden con comentarios explicando su propósito.

El siguiente es un ejemplo de la directiva **ServerAdmin**, usada para establecer la dirección donde los usuarios envían correo para problemas administrativos. Reemplace la entrada **usted@su.direccion** con la dirección que quiera utilizar para recibir correo relacionado con la administración del sistema. Como opción predeterminada, se establece en **root@localhost**.

```
# ServerAdmin: Su dirección, a la que deben enviarse por correo electrónico las
descripciones de problemas.
ServerAdmin usted@su.direccion
```

Generalmente, un servidor Web utiliza el puerto 80, el predeterminado de Apache. Si quiere utilizar un puerto diferente, especifíquelo con la directiva **Port**.

Port 80

La directiva **ServerName** contiene el nombre de host de su servidor Web. Es importante especificar un nombre de host para evitar fallas innecesarias en las búsquedas DNS que pueden dejar a su servidor en espera. Observe que la entrada se comenta con un signo # al principio. Sólo elimine el # y escriba el nombre de host de su servidor Web en lugar de *nuevo.nombre.host*. Si utiliza un puerto diferente de 80, asegúrese de especificarlo junto a su nombre de host, como en **tortuga.mipista.com:80**. Aquí se muestra la entrada predeterminada original:

```
# ServerName permite establecer un nombre host enviado
# de regreso a los clientes de su servidor, si es diferente del
# que debería recibir el programa (es decir utilice
# "www" en vez del nombre real del host).

#ServerName nuevo.nombre.host:80
```

Una entrada modificada de **ServerName** se vería así:

```
ServerName tortuga.mipista.com
```

Cuando recibe solicitudes URL para el sistema de servidor, como las de los archivos locales del sistema, la directiva **UseCanonicalName** usará las directivas **ServerName** y **Port** para generar el nombre de servidor del URL de host. Cuando está desactivado, sólo usará el nombre proporcionado por la solicitud del cliente. Esto puede ser confuso si conoce al servidor Web por un nombre, pero utiliza otro, como **www.mipista.com** empleado para hacer referencia a *tortuga.mipista.com*. **UseCanonicalName** se establece para evitar este problema, al generar el URL local correcto.

En sistemas Linux, ya se han hecho entradas para la instalación del servidor Web estándar, como **/var/www** del directorio de su sitio Web. Personalice su sitio Web de acuerdo con sus propias necesidades, cambiando las directivas apropiadas. La directiva **DocumentRoot** determina el directorio de inicio de sus páginas Web.

```
DocumentRoot /var/www/html
```

NOTA También puede configurar Apache para operar sólo como servidor proxy, de caché o ambos. Las directivas predeterminadas de servidor proxy y caché, ya se incluyen en el archivo **httpd.conf**. La directiva **ProxyRequests** establece la actividad proxy. La caché puede configurarse con directivas como **CacheRoot** para especificar el directorio de la caché, **CacheSize** para el tamaño de la caché (la opción predeterminada es 500KB) y **CacheMaxExpire**, para definir un tiempo límite de espera en documentos no modificados.

Configuración de nivel de directorio: **.htaccess** y **<Directory>**

Uno de los aspectos más flexibles de Apache es la capacidad para configurar directorios individuales. Con la directiva **Directory**, se define un bloque de directivas aplicando sólo a un directorio particular. Esta directiva se coloca en el archivo de configuración **httpd.conf** o **access.conf**. También se utiliza **.htaccess** en un directorio particular para almacenar directivas de configuración. Luego, estas directivas se aplican sólo a ese directorio. El nombre **".htaccess"** se establece con la directiva **AccessFileName**. Puede cambiar esto, si así lo desea.

```
AccessFileName .htaccess
```

Un bloque **Directory** comienza con una directiva **<Directory nombredediruta>**, donde *nombredediruta* es el directorio que se configura. La directiva de terminación usa los mismos símbolos

<>, pero con una diagonal antes de la palabra “Directory”: </Directory>. Las directivas colocadas en este bloque sólo aplican al directorio especificado. En el siguiente ejemplo, se niega el acceso al directorio **misimagenes** sólo a solicitudes de **www.misvideos.com**.

```
<Directory /var/www/html/misimagenes>
  Order Deny,Allow
  Deny from www.misvideos.com
</Directory>
```

Con la directiva **Options**, puede habilitar ciertas características de un directorio, como el uso de vínculos simbólicos, indización automática, ejecución de secuencias de comandos CGI y negociación de contenido. La opción predeterminada es **All**, que activa todas las características excepto negociación de contenido (**Multiviews**). En el siguiente ejemplo se habilita indización automática (**Indexes**), vínculos simbólicos (**FollowSymLinks**) y negociación de contenido (**Multiviews**).

```
Options Indexes FollowSymLinks Multiviews
```

Las configuraciones hechas mediante directivas en los principales archivos de configuración o directorios de nivel más alto, son heredadas por directorios de nivel más bajo. Las directivas de un directorio particular se almacenan en archivos **.htaccess**, y puede permitir que los bloques **Directory** sustituyan tales configuraciones. Esta capacidad se controla con la directiva **AllowOverride**. Con el argumento **all**, los archivos **.htaccess** pueden sustituir cualquier configuración previa. El argumento **none** deshabilita la sustitución, al deshabilitar efectivamente el archivo **.htaccess**. Puede controlar más la sustitución de grupos específicos de directivas. **AuthConfig** permite usar directivas de autorización, **FileInfo** es para directivas de tipos, **Indexes** para directivas de indización, **Limit** para directivas de control de acceso y **Options** para la directiva de opciones. En el siguiente ejemplo, se permite acceso a todos los usuarios:

```
AllowOverride all
```

Control de acceso

Con las directivas de control de acceso, como **allow** y **deny**, se controla el acceso por usuarios remotos y hosts a su sitio Web. La directiva **allow** seguida por una lista de nombres de host restringe el acceso a sólo dichos hosts. La directiva **deny** con una lista de nombres de host niega el acceso a tales sistemas. El argumento **all** aplica la directiva a todos los host. La directiva **order** especifica en qué orden se aplicarán las directivas de control de acceso. Otras directivas de control de acceso, como **require**, establecen controles de autenticación, que requieren que los usuarios inicien sesión. Las directivas de control de acceso se utilizan de forma global para controlar el acceso a todo el sitio o se utilizan colocadas dentro de las directivas **Directory** para controlar el acceso a directivas individuales. En el siguiente ejemplo, todos los usuarios tienen acceso permitido:

```
order allow,deny
allow from all
```

Nombres de rutas URL

Ciertas directivas modifican o completan segmentos de nombre de ruta de un URL utilizado para acceder a su sitio. El segmento del nombre de ruta del URL especifica un directorio o página Web particulares de su sitio. Las directivas permiten crear alias o redirigir nombres de ruta, además de elegir una página Web predeterminada. Con la directiva **Alias**, puede permitir

que los usuarios accedan recursos ubicados en otras partes de su sistema, en otros sistemas de archivo u otros sitios Web. Un alias puede usar un URL para sitios de Internet, en lugar de un nombre de ruta para un directorio de su sistema. Con la directiva **Redirect**, redirigirá a un usuario a otro sitio.

```
Alias /mitren /home/daniel/proytren
Redirect /miscarros http://www.misautos.com/miscarros
```

Si Apache sólo se recibe un directorio para acceso, en lugar de una página Web específica, busca una página Web de índice en ese directorio y la despliega. Los nombres posibles de una página Web predeterminada se muestran en una lista de la directiva **DirectoryIndex**. El nombre más común es **index.html**, pero puede agregar otros. Aquí se muestran los nombres estándar. Si a Apache sólo se le da un directorio Web para acceso, busca y despliega la página Web **index.html** ubicada en éste.

```
DirectoryIndex index.html index.shtml index.cgi
```

Apache también permite a un usuario mantener páginas Web localizadas en un subdirectorio especial del directorio de inicio del usuario, en lugar del directorio del sitio Web principal. Al utilizar un ~ seguido por el nombre de usuario, se accede a este directorio. El nombre de este directorio se especifica con la directiva **UserDir**. El nombre predeterminado es **public_html**, como se muestra aquí. El sitio **tortuga.mipista.com/~daniel** accede al directorio **tortuga.mipista.com/home/daniel/public_html** en el host **tortuga.mipista.com**.

```
UserDir public_html
```

Si, en cambio, quiere permitir el uso de un nombre de ruta completo, entonces maneje una referencia a nombre de ruta. Por ejemplo, para el usuario **daniel**, **/usr/www** se traduce en una referencia URL de **/usr/www/daniel**, donde se ubican los archivos HTML; **/home/*www** se traduce en **/home/daniel/www**, un directorio **www** en el directorio de inicio de **daniel**.

```
UserDir /usr/www
UserDir /home/*/www
```

Como opción predeterminada, el acceso a **Userdir** es un comentario en el archivo de configuración estándar. Usualmente, existen usuarios como **root**, a los que quiere se les niegue el acceso. Con opciones para habilitar y deshabilitar, puede abrir el acceso a ciertos usuarios, mientras deshabilita el acceso a otros, como se muestra aquí:

```
UserDir disable root
UserDir disabled
UserDir enabled daniel chris julian
```

Tipos MIME

Cuando un explorador accede a páginas de un sitio Web, suele acceder a diferentes tipos de objetos, incluidos archivos HTML, archivos de sonido o imagen y secuencias de comandos. Para desplegar estos objetos correctamente, el explorador debe tener alguna indicación de qué tipo de objetos se trata. Un archivo de imagen JPEG se maneja diferente de un archivo de texto simple. El servidor proporciona este tipo de información en forma de tipos MIME. Los tipos MIME son los mismos utilizados para enviar archivos adjuntos a través de agentes de correo de Internet, como Pine. Cada tipo de objeto se asocia a un tipo MIME determinado. Si se proporciona el tipo Mime, el explorador maneja y despliega el objeto de manera correcta.

El protocolo MIME asocia cierto tipo con archivos de una extensión determinada. Por ejemplo, los archivos con extensión **.jpg** tienen el tipo MIME `image/jpeg`. La directiva **TypesConfig** almacena la ubicación del archivo **mime.types**, mostrando una lista de todos los tipos MIME y extensiones de archivo asociadas. **DefaultType** es el tipo MIME predeterminado para cualquier archivo cuyo tipo no se puede determinar. **AddType** permite modificar la lista de tipos **mime.type**, sin editar el archivo MIME.

```
TypesConfig /etc/mime.types
DefaultType text/plain
```

Otras directivas de tipo se utilizan para especificar acciones que deben tomarse en ciertos documentos. **AddEncoding** permite a los exploradores descomprimir al vuelo archivos comprimidos. **AddHandler** asocia extensiones de archivos con acciones y **AddLanguage** especifica el idioma de un documento. En el siguiente ejemplo se marcan los nombres de archivo con extensión **.gz**, como archivos **gzip** codificados y los archivos con extensión **.fr**, como archivos en idioma francés:

```
AddEncoding x-gzip gz
AddLanguage fr .fr
```

Un servidor Web despliega y ejecuta muchos tipos diferentes de archivos y programas. Sin embargo, no todos los exploradores Web pueden desplegar esos archivos. Los exploradores más antiguos son los más limitados. Algunos exploradores, como Lynx, no están diseñados para desplegar siquiera imágenes simples. Para permitir que un explorador Web despliegue una página, el servidor negocia con éste, para determinar qué tipo de archivos puede manejar. Para habilitar tal negociación, necesita habilitar la opción **Multiviews**.

```
Option multiviews
```

Archivos CGI

Los archivos de la interfaz común de puerta de enlace (CGI, Common Gateway Interface), son programas ejecutados por los exploradores Web al acceder a su sitio. Los archivos CGI suelen iniciarse con páginas Web ejecutando los programas como parte del contenido que despliegan. En general, los programas CGI se colocan en un directorio llamado **cgi-bin** y sólo pueden ejecutarse si residen en ese directorio especial. Usualmente, sólo existe un directorio **cgi-bin** por sitio Web. Es normal que las distribuciones configuren un directorio **cgi-bin** en el directorio predeterminado del servidor Web (**/var/www/cgi-bin** en Fedora). Aquí, puede colocar cualquier programa CGI ejecutable en su sitio Web. La directiva **ScriptAlias** especifica un alias para su directorio **cgi-bin**. Cualquier página Web o explorador, puede utilizar el alias para referir ese directorio.

```
ScriptAlias /cgi-bin/ /var/www/cgi-bin/
```

Indización automática de directorios

Cuando se da un URL para un directorio, en lugar de un archivo HTML, y no hay una página Web predeterminada en el directorio, Apache crea una página al vuelo y la despliega. Suele tratarse sólo de una lista de diferentes archivos del directorio. En efecto, Apache indiza los elementos del directorio. Puede establecer varias opciones para generar y desplegar ese índice. Si **FancyIndexing** está activado, los elementos de la página Web se despliegan con iconos y encabezados de columna usados para ordenar la lista.

```
Fancyindexing on
```

Autenticación

Su servidor Web también controla el acceso a directorios particulares de su sitio Web por usuario o grupo. Puede solicitar varios niveles de autenticación. El acceso puede limitarse a usuarios particulares y pedir contraseñas o expandirse a miembros permitidos de un grupo de acceso. Puede distribuir contraseñas o configurar un tipo de acceso anónimo, como se utiliza con FTP.

Para aplicar directivas de autenticación a ciertos directorios, coloque esas directivas en un bloque **Directory** o archivo **.htaccess** del directorio. Utilice la directiva **require** para determinar qué usuarios tendrán acceso al directorio. Haga una lista de usuarios particulares o grupos. La directiva **AuthName** proporciona un reino de autenticación para el usuario, el nombre utilizado para identificar el conjunto particular de recursos a los que accede este proceso de autenticación. La directiva **AuthType** especifica el tipo de autenticación, como basic o digest. Una directiva **require** necesita también **AuthType**, **AuthName** y directivas especificando las ubicaciones de archivos de autenticación de grupos o usuarios. En el siguiente ejemplo, sólo se permite a los usuarios **jorge**, **roberto** y **marco** el acceso al directorio **nuevasimagenes**:

```
<Directory /var/www/html/nuevasimagenes
    AuthType Basic
    AuthName Nuevasimagenes
    AuthUserFile /web/users
    AuthGroupFile /web/groups
    <Limit GET POST>
        require users jorge roberto marco
    </Limit>
</Directory>
```

En el siguiente ejemplo se permite el acceso de grupo para administradores al directorio CGI:

```
<Directory /var/www/html/cgi-bin
    AuthType Basic
    AuthName CGI
    AuthGroupFile /web/groups
    <Limit GET POST>
        require group admin
    </Limit>
</Directory>
```

Para configurar acceso anónimo a un directorio, coloque la directiva **Anonymous** con el usuario anónimo como argumento en el bloque **Directory** o archivo **.htaccess** del directorio. También utilice la directiva **Anonymous**, para ofrecer acceso a usuarios particulares sin necesitar contraseña.

Apache mantiene sus propios archivos de autenticación de usuario y grupo para especificar a qué usuarios y grupos se permite el acceso a ciertos directorios. Estos archivos suelen ser archivos de texto simple, como sus archivos de contraseñas y grupos de su sistema. Sin embargo, pueden volverse grandes, y llegan a hacer lentas las búsquedas de autenticación. Como opción, muchos sitios han utilizados archivos de administración de bases de datos en lugar de archivos simples. Así, los métodos de bases de datos se utilizan para acceder a los archivos, proporcionando un tiempo de respuesta más rápido. Apache tiene directivas para especificar los archivos de autenticación, dependiendo del tipo de archivo que está utilizando. Las directivas **AuthUserfile** y **AuthGroupFile** se usan para especificar la ubicación de archivos de autenticación con formato de archivo simple y estándar. Las directivas **AuthDBUserFile** y **AuthDBGroupFile** se usan para archivos de bases de

datos DB y directivas **AuthDBMGroupFile** y **AuthDBMGroupFile** se utilizan para archivos de base de datos DBMG.

Los programas **htdigest**, **htpasswd** y **dbmmanage** son herramientas ofrecidas con el paquete de software Apache para crear y mantener *archivos de autenticación de usuario*, archivos de contraseña de usuarios mostrando listas de usuarios que han accedido a directorios o recursos específicos de su sitio Web. Los programas **htdigest** y **htpasswd** manejan un archivo plano simple, de registros de autenticación de usuario, mientras **dbmmanage** utiliza un formato de administración de base de datos más complejo. Si su lista de usuario es extensa, tal vez requiera usar un archivo de base de datos para búsquedas rápidas. **htdigest** toma como argumentos archivo de autenticación, reino y nombre de usuario, creando o actualizando la entrada del usuario. **htpasswd** también emplea cifrado en la contraseña. **dbmmanage** tiene un amplio conjunto de opciones para agregar, eliminar y actualizar entradas de usuario. Se utilizan diversos formatos de base de datos para configurar tales archivos. Tres bases de datos comunes son Berkeley DB2, NDBM y GNU GBDM. **dbmmanage** busca bibliotecas de sistema de estos formatos en ese orden. Tenga cuidado para ser consistente en el uso del mismo formato para sus archivos de autenticación.

Archivos de registro

Apache mantiene registros de todas las solicitudes hechas por los usuarios a su sitio Web. Como opción predeterminada, estos registros incluyen registros empleando el formato común de registro (CLF, Common Log Format). El registro de cada solicitud toma hasta una línea compuesta por varios campos: host, revisor de identidad, usuario autenticado (para inicios de sesión), fecha, línea de solicitud remitida por el cliente, estado enviado al cliente y tamaño del objeto enviado en bytes.

Webalizer

La herramienta Webalizer genera informes a partir de registros Web. Webalizer desplegará información de uso de su sitio Web. Cuando ejecute el comando **webalizer**, los informes de uso se colocarán en el directorio **/var/www/html/usage**. Acceda la página de índice para desplegar una página con vínculos a reportes mensuales, **file:/var/www/html/usage/index.html**. La configuración de informe se especifica en el archivo **/etc/webalizer.conf**. Los resúmenes previos se almacenan en el archivo **/etc/webalizer.history**.

Personalización de registros

Mediante las directivas **LogFormat** y **CustomLog**, puede personalizar su registro para agregar más campos con niveles de detalle variables. Estas directivas usan un formato de cadena constando de especificadores de campo para determinar que registros se incluirán. Agregue los campos que quiera en cualquier orden. Un especificador de campo está integrado por un símbolo (%), seguido por un carácter de identificación. Por ejemplo, **%h** es el especificador de archivo para un host remoto, **%b** para el tamaño en bytes y **%s** para el estado. Consulte la documentación del módulo **mod_log_config** para conocer una lista completa. Debe incluir los campos entre comillas, cuyo contenido pueden ocupar más de una palabra. Las propias comillas deben marcarse con una diagonal invertida para incluirse en la cadena. En el siguiente ejemplo se muestra el formato común de registro implementado como directiva **FormatLog**:

```
FormatLog "%h %l %u %t \"%r\" %s %b"
```

Es posible calificar ciertos especificadores de campo en el formato de registro para registrar información específica. El especificador **%i** registra líneas de encabezado en solicitudes que recibidas por el servidor. La referencia a la línea de encabezado específica que habrá de registrarse,

se coloca entre corchetes, a partir de % y el especificador de campo. Por ejemplo, **User-agent** es la línea de encabezado indicando el software de explorador utilizado en la solicitud. Para registrar la información de encabezado de User-agent, se usa el especificador de conversión %{User-agent}i.

Para mantener la compatibilidad con servidores NCSA, Apache implementó originalmente las directivas **AgentLog** y **RefererLog** para registrar encabezados User-agent y Referer. Estas directivas se han remplazado desde entonces con especificadores de campo %i calificados, empleados para directivas **LogFormat** y **CustomLog**. Un encabezado Referer registra información de vínculo de clientes, al detectar quién puede tener vínculos a su sitio. El siguiente es un formato de registro compatible con NCSA:

```
"%h %l %u %t \"%r\" %s %b\"%{Referer}i\" \"%{User-agent}i\"".
```

Generación y administración de archivos de registro

En lugar de mantener un archivo de registro grande, se crean varios archivos de registro usando la directiva **CustomLog** o **TransferLog**. Esto es útil en el caso de host virtuales en que tal vez quiera mantener un archivo de registro separado para cada host. Utilice la directiva **FormatLog** para definir un formato predeterminado para registros. Luego, **TransferLog** utiliza esta opción predeterminada como su formato, cuando crea un nuevo archivo de registro. **CustomLog** combina ambas operaciones, permitiéndole crear un nuevo archivo y definir un formato para éste.

```
FormatLog "%h %l %u %t \"%r\" %s %b"
# Crea un nuevo archivo de registro llamado miproyreg al utilizar el formato
FormatLog
TransferLog miproyreg
# Crea un nuevo archivo de registro llamado misimagenesreg al utilizar su propio
formato
CustomLog misimagenesreg "%h %l %u %t \"%r\" %s %b"
```

Apache proporciona dos utilerías para procesar y administrar archivos de registro: **logresolve** resuelve direcciones IP en su archivo de registro para nombres de host; **rotatelogs** gira los archivos de registro sin desactivar el servidor. Se especifica el tiempo de rotación.

NOTA El servidor Web Apache también ofrece informes detallados sobre la actividad y configuración del servidor, permitiéndole desplegar esta información a servidores remotos. La directiva **Location** de **server-info**, desplegará detalles de configuración de su servidor Web y la directiva de estado de servidor mostrará los procesos Web. Las páginas **server-info** y **server-status** desplegarán informes, como en <http://localhost/server-info>. Use la directiva **ExtendedStatus** para habilitar informes detallados.

Hosts virtuales en Apache

Los hosts virtuales permiten que el servidor Web Apache hospede varios sitios Web. En efecto, el servidor actúa como varios servidores y cada sitio Web hospedado aparece aparte, ante los usuarios externos. Apache soporta hosts virtuales basados en dirección IP y nombre. Los host virtuales basados en dirección IP usan direcciones IP validas registradas, mientras los host virtuales basados en nombre usan direcciones de dominio plenamente calificadas. El encabezado de host del explorador que hizo la solicitud, ofrece estas direcciones de dominio. Luego, el servidor determina el host virtual correcto que habrá de utilizarse, con base en el nombre de dominio. Observe que los servidores SSL requieren host virtual IP. Visite httpd.apache.org para conocer más información.

Hosts virtuales basados en IP

En el método de host virtual basado en la dirección IP, su servidor debe tener una dirección IP diferente para cada host virtual. La dirección IP en uso, ya está configurada para hacer referencia a su sistema. Las operaciones de administración de sistemas de red configuran su máquina para soportar varias direcciones IP. Su máquina tiene varias conexiones de red físicas para cada una, o se configura una conexión particular para escuchar varias direcciones IP al mismo tiempo. En efecto, con cualquiera de las direcciones IP se accede a su sistema.

Puede configurar Apache para ejecutar un daemon separado para cada host virtual, al escuchar de manera separada cada dirección IP o tener un solo daemon en ejecución escuchando las solicitudes de todos los host virtuales. Para configurar un solo daemon para administrar todos los host virtuales, use las directivas **VirtualHost**. Para configurar un daemon separado para cada host, también utilice la directiva **Listen**.

Hosts virtuales basados en nombre

Con el host virtual basado en IP, está limitado al número de direcciones IP que su sistema soporta. Con el host virtual basado en nombre, se soporta cualquier cantidad de hosts virtuales sin usar direcciones IP adicionales. Con sólo una dirección IP para su máquina, todavía se da soporte a un número ilimitado de hosts virtuales. Esta capacidad es posible por el protocolo HTTP/1.1, permitiendo a un servidor identificar el nombre con que se accede a él. Este método requiere que el cliente, el usuario remoto, use un explorador soportando el protocolo HTTP/1.1, como hacen los exploradores actuales (aunque los antiguos tal vez no). Un explorador usando este protocolo envía un encabezado de host que especifica el host particular para usarse en una máquina.

Si su sistema sólo tiene una dirección IP, la implementación de hosts virtuales evita el acceso a su servidor principal con esa dirección. Ya no puede utilizar su servidor principal como servidor Web directamente; sólo se utilizará indirectamente para administrar su host virtual. Sin embargo, puede configurar un host virtual para administrar las páginas Web de su servidor principal. Entonces podrá usar su servidor principal para soportar un conjunto de hosts virtuales que funcionarían como sitio Web, en lugar del servidor principal operando como sitio directamente. Si su máquina tiene una o más direcciones IP, use una para el servidor principal y otra para hosts virtuales. Incluso puede combinar hosts virtuales basados en IP y nombre. También puede usar direcciones IP separadas para soportar conjuntos diferentes de hosts virtuales. También hacer que varias direcciones de dominio accedan al mismo host virtual. Para ello, coloque la directiva **ServerAlias** mostrando una lista de nombres de dominio en el bloque **VirtualHost** seleccionado.

```
ServerAlias www.misimagenes.com www.grandesfotos.com
```

Las solicitudes enviadas a la dirección IP usadas para su host virtual deben coincidir con uno de los nombres de dominio virtuales configurados. Para detectar solicitudes sin coincidencia con uno de estos hosts, configure un host virtual predeterminado con `_default_*`. Este host virtual manejará entonces solicitudes no coincidentes.

```
<VirtualHost _default_*>
```

Host virtual dinámico

Si ha implementado muchos hosts virtuales en su servidor con la misma configuración, se usa una técnica llamada *host virtual dinámico* para que éstos se generen de forma dinámica. El código para implementar hosts virtuales se vuelve mucho más pequeño y, como resultado, su servidor accede a

éste más rápido. Para añadir más hosts virtuales sólo necesita crear directorios apropiados y agregar entradas para éstos en el servidor DNS.

Para que el host virtual dinámico funcione, el servidor utiliza los comandos del módulo `mod_vhost_alias` (con soporte en la versión 1.3.6 y posteriores de Apache) para reescribir el nombre del servidor y directorio raíz de los documentos con los del servidor virtual apropiado (en el caso de versiones anteriores a 1.3.6 de Apache, se utiliza el módulo `mod_rewrite_module`). Los hosts virtuales dinámicos se basan en nombre o IP. En cualquier caso, debe configurar la directiva **UserCanonicalName** de tal forma que permita al servidor utilizar el nombre de host virtual, en vez del propio nombre del servidor. En el caso de hosts basados en nombre, sólo desactive **UseCanonicalName**. Esto permite a su servidor obtener el nombre de host del encabezado de la solicitud de usuario. En el caso de hosts basados en IP, debe establecer la directiva **UseCanonicalName** en DNS. Esto permite que el servidor busque el host en el servidor DNS.

```
UserCanonicalName off
UseCanonicalName DNS
```

Después debe permitir que el servidor localice el directorio raíz de documentos y directorios bin CGI de sus hosts virtuales. Se utiliza la directiva **VirtualDocumentRoot** para especificar la plantilla de directorios de hosts virtuales. Por ejemplo, si coloca diferentes directorios de host en el directorio `/var/www/hosts`, entonces se establece la directiva **VirtualDocumentRoot** de manera correspondiente.

```
VirtualDocumentRoot /var/www/host/%0/html
```

El `%0` se remplazará con el nombre del host virtual, cuando acceda al host virtual. Es importante crear el directorio del host dinámico mediante el nombre del host. Por ejemplo, para un host virtual dinámico llamado **www.migolf.org**, primero debe crear un directorio llamado `/var/www/hostswww.migolf.org` y después los subdirectorios para el directorio raíz de documentos y programas CGI, como en `/var/www/hosts/www.migolf.org/html`. En el caso del directorio CGI, use la directiva **VirtualScriptAlias** para especificar el subdirectorio CGI que utiliza.

```
VirtualScriptAlias /var/www/hosts/%0/cgi-bin
```

Un ejemplo simple de host virtual dinámico basado en nombre sería:

```
UserCanonicalName Off
VirtualDocumentRoot /var/www/hosts/%0/html
VirtualScriptAlias /var/www/hosts/%0/cgi-bin
```

Una solicitud para **www.migolf.com/html/mipagina** se evalúa como

```
/var/www/hosts/www.migolf.com/html/mipagina
```

Aquí se muestra un ejemplo simple de host virtual dinámico:

```
UserCanonicalName Off

NameVirtualHost 192.168.1.5

<VirtualHost 192.168.1.5>
  ServerName www.migolf.com
```

```
ServerAdmin webmaster@correo.migolf.com
VirtualDocumentRoot /var/www/hosts/%0/html
VirtualScriptAlias /var/www/hosts/%0/cgi-bin
...
</VirtualHost>
```

Para implementar un host virtual dinámico basado en IP, configure **UseCanonicalName**, en DNS en vez de Off.

```
UserCanonicalName DNS
VirtualDocumentRoot /var/www/hosts/%0/html
VirtualScriptAlias /var/www/hosts/%0/cgi-bin
```

Cadenas interpoladas

El módulo `mod_vhots_alias` soporta varias cadenas interpoladas, cada una con un símbolo `%` al principio, seguido por un número. Como ha visto, `%0` hace referencia a la dirección Web completa; `%1` sólo al primer segmento; `%2` al segundo; `%-1` a la última parte y `%2+` de la segunda parte en adelante. Por ejemplo, para utilizar sólo la segunda parte de la dirección Web del nombre de directorio, utilizaría las siguientes directivas:

```
VirtualDocumentRoot /var/www/hosts/%2/html
VirtualScriptAlias /var/www/hosts/%2/cgi-bin
```

En este caso, una solicitud hecha para **www.migolf.com/html/mipagina** sólo usa la segunda parte de la dirección Web. Sería “migolf” en `www.migolf.com`, y se evaluaría como

```
/var/www/hosts/migolf/html/mipagina
```

Si, en cambio, utiliza `%2+`, como en `/var/www/hosts/%2/html`, la solicitud para **www.migolf.com/html/mipagina** se evaluaría como

```
/var/www/hosts/migolf.com/html/mipagina
```

El mismo método funciona para direcciones IP, donde `%1` hace referencia al primer segmento de la dirección IP, `%2` hace referencia al segundo, etc.

Registros en el caso de host virtuales

Un inconveniente del host virtual dinámico es que sólo se configura un registro para todos sus hosts. Sin embargo, puede crear su propio programa shell, para simplemente dejar fuera las entradas de diferentes hosts en ese registro.

```
LogFormat "%V %h %l %u %t \"%r\" %s %b" vcommon
CustomLog logs/acess_log vcommon
```

Direccionamiento IP

Al implementar hosts virtuales dinámicos en forma estándar, como se mostró anteriormente, se hará más lento el proceso, porque su servidor deberá realizar una búsqueda DNS para descubrir el nombre de su servidor mediante su dirección IP. Evitará este paso si usa la dirección IP para su directorio del host virtual. Así que, para el host virtual IP 192.168.1.6, crearía un directorio `/var/www/hosts/192.168.1.6`, con un subdirectorio `html` para la raíz de documentos del host. Debe usar las directivas **VirtualDocumentRootIP** y **VirtualScriptAliasIP** para direcciones IP como nombres

de directorio. Ahora la dirección IP se asocia directamente al nombre del directorio raíz de los documentos, por lo que ya no es necesaria una búsqueda DNS. También, asegúrese de incluir la dirección IP en su registro, **%A**.

```
UserCanonicalName DNS
LogFormat "%A %h %l %u %t \"%r\" %s %b" vcommon
CustomLog logs/acces_log vcommon
VirtualDocumentRoot /var/www/hosts/%0/html
VirtualScriptAlias /var/www/hosts/%0/cgi-bin
```

Includes desde el servidor

Includes del lado del servidor (SSI, Server-Side Includes) está diseñado para proporcionar un control mucho más definido del contenido de su sitio Web (es decir, las propias páginas Web). Se trata de directivas de Apache colocadas en páginas Web particulares, como parte del código HTML de la página. Su servidor Web Apache se configura para buscar directivas SSI en páginas Web particulares y ejecutarlas. Primero, debe usar la directiva **Options** con la opción **include** para permitir directivas SSI.

Options Includes

Necesita instruir al servidor para analizar sintácticamente páginas Web particulares. La forma más sencilla de habilitar el análisis consiste en instruir a Apache para analizar archivos HTML con extensiones específicas. En general, la extensión **.shtml** se usa para páginas Web que tienen directorios SSI. En realidad, en archivos de configuración predeterminados de Apache, se encuentra la siguiente entrada para habilitar el análisis para directivas SSI en archivos HTML. Aquí, la directiva **AddType** agrega **.shtml** como tipo de archivo HTML y la directiva **AddHandler** especifica qué archivos **.shtml** se analizarán (server-parsed):

```
# Para utilizar archivos HTML server-parsed
AddType text/html .shtml
AddHandler server-parsed .shtml
```

En lugar de crear un tipo de archivo separado, se utiliza la directiva **XBitHack**, para que Apache analice cualquier archivo ejecutable en busca de directivas SSI. En otras palabras, cualquier archivo con permiso de ejecución se analizará en busca de directivas SSI.

Las directivas SSI operan de manera muy similar a las instrucciones en un lenguaje de programación. Se definen variables, se crean bucles y se utiliza una prueba para seleccionar directivas alternas. La directiva SSI está integrada por un elemento seguido por atributos que se asignan a valores. Aquí se muestra la sintaxis de una directiva SSI:

```
<!--#elemento atributo=valor ... -->
```

Considere que un elemento funciona de manera muy parecida a un comando en un lenguaje de programación; los atributos funcionan como argumentos. Por ejemplo, para asignar la variable, utilice el elemento **set**, con la asignación de variable como atributo. La directiva **if** despliega cualquier texto que le sigue en una página Web determinada. La directiva **if** toma **expr** como su atributo, mismo que se asigna a la expresión que habrá de probarse. La prueba puede comparar dos cadenas al usar comandos de comparación estándar como **<=**, **!=** o **=**. Las variables utilizadas en la prueba se evalúan con el operador **\$**.

```
<!--#set mivar="Adios" -->
<!--#if expr="$mivar = Hola" -->
```

Otros elementos útiles de SSI son **exec**, ejecutando programas CGI o comandos shell, que leen el contenido de un archivo en la página Web y también ejecutan archivos CGI. El elemento **echo** despliega valores como fecha, nombre del documento y URL de la página. Con el elemento **config**, se configuran ciertos valores, como fecha y tamaño de archivo.

PHP

PHP (PHP: Hypertext Preprocessor) es un lenguaje de secuencias de comandos, diseñado para utilizarse en páginas Web. Las páginas con PHP habilitado permiten crear páginas Web dinámicas que realizan tareas, en lugar de sólo desplegar datos. PHP es un proyecto oficial de Apache Software Foundation. Encontrará más acerca de PHP en **php.net**.

A diferencia de los programas CGI, que se ejecutan de manera separada de una página Web, los comandos PHP están incrustados en etiquetas dentro de la propia página, de manera muy similar a los comandos SSI. El soporte ofrecido por PHP para interpretar y ejecutar estos comandos lo proporciona directamente el servidor Web. Este soporte incrustado se habilita en Apache con el módulo mod-php (archivo de configuración **/etc/httpd/conf.d/php.conf**). Esto es, en vez de construir programas de forma separada para ser invocados y ejecutados fuera del servidor Web, en PHP estos comandos se incrustan en una página Web y los ejecuta el servidor Web. Éste mantiene control completo en todo momento, siempre que las tareas se realicen. Sin embargo, es posible implementar PHP en modo CGI, donde las páginas PHP se construyan como programas separados para ser invocados desde una página Web, de manera similar a los programas CGP basados en Perl.

PHP tiene capacidades de programación poderosas y flexibles, al mismo nivel de C y Perl. Como en esos lenguajes, se crean estructuras de control como instrucciones y bucles if. Además, PHP tiene capacidades ideales y específicas para tareas de página Web. PHP interactúa directamente con bases de datos como Oracle, MySQL e DB2 de IBM. Interactúa también, de manera sencilla, con todos los protocolos estándar, como IMAP, LDAP, HTTP y POP3. Incluso tiene capacidades de procesamiento de texto, como interpretar expresiones regulares y desplegar documentos XML. También existen extensiones para búsqueda, herramientas de compresión como gzip y traducciones de lenguaje. PHP soporta una amplia colección de operaciones posibles. Revise su página Web, además de los manuales y tutoriales en línea, para conocer una lista completa.

Herramienta de configuración de Apache

La herramienta de configuración de Apache se abre con una ventana desplegando paneles para Principal, Hosts virtuales, Servidor y Ajuste del rendimiento. En cada uno de estos verá botones para abrir cuadros de diálogo donde se insertan configuraciones predeterminadas. También podrá insertar configuraciones para elementos particulares, como host virtuales y directorios. Por ejemplo, en el panel Host virtuales se insertan configuraciones predeterminadas para todos los host virtuales, además de agregar y editar host virtuales particulares. Haga clic en el botón Ayuda para desplegar un manual de referencia basado en página Web mostrando detalles de cómo utilizar cada panel.

- En el panel Principal, escriba su dirección de Servidor Web, la dirección de correo electrónico del Webmaster y puertos del servidor Web en que estará escuchando.

- En el panel Hosts virtuales, asegúrese de seleccionar Default Virtual Host y haga clic en Modificar, para establecer configuraciones predeterminadas de opciones de servidor, búsquedas de página, soporte SSL, archivos de registro, soporte de entorno CGI y directorios (Performance). Para agregar host virtuales, haga clic en Añadir, para abrir una ventana donde se insertará la información de host como nombre de host virtual y dirección IP. Seleccione paneles de configuración diferentes para el host virtual, como log files y directory controls.
- En el panel Servidor, se establecen configuraciones administrativas como archivos de ID de usuario e ID de proceso del servidor, junto con usuario y grupo.
- El panel Ajuste del rendimiento permite configurar diferentes límites de uso, como número máximo de solicitudes y número de solicitudes por conexión.

Cuando la herramienta de configuración de Apache guarda sus configuraciones, sobrescribirá el archivo de configuración de Apache, `/etc/httpd/conf/httpd.conf`. Es recomendable que primero haga un respaldo de su archivo `httpd.conf`, en caso de restaurar las configuraciones originales creadas por su distribución. Si ya ha editado este archivo manualmente, recibirá una advertencia y la herramienta de configuración Apache hará un respaldo en `/etc/httpd/conf/httpd.conf.bak`.

Seguridad del servidor Web: SSL

La seguridad del servidor Web se relaciona con dos tareas diferentes: proteger su servidor Web de acceso no autorizado y proporcionar seguridad para transacciones llevadas a cabo entre un cliente de explorador Web y su servidor Web. Para proteger su servidor del acceso no autorizado, utilice un servidor proxy como Squid. Squid es un servidor proxy GNU usado a menudo con Apache en sistemas Linux. Apache por sí solo tiene varios módulos que proporcionan capacidades de seguridad. Entre éstos se incluyen `mod_access`, para controles obligatorios; `mod_auth`, `mod_auth_db`, `mod_auth_digest` y `mod_auth_dbm`, soportando autenticación y `mod_auth_anon`, para registros anónimos parecidos a FTP (consulte las secciones anteriores sobre control de acceso y autenticación).

Para asegurar las transmisiones, necesita realizar tres tareas. Debe verificar las identidades, revisar la integridad de los datos y asegurar la privacidad de la transmisión. Para verificar las entidades de los host participando en la transmisión, realice procedimientos de autenticación. Para revisar la integridad de datos, agregue firmas digitales conteniendo un valor digest para los datos. El valor digest representa únicamente los datos. Por último, para asegurar la privacidad de la transmisión, cifrela. Las transacciones entre un explorador y su servidor se cifran, el explorador y servidor por sí solos son capaces de descifrar transmisiones. El protocolo usado más a menudo para implementar transmisiones seguras con los servidores Web Apache es el protocolo de capa de conectores seguros (SSL, Secure Sockets Layer), desarrollado originalmente por Netscape para transacciones seguras en Web.

Como la shell segura (SSH, Secure SHell) y la salvaguarda de privacidad de GNU, SSL utiliza una forma de cifrado para autenticación de claves pública y privada. Los datos se cifran con la clave pública, pero sólo se descifran con la clave privada. Una vez que los datos se autentifiquen, se utiliza un cifrador acordado para cifrarlos. El cifrado de firmas digitales y el valor MD5 digest para datos, aseguran la integridad. La autenticación se lleva a cabo con el uso de certificados de autoridad. Los certificados identifican diferentes entidades en una transmisión segura, al verificar que son quienes dicen ser. Un servidor Web tendrá un certificado verificando su identidad, al verificar que es el servidor que dice ser. El explorador en contacto con el servidor, también tendrá un certificado de identidad. A su vez, una autoridad de certificación firma ambos certificados,

verificando su validez. Una autoridad de certificación es una entidad independiente en la que confían ambos.

Un certificado contiene la clave pública dada al servidor o explorador particular, junto con la firma digital del certificado de autoridad e información de identidad como el nombre del usuario o compañía ejecutada por el servidor o explorador. La efectividad de un certificado depende directamente de la confiabilidad de la autoridad de certificación que lo expide. Para ejecutar un servidor Web seguro en Internet, debe obtener un certificado de autoridad de certificación, como VeriSign. Un vendedor comercial como Stronghold hace esto por usted. Muchas compañías establecidas ya mantienen sus propias autoridades de certificación, asegurando transmisiones en sus redes de la compañía. Una sesión SSL se configura al utilizar una secuencia de saludo en que servidor y explorador se autentican intercambiando certificados, se acuerda un cifrador para codificar las transmisiones y se elige el tipo de revisión de integridad digest. También existe una opción para el tipo de cifrado de clave pública utilizada para la autenticación, ya sea RSA o DSA. Para cada sesión, se configura una clave de sesión única para que la utilicen explorador y servidor.

Una versión de fuente abierta gratuita de SSL llamada OpenSSL está disponible para su uso con Apache (visite openssl.org). Está basada en SSLeay, de Eric A. Young y Tim J. Hudson. Sin embargo, las restricciones del gobierno de Estados Unidos previenen que el servidor Web Apache se distribuya libremente con capacidades SSL integradas. Debe obtener SSL por separado y actualizar su servidor Apache para incorporar esta capacidad.

El gobierno de Estados Unidos mantiene restricciones de exportación en tecnología de cifrado superior a 40 bits. Sin embargo, SSL da soporte a gran número de cifradores usando claves de 168, 128 y 40 bits (128 se considera seguro, así que por comparación, la versión de exportación de 40 bits ya no es útil). Esto significa que si Apache incluye SSL, no se puede distribuir fuera de Estados Unidos. Sin embargo, fuera de ese país existen proyectos distribuyendo SSL para Apache al utilizar OpenSSL. Estos suelen ser gratuitos para uso no comercial en Estados Unidos, aunque aplican las restricciones de exportación. El proyecto Apache-SSL distribuye de manera libre Apache con SSL integrado, Apache+ssl. Puede descargarlo de su sitio Web en Apache-ssl.org (aunque existen restricciones para la exportación de la tecnología de cifrado, no existe una para importación). Además, el proyecto mod-ssl proporciona un módulo SSL con parches usados para actualizar su servidor Web Apache, con el fin de incorporar SSL (modssl.org). mod_ssl es gratuito para uso comercial y no comercial bajo la licencia Apache-style (archivo de configuración `/etc/httpd/conf.d/ssl.conf`).

La implementación mod_ssl de SSL ofrece un acceso alternativo a su servidor Web mediante un puerto diferente (443) y un protocolo diferente, https. En efecto, tiene un servidor SSL y una versión no segura. Para acceder a la versión SSL segura, use el protocolo https, en vez de http, para la dirección URL del servidor Web. Por ejemplo, para acceder a la versión SSL para el servidor Web en ejecución en mipista.com, utilice el protocolo https en su URL, como se muestra aquí:

<https://www.mipista.com>

Se configura mod_ssl a través de varias directivas de configuración en el archivo de configuración de Apache, **smb.conf**. El archivo de configuración predeterminado instalado con Apache, contiene una sección de directivas SSL junto con componentes detallados. Revise la documentación en línea para mod_ssl en modssl.org para conocer una lista de referencia detallada de todas las directivas. Están disponibles directivas globales, basadas en servidor y en directorio.

En el archivo **smb.conf**, la inclusión de directivas SSL se controla con bloques `IfDefine`, habilitados por la marca `HAVE_SSL`. Por ejemplo, el siguiente código descargará el módulo SSL:


```
<IfDefine HAVE_SSL>
LoadModule ssl_module      modules/libssl.so
</IfDefine>
```

La versión SSL de su servidor Web Apache se configura en el archivo **smb.conf** como host virtual. Las directivas SSL se habilitan con un bloque `IfDefine`, mediante una marca `HAVE_SSL`. Varias directivas predeterminadas se implementan, como la ubicación de directorios clave SSL y el puerto (443), en que escuchará esa versión SSL del servidor. Otras están convertidas en comentarios. Habilítelas al eliminar el símbolo `#` encontrado al principio, configurando sus propias opciones. Aquí se muestran varias de las directivas:

```
<IfDefine HAVE_SSL>
## SSL Virtual Host Context

# Server Certificate:
SSLCertificateFile (etc/httpd/conf/ssl.crt/server.crt

# Server Private Key:
SSLCertificateKeyFile /etc/httpd/conf/ssl.key/server.key

# Certificate Authority (CA):
# SSLCACertificatePath /etc/httpd/conf/ssl.crt
# SSLCACertificateFile /etc/httpd/conf/ssl.crt/ca-bundle.crt
```

En el directorio **/etc/httpd/conf**, `mod_ssl` configurará varios directorios SSL, que contienen autenticación SSL y claves de cifrado y datos. El directorio **ssl.crt** contendrá certificados para el servidor. El directorio **ssl.key** almacenará las claves pública y privada usadas en el cifrado de actualización. Las listas de revocación para negar certificados expirados se almacena en **ssl.crl**. El directorio **ssl.csr** almacena las solicitudes de firma de certificados, usadas para pedir un certificado oficial a una autoridad de certificación. **ssl.prm** almacena archivos de parámetros, empleados por el método de cifrado de clave DSA. Revise los archivos **readme** de cada directorio para conocer más detalles acerca de los archivos SSL que contienen.

La instalación de `mod_ssl` proporcionará un certificado de demostración llamado **snakeoil**, que usado para probar una configuración SSL. Cuando tenga un certificado oficial, instálelo con el comando **make certificate** en el directorio **ssl.crt**. Esto sobrescribirá el archivo de certificado de servidor **server.crt**.

Servidores Proxy

Los servidores proxy operan como intermediario entre una red local y servicios disponibles en una red más grande como Internet. El servidor proxy maneja solicitudes de clientes locales para servicios Web, acelerando las transacciones, además de controlar el acceso. Los servidores proxy mantienen copias actuales de páginas Web a las que se accede con frecuencia, acelerando tiempos de acceso Web al eliminar la necesidad de acceder constantemente al sitio original. También realizan funciones de seguridad, protegiendo los servidores de acceso no autorizado. **Squid** es un servidor proxy-de caché gratuito y fuente abierta para clientes Web, diseñado para acelerar el acceso a Internet y proporcionar controles de seguridad para servidores Web. Implementa una copia del servicio proxy-de caché para clientes Web, para capturar páginas Web mientras los usuarios hacen solicitudes. Las copias de páginas Web a las que acceden los usuarios se mantienen en la caché de Squid y, a medida que se hacen las solicitudes, Squid revisa si tiene una copia actual. Si la tiene, regresa la copia de su caché en lugar de consultar el sitio original. Si no la tiene, recuperará una del sitio original. El reemplazo periódico de algoritmos reemplaza los objetos antiguos en la caché. De esta forma, los exploradores Web utilizan la caché de Squid local, como un servidor proxy de HTTP. Actualmente, Squid maneja páginas Web soportando protocolos HTTP, FTP y SSL (Squid no puede usarse con clientes FTP), cada uno con un puerto predeterminado asociado (consulte la tabla 24-1). También soporta los protocolos caché de Internet (ICP, Internet Cache Protocol), caché de hipertexto (HTCP, HyperText Caching Protocol), caché de Web y administración de red simple (SNMP, Simple Network Management Protocol), para ofrecer información de estado.

Encontrará más acerca de Squid en squid-cache.org. Para conocer información detallada, revise las preguntas más frecuentes de Squid y el manual de usuario ubicado en su sitio Web. Las preguntas más frecuentes también se instalan en su `/usr/share/doc` bajo el directorio **squid**.

Como proxy, Squid hace más que sólo incluir en caché objetos Web. Opera como intermediario entre exploradores Web (clientes) y los servidores que acceden. En lugar de que las conexiones se hagan directamente al servidor, un cliente se conecta al servidor proxy. Luego, el proxy retransmite solicitudes al servidor. Esto es útil en situaciones que un servidor Web se coloca tras un servidor de firewall, para protegerlo del acceso externo. El proxy queda accesible en la firewall, después transfiere solicitudes y respuestas entre cliente y servidor Web. El diseño a menudo se usa para permitir a servidores Web operar en redes locales protegidas, mientras aún se tiene acceso desde Internet. También se utiliza el proxy Squid, para acceder Web a Internet en host locales. En lugar de utilizar una puerta de enlace proporcionando acceso a Internet, los hosts locales utilizan un proxy para permitirles sólo acceso Web (consulte el capítulo 5). También puede combinar los dos, permitiendo el acceso a la

Protocolo	Descripción y puerto
HTTP	Páginas Web, puerto 3128
FTP	Transferencias FTP a través de sitios Web, puerto 3128
ICP	Protocolo de caché de Internet, puerto 3130
HTCP	Protocolo de caché de hipertexto, puerto 4827
CARP	Protocolo de caché de enrutamiento de matrices
SNMP	Protocolo simple de administración de red, puerto 3401
SSL	Capa de conector seguro

TABLA 24-1 Protocolos permitidos por Squid

puerta de enlace, pero empleando el servidor proxy para ofrecer más control para acceso Web. Además, la capacidad de uso de la caché de Squid ofrece a los host locales acceso Web más rápido.

Técnicamente, podría usar un servidor proxy, sólo para manejar tráfico entre un servidor Web y clientes que quieren comunicarse con él, sin guardar en la caché. Squid combina ambas capacidades como un servidor proxy-de caché.

Squid también ofrece opciones de seguridad que ejercen control sobre hosts accediendo a su servidor Web. Se niega el acceso a ciertos hosts y permite el acceso a otros. Squid también soporta el uso de protocolos de cifrado como SSL (consulte el capítulo 23). Las comunicaciones cifradas se entunelán (pasan sin leerse) a través del servidor Squid, directamente al servidor Web.

Squid tiene soporte y se distribuye bajo licencia GNU Public License, del National Laboratory for Applied Network Research (NLNLR), de la Universidad de California, en San Diego. El trabajo se basa en el Harvest Project, para crear un sistema de indización Web, incluyendo un daemon de caché de alto rendimiento llamado **cached**. Se obtienen versiones actuales de código fuente y documentación, en línea de la página de inicio de Squid en **squid-cache.org**. El paquete de software Squid consta de servidor Squid, secuencias de comandos de soporte para servicios como LDAP y HTTP, y una secuencia de comandos de administrador de caché, llamada **cachemgr.cgi**. **cachemgr.cgi** permite ver estadísticas del servidor Squid conforme se ejecuta. Se configura el servidor Squid para iniciar automáticamente mediante **chkconfig**, **services-admin** de GNOME, **rcconf** de Debian o **sysv-rc-conf**.

Configuración de exploradores de cliente

Squid soporta cachés de proxy estándar y cachés transparentes. Con una caché de proxy estándar, los usuarios necesitarán configurar sus exploradores Web para acceder específicamente al servidor Squid. Una caché transparente, por otra parte, no requiere de los usuarios configuren el explorador. La caché es transparente para acceder como si fuera un sitio Web normal. Las caches transparentes se implementan con IPtables usando filtrado de red para interceptar solicitudes y dirigir las a la caché de proxy (consulte el capítulo 20).

Con una caché de proxy estándar, los usuarios necesitan especificar su servidor proxy en la configuración de su explorador Web. Para esto necesitan la dirección IP del host, ejecutando el servidor proxy de Squid, además del puerto que usa. Los proxies suelen emplear el puerto 3128. Para configurar el uso de un servidor proxy ejecutado en la red de ejemplo local, descrita en el capítulo 5, inserte lo siguiente. (El servidor proxy se ejecuta en **tortuga.mipista.com** (192.168.0.1) y utiliza el puerto 3128):

```
192.168.0.1 3128
```

En Firefox, Mozilla y Netscape, el usuario en la red local de ejemplo, selecciona primero el panel Proxy, ubicada en Opciones, en el menú Herramientas. Luego, en el panel Ver, de la configuración manual de proxy, se ingresa la información previa. El usuario verá entradas para FTP, HTTP y proxys seguros. En el caso de acceso Web estándar, inserte la dirección IP en los cuadros FTP y Web. Para sus cuadros de puerto, escriba 3128.

En el caso de GNOME, seleccione Proxy de red, en el menú o la ventana Preferencias y el caso de Konqueror, de KDE Desktop, seleccione el panel Proxys, en la ventana del menú Preferencias | Navegador Web. Aquí, escriba la dirección de servidor proxy y números de puerto. Si su host local utiliza Internet Explorer (como hace un sistema Windows), configure las entradas de proxy en las opciones de Configuración, de la red de área local, accesible en la ventana Opciones de Internet.

En sistemas Linux y Unix, los hosts locales establecen las variables shell `http_proxy` y `ftp_proxy` para configurar el acceso por parte de exploradores Web soportados por Linux, como Lynx. Coloque estas definiciones en su archivo `.bash_profile` o `/etc/profile`, para definir las automáticamente siempre que inicie sesión.

```
http_proxy=192.168.0.1:3128
ftp_proxy=192.168.0.1:3128
export http_proxy ftp_proxy
```

De manera alterna, se utiliza el URL del proxy.

```
http_proxy=http://tortuga.mipista.com:3128
```

Para el explorador Elinks, puede especificar un proxy en su archivo de configuración, `/etc/elinks.conf`. Establezca las opciones de host de proxy FTP y Web, como en:

```
protocol.http.proxy.host  tortuga.mipista.com:3128
protocol.ftp.proxy.host   tortuga.mipista.com:3128
```

Antes de que un cliente en un host local use el servidor proxy, debe otorgar permisos de acceso a éste en el archivo `squid.conf` del servidor, descrito en la sección “Seguridad”, en páginas posteriores. Es posible ofrecer acceso fácilmente a una red completa. En el caso de la red de ejemplo, empleada aquí, debería colocar las siguientes entradas en el archivo `squid.conf`. Estas se explican en detalle en las siguientes secciones.

```
acl milan src 192.168.0.0/255.255.255.0
http_access allow milan
```

SUGERENCIA Los clientes Web necesitando acceso a su servidor Squid, como una caché de proxy estándar, necesitarán conocer la dirección del servidor y el puerto para servicios HTTP de Squid, que es 3128 como opción predeterminada.

El archivo squid.conf

El archivo de configuración de Squid es `squid.conf` y ubicado en el directorio `/etc/squid`. En el archivo `/etc/squid/squid.conf` se establecen opciones generales, como los puertos usados, opciones de seguridad para controlar el acceso al servidor y opciones de caché para configurar las operaciones de caché. Se maneja una versión de respaldo llamada `/etc/squid/squid.conf.default` para restaurar sus opciones predeterminadas originales. La versión predeterminada de `squid.conf`,

ofrecida por el software de Squid incluyendo explicaciones detalladas de todas las entradas estándar, junto con entradas predeterminadas comentadas. Las entradas constan de etiquetas especificando diferentes atributos. Por ejemplo, `maximum_object_size` y `maximum_object` establecen límites en objetos transferidos.

```
maximum_objetc_size 4096 KB
```

Como proxy, Squid manejará ciertos puertos para servicios específicos, como 3128 para servicios HTTP en exploradores Web. Los números de puerto predeterminados ya están configurados para Squid. Si necesita otros puertos, debe configurarlos en el archivo `/etc/squid/squid.conf`. La siguiente entrada muestra cómo se establece el puerto de explorador Web:

```
http_port 3128
```

NOTA *Squid usa el protocolo simple de administración de red (SNMP, Simple Network Management Protocol), para ofrecer información de estado y estadísticas a agentes SNMP administrando su red. Se controla SNMP con las configuraciones `snmp acces` y `port` del archivo `squid.conf`.*

Seguridad

Squid utiliza su función como intermediario entre clientes Web y un servidor Web para implementar controles de acceso, determinando quién accede el servidor Web y cómo accede. Squid hace esto al revisar la lista de control de acceso (ACL, Access Control List) de hosts y dominios que tienen controles colocados en éstas. Cuando encuentra un cliente Web de uno de los hosts intentando conectarse al servidor Web, ejecuta el control. Squid permite varios controles con los que niega o permite al cliente Web del host remoto el acceso al servidor Web (consulte la tabla 24-2). En efecto, Squid configura una firewall sólo para el servidor Web.

El primer paso para configurar la seguridad de Squid, consiste en crear ACL. Son listas de hosts y dominios para las que quiere configurar controles. Se definen las ACL mediante el comando `acl`, creando una etiqueta para el sistema en que configura los controles. Luego puede usar comandos como `http_access`, para definir estos controles. Defina un sistema o grupo de sistemas, usando varias opciones de `acl`, como dirección IP de origen, nombre de dominio e incluso hora y fecha. Por ejemplo, la opción `src` se utiliza para definir un sistema o grupo de sistemas con ciertas direcciones fuente. Para definir una entrada `milan acl`, para sistemas en un red local con las direcciones 192.168.0.0 a 192.168.0.255, utilice la siguiente definición ACL:

```
acl milan src 192.168.0.0/255.255.255.0
```

Una vez definido, puede usar la definición ACL en la opción Squid, para especificar un control que quiera colocar en tales sistemas. Por ejemplo, para permitir el acceso por parte del grupo milan, de sistemas locales a la Web mediante un proxy, use una opción `http_access` con la acción `allow`, para especificar `milan` como la definición `acl` que habrá de utilizarse, mostrada aquí:

```
http_access allow milan
```

Al definir varias ACL y usarlas en opciones de Squid, se personaliza el sitio Web con el tipo de seguridad que quiere. En el siguiente ejemplo, sólo se permite el acceso a Web a través del proxy al grupo `milan` de sistemas locales, negando el acceso a todos los demás.

Opción	Descripción
src <i>direc-ip/mascdere</i>	Dirección IP de cliente
src <i>direc1-direc2/mascdere</i>	Rango de direcciones
dst <i>dirección-ip/nmascdere</i>	Dirección IP de destino
myip <i>dirección-ip/nmascdere</i>	Dirección IP de conector local
srcdomain <i>dominio</i>	Búsqueda en reversa, cliente IPP
dstdomain <i>dominio</i>	Servidor de destino de URL; para dstdomain y dstdom_regex , se utiliza una búsqueda en reversa si se utiliza un URL basado en IP
srcdom_regex [-i] <i>expresión</i>	Expresión regular que busca coincidencias con el nombre del cliente
dstdom_regex [-i] <i>expresión</i>	Expresión regular que busca coincidencias con el destino
time [<i>día-abreviaturas</i>] [<i>h1:m1-h2:m2</i>]	Hora especificada por día, hora y minutos. Las abreviaturas de día son: S = Domingo, M = Lunes, T = Martes, W = Miércoles, H = Jueves, F = Viernes, A = Sábado
url_regex [-i] <i>expresión</i>	Expresión regular que busca coincidencias con el URL completo
urlpath_regex [-i] <i>expresión</i>	Expresión regular que busca coincidencias con la ruta URL
port <i>puertos</i>	Un puerto específico o rango de puertos
proto <i>protocolo</i>	Un protocolo específico, como HTTP o FTP
method <i>método</i>	Métodos específicos, como GET y POST
browser [-i] <i>regexp</i>	Coincidencia de patrón en encabezados usuario-agente
ident <i>username</i>	Coincidencia de cadena en la salida ident
src_as <i>número</i>	Utilizada para enrutamiento de solicitudes a cachés específicas
dst_as <i>número</i>	Utilizada para enrutamiento de solicitudes a cachés específicas
proxy_auth <i>nombredusuario</i>	Lista de nombres de usuario válidos
snmp_community <i>cadena</i>	Una cadena comunitaria para limitar el acceso a su agente SNMP

TABLA 24-2 Opciones ACL de Squid

Se establecen dos entradas **acl**: una para el sistema local y otra para los demás; las opciones de **http_access** permiten primero el acceso al sistema local y después lo niegan a los demás.

```
acl milan src 192.168.0.0/255.255.255.0
acl all src 0.0.0.0/0.0.0.0
http_access allow milan
http_access deny all
```

Aquí se muestran las entradas predeterminadas que encontrará en su archivo **squid.conf**, junto con entradas para la red de ejemplo milan. Encontrará estas entradas en la sección ACCESS CONTROLS del archivo **squid.conf**.

```
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl milan src 192.168.0.0/255.255.255.0
acl SSL_ports port 443 563
```

El orden de las opciones **http_access** es importante. Squid empieza desde el principio y trabaja hacia abajo, se detiene en la primera opción de **http_access** con una entrada ACL coincidente. En el ejemplo anterior, se permitieron sistemas locales que coincidieron con el primer comando de **http_access**, mientras otros cayeron hasta el segundo comando **http_access** y se negaron.

En el caso de sistemas que utilizan proxy, también se controla a qué sitios acceden. Para una dirección de destino, debe crear una entrada **acl** con el calificador **dst**. Éste toma como argumento la dirección del sitio. Luego se crea una opción **http_access** para controlar el acceso a esa dirección. En el siguiente ejemplo se niega el acceso a cualquiera que utilice el proxy cuyo sitio de destino sea **conejo.mipista.com**. Si una red local accede a la Web a través del proxy, utilice esos comandos para restringir el acceso a ciertos sitios.

```
acl miconejo dst conejo.mipista.com
http_access deny miconejo
```

Aquí se muestran las entradas **http_access** ya están definidas en el archivo **squid.conf**, junto con una entrada para la red **milan**. Se niega el acceso a usuarios externos, mientras se permite el acceso a host de una red local y el host local (el host de servidor Squid).

```
http_access allow localhost
http_access allow milan
http_access deny all
```

También se califican las direcciones por dominio. A menudo, se hace referencia a sitios Web utilizando únicamente el dominio. Por ejemplo, se puede hacer referencia a un sitio llamado **miplaya.com** usando sólo el dominio **miplaya.com**. Para crear una entrada **acl** que refiriendo el dominio, utilice la opción **dstdomain** o **srcdomain** para dominios de destino y fuente, respectivamente. Recuerde que esa referencia abarcará todos los host de ese dominio. Una entrada **acl** con la opción **dstdomain** para **miplaya.com**, restringe el acceso a **miplaya.com**, **ftp.miplaya.com**, **surf.miplaya.com**, etc. En el siguiente ejemplo, se restringe el acceso al sitio **miplaya.com** junto con todos los demás sitios y cualquier host del dominio **miplaya.com**:

```
acl laplaya dstdomain .miplaya.com
http_access deny laplaya
```

Tiene la opción de incluir varios dominios o direcciones en una entrada **acl** para hacer referencia a éstos como grupo, pero no puede hacer que un dominio sea subdominio de otro. Por ejemplo, si **mimantadeplaya.com** es subdominio de **miplaya.com**, no pueden incluirse ambos en la misma lista **acl**. En el siguiente ejemplo se restringe el acceso a **miplaya.com** y **misurf.com**:

```
acl playas dstdomain .miplaya.com .misurf.com
http_access deny playas
```


Una entrada **acl** también puede utilizar un patrón para especificar ciertas direcciones y dominios. En el siguiente ejemplo, se niega el acceso a cualquier URL con el patrón “chocolate”, pero se permite a los demás.

```
acl Choc1 url_regex chocolate
http_access deny Choc1
http_access allow all
```

Squid también soporta métodos de autenticación **ident** y **proxy** para controlar el acceso de usuario. En el siguiente ejemplo se permite sólo a los usuarios **daniel** y **chris** utilizar la caché de Squid:

```
ident_lookup on
acl goodusers user carlos daniel
http_access allow buenosusuarios
http_access deny all
```

Cachés

Squid usa primordialmente el protocolo de caché de Internet (ICP), para comunicarse con otras cachés Web. También ofrece soporte al protocolo de caché de hipertexto (HTCP) y el protocolo de caché de enrutamiento de matriz (CARP), más experimentales.

Al utilizar los protocolos ICP, su caché de Squid se conecta con otras cachés de Squid o servidores de caché, como el servidor proxy de Microsoft, el servidor proxy Netscape y Novell BorderManager. De esta forma, si el caché de Squid de su red no tiene copia de una página Web solicitada, se pone en contacto con otra caché para ver si está ahí, en lugar de acceder al sitio original. Se configura a Squid para conectarse con otras cachés de Squid, al conectarlo a una jerarquía de cachés. Squid permite jerarquías de cachés representadas por términos *secundarios*, *iguales* y *principales*. Se accede a las cachés iguales y secundarias en el mismo nivel y consultan automáticamente, siempre que no localiza una solicitud en la caché de su propio Squid. Si esta consulta falla, se consulta una caché principal, que después busca sus propias cachés secundaria e iguales (o su propia caché principal, si se necesita), etcétera.

Configurará una jerarquía de cachés para conectarse al servidor NLANR, al registrar su caché mediante las siguientes entradas en su archivo **squid.conf**:

```
cache_announce 24
announce_to sd.cache.nlanr.net:3131
```

Conexión a cachés

Utilice **cache_peer** para configurar conexiones principal, igual y secundaria con otras cachés. Esta opción tiene cinco campos. Los primeros dos contienen el nombre de host o dirección IP de la caché consultada y tipo de caché (principal, secundaria o igual). El tercero y cuarto son puertos HTTP e ICP de la caché, generalmente 3128 y 3130. El último se usa para opciones de **cache_peer**, como **proxy-only** para no guardar localmente objetos buscados, **no-query** para cachés sin soporte ICP y **weight**, que asigna prioridad a una caché principal. En el siguiente ejemplo se configura una conexión a una caché principal:

```
cache_peer sd.cache.nlanr.net parent 3128 3130
```


Configuración de memoria y disco

Squid ofrece varias opciones para configurar la memoria caché. La opción `cache_mem` define la memoria asignada principalmente a objetos en uso (objetos en transmisión). Si está disponible, el espacio también se usa para objetos que se acceden con frecuencia (objetos muy activos) y solicitudes fallidas (objetos de caché negativo). La opción predeterminada es 8 MB. En el siguiente ejemplo se asignan 268 MB:

```
cache_mem 256 MB
```

Se especifican con más detalle los tamaños mínimo y máximo de objetos guardados en el disco o memoria. En el disco, use `maximum_object_size` y `minimum_object_size`. El máximo predeterminado es 4 KB. El mínimo predeterminado se establece en 0, indicando que no existe valor mínimo. En el caso de la memoria, utilice `maximum_object_size_in_memory` y `minimum_object_size_in_memory`.

La opción `cache_swap_low` permite establecer barras para reemplazar objetos en su caché.

Para designar dónde se ubican los objetos de la caché, use la opción `cache_dir`. Aquí se especifican los directorios a usarse para su caché.

Configuraciones administrativas

La dirección de correo electrónico para el administrador de su caché de Squid se especifica en la opción `cache_mgr`.

Si ejecuta Squid como usuario root, entonces Squid cambiará su ID de grupo y usuario de **root** a **nobody**. El ID de grupo cambiará a **nogroup**. Esto es para proteger el acceso del usuario root. Si quiere ejecutar Squid como otro usuario distinto de root, Squid mantendrá al usuario original con su ID de usuario. Si, cuando ejecuta Squid desde el usuario root, quiere designar otro usuario que no sea nobody, utilice `cache_effective_user` para cambiar el ID de usuario y `cache_effective_group`, para cambiar el grupo.

También tiene la opción de especificar un nombre de host especial, que habrá de desplegarse en mensajes de error. Use `visible_hostname` para establecer el nombre.

Registros

Squid mantiene varios registros detallando acceso, rendimiento de la caché y mensajes de error.

- **access.log** almacena solicitudes enviadas a su proxy.
- **cache.log** almacena mensajes del servidor Squid, como mensajes de errores e inicio.
- **store.log** almacena información acerca de la caché de Squid, como objetos agregados o eliminados.

Se utiliza el administrador de caché (`cachemgr.cgi`) para administrar caché y ver estadísticas en el administrador de caché, conforme se ejecuta. Para ejecutar el administrador de caché, utilice browser para ejecutar la secuencia de comandos `cachemgr.cgi` (esta secuencia de comandos se debe colocar en el directorio `cgi-bin` de su servidor Web).

Aceleración del servidor Web: caché de proxy inversa

Aunque las cachés de Squid mejoran el acceso de clientes al servidor Web, Squid también reduce la carga en un servidor Web. Los servidores Web sobrecargados con solicitudes pueden mover las

páginas susceptibles de usar la caché al servidor proxy Squid, que sirve entonces como especie de sitio alterno, manejando solicitudes para esas páginas. En efecto, el servidor Web se acelera. A esta caché se le conoce como caché de proxy inversa, porque se enfoca en el servidor y no en el cliente. Una caché de proxy inversa interceptará solicitudes a un servidor, procesando cualquiera de sus páginas guardadas en caché. Sólo las solicitudes de páginas no guardadas en caché se reenvían al servidor Web original.

Para configurar una caché de proxy inversa, use la directiva **http_port** con la opción **accel** (para Squid 2.4 y anteriores utilice las directivas **httpd_accel**). Para especificar una ubicación particular para aceleradores, utilice **defaultsite** (esto reemplaza a **http_accel_host** utilizado en Squid 2.4 y antes).

```
http_port 3128
http_port 192.168.0.25:80 accel defaultsite=conejo.mipista.com
```

Si su servidor proxy Squid y el servidor Web operan en el mismo host, necesita especificar el puerto usado por el servidor Web. No puede ser el mismo puerto que Squid usa. Emplee la directiva **cache_peer** con la opción **port**, para especificar el puerto del servidor. Luego especifique la dirección del servidor Web con la directiva **cache_peer** y la opción **originserver**. En el siguiente ejemplo, el servidor Web utiliza el puerto 80, mientras Squid emplea el 3128:

```
http_port 3128 # Puerto de proxy Squid
cache_peer port 80
cache_peer originserver localhost # Dirección IP del servidor Web
```

Con Squid 2.6, las cachés transparentes tienen soporte directo como una opción de **http_port**, **transparent**.

```
http_port 3218 transparent
```

Para especificar el uso de directivas host (host virtuales), recurra a la opción **vhost**. Esto reemplaza la directiva **http_accel_uses_host_header** usada en versiones anteriores. En el caso de hosts IP virtuales utilice **vport**.

```
http_port vhost
```

Además, las entradas DNS para la red externa usarían la dirección IP del servidor proxy como nombre de host del servidor Web, al dirigir todos las solicitudes de éste al servidor proxy. Las entradas DNS para la red interna utilizarían la dirección IP del servidor como nombre host del servidor Web, permitiendo que el proxy redirija a éste solicitudes no guardadas en la caché. Si su red sólo utiliza un servidor DNS, configure un servidor Split DNS, para especificar direcciones internas y externas.

Servidores de correo

Los servidores de correo proporcionan a los usuarios de Internet servicios de correo electrónico. Tienen protocolos TCP/IP propios, como el protocolo simple de transferencia de correo (SMTP, Simple Mail Transfer Protocol), protocolo de oficina postal (POP) y protocolo de acceso a correo de Internet (IMAP, Internet Mail Access Protocol). Los mensajes enviados por Internet a través de servidores de correo para dar servicio de dominios locales. Un *dominio* se ve como subred de Internet, con servidor propio, para manejar mensajes de correo enviados a usuarios de esa subred o recibidos de éstos. Cuando un usuario envía un mensaje por correo, primero lo envía desde su sistema host al servidor de correo. Luego, el servidor de correo envía el mensaje a otro servidor de correo en Internet, para dar servicio a la subred en que se ubica el destinatario. Después, el servidor de recepción de correo envía el mensaje al sistema host del destinatario.

En cada etapa, toma lugar un tipo de operación diferente al usar diferentes agentes (programas). Un agente de correo de usuario (MUA, Mail User Agent) es un programa de cliente de correo, como mail o Elm. Con un MUA, un usuario redacta un mensaje de correo y lo envía. Después, un agente de transferencia de correo (MTA, Mail Transfer Agent) transporta los mensajes a través de Internet. Los MTA son servidores de correo que utilizan SMTP para enviar mensajes en Internet de un servidor de correo a otro, que transporta entre subredes. En sistemas Linux y Unix, el MTA de uso común es Sendmail, un daemon servidor de correo que revisa constantemente mensajes entrantes de otros servidores de correo y envía mensajes salientes a servidores apropiados. Otros MTA populares son Postfix, Exim, Courier y Qmail (véase la tabla 25-1). Los mensajes entrantes recibidos en un servidor de correo se distribuyen al usuario mediante agentes de entrega de correo (MDA, Mail Delivery Agents). Casi todos los sistemas Linux utilizan procmail como su MDA, que toma mensajes recibidos en el servidor de correo y entrega a las cuentas de usuario (véase procmail.org para conocer más información).

Agentes de transporte de correo

Muchas distribuciones de Linux instalarán y configurarán automáticamente Sendmail o Postfix. Al iniciar su sistema, pueden enviarse y recibirse mensajes entre usuarios locales mediante el uso de Sendmail o Postfix. También es posible configurar su sistema Linux para ejecutar un servidor POP. Los servidores POP almacenan correo de usuarios hasta iniciar sesión para acceder a sus mensajes, en lugar de ser enviados directamente los correos a sus respectivos host. Postfix y Sendmail se discutirán en este capítulo.

Agente	Descripción
Sendmail	Agente de transferencia de correo Sendmail, con soporte del consorcio Sendmail sendmail.org
Postfix	Agente de transferencia de correo rápido, fácil de configurar y seguro compatible con Sendmail y diseñado para remplazarlo postfix.org
Qmail	MTA rápido, flexible y seguro con su propia implementación; compite con Postfix qmail.org
Exim	MTA basado en smail3 exim.org
Courier	Courier MTA courier-mta.org

TABLA 25-1 Agentes de transferencia de correo

Courier es un MTA rápido, pequeño y seguro que mantiene cierta compatibilidad con Sendmail. El paquete de software Courier también incluye POP, IMAP y servidores de correo Web, junto con servicios de listas de correo. También soporta amplios métodos de autenticación, incluyendo contraseñas Shadow, PAM y LDAP.

Exim es un MTA rápido y flexible similar a Sendmail. Desarrollado en la Universidad de Cambridge, tiene una implementación muy diferente a Sendmail.

Qmail también es un MTA rápido y seguro, pero tiene poca compatibilidad con Sendmail. Cuenta con archivos de configuración y mantenimiento propios. Como Postfix, tiene un diseño modular, que utiliza diferentes programas para cada tarea de correo. También se enfoca en seguridad, velocidad y configuración sencilla.

***NOTA** Los mensajes enviados en un sistema independiente sencillo requieren de una interfaz de bucle cerrado. Casi todas las distribuciones hacen esto automáticamente durante el procedimiento de instalación. Una interfaz de bucle cerrado permite a su sistema dirigirse a sí mismo, con lo que se envía correo a sí mismo. Una interfaz de bucle cerrado usa el nombre de host **localhost** y una dirección IP especial, reservada para uso del sistema local, 127.0.0.1. Examine su archivo **/etc/hosts** para ver si su interfaz de bucle cerrado se ha configurado como host local. Debe ver **127.0.0.1 localhost** en la lista como la primera entrada.*

Correo recibido: registros MX

Una dirección de correo consta de nombre de usuario y dirección host. La dirección host toma la forma de un nombre de dominio plenamente calificado, mostrando una lista de nombres de host y nombres de dominio, separados por puntos. Casi todos los usos de nombre de usuario, como conexiones FTP, traducen el nombre de host en dirección IP y utilizan ésta para ubicar el sistema host. Los mensajes de correo operan casi de la misma forma. Sin embargo, se aseguran que el servicio de nombres de dominio (DNS, Domain Name Service) determine a qué host se envía realmente el mensaje. Tal vez el host especificado en la dirección de correo no sea el host al que se debe hacer realmente la entrega. A menudo, redes diferentes especificarán un servidor de correo al que debe enviarse el correo para los host de una red. Por ejemplo, un correo dirigido al host **conejo.mipista.com**, realmente podría enviarse al host **tortuga.mipista.com**. Es probable que éste ejecute un servidor de correo POP, al que los usuarios en **conejo.mipista.com** acceden para leer su correo.

Tales servidores de correo se asocian con diferentes host mediante registros de intercambio de correo, conocidos como registros MX, en una configuración DNS de la red. Cuando el correo se recibe en una red, se revisa primero la configuración DNS de la red, en busca de registros MX para determinar si el correo se entregará a un host diferente de la dirección del mensaje de correo. Por ejemplo, el siguiente registro MX dice que cualquier correo para el host **conejo.mipista.com** se entregará a **tortuga.mipista.com**; **tortuga.mipista.com** es un intercambiador de correo para **conejo.mipista.com**:

```
conejo.mipista.com. IN MX 0 tortuga.mipista.com
```

Un host tiene varios intercambiadores, cada uno con prioridad diferente. Si uno está inactivo, se accederá al siguiente con la prioridad más alta. Este diseño ofrece una entrega de correo más robusta, permitiendo que unos cuantos servidores bien mantenidos manejen el correo recibido, en vez de hacerlo cada host por cuenta propia.

Los registros de intercambio de correo también se usan para direcciones de correo en las que no existen hosts. Por ejemplo, se puede designar un host virtual o utilizar el nombre de dominio como dirección. Para utilizar un nombre de dominio, se correlaciona un registro MX con el nombre de dominio para un servidor de correo en la red. El correo dirigido al nombre de dominio se envía al servidor de correo. Por ejemplo, con el siguiente registro MX, el correo enviado a **mipista.com** se entregaría a **tortuga.mipista.com**, que ejecutaría un servidor de correo como Sendmail:

```
mipista.com. IN MX 0 tortuga.mipista.com
```

El correo dirigido a **jorge@mipista.com** se enviaría a **jorge@tortuga.mipista.com**.

NOTA Los registros MX no sólo se usan para correo entrante, sino para correo saliente. Un registro MX especifica un servidor de correo que se usará para transmitir correo de un host dado a una red más grande.

Los registros MX entran en juego con ciertas configuraciones de correo, como enmascaramiento o servicios de correo centralizados. No se requieren registros MX. Si tiene un sistema independiente o red pequeña sólo con unos cuantos host, tal vez quiera se reciba el correo directamente en diferentes host.

Postfix

Postfix es un MTA rápido, seguro y flexible, diseñado para remplazar Sendmail conforme mantiene la mayor compatibilidad posible. Escrito por Wietse Venema y lanzado originalmente como IBM Secure Mailer, ahora está disponible bajo la licencia GNU (postfix.org). Postfix fue creado con la seguridad en mente, tratando todos los correos entrantes como posibles riesgos de seguridad. Postfix usa muchos de los mismos directorios y archivos de Sendmail, haciendo uso de las envolturas de Sendmail, permitiendo a los clientes de Sendmail interactuar de manera perfecta con servidores Postfix. Postfix también es más sencillo que Sendmail, empleando su propia configuración.

En vez de un programa grande, Postfix se implementa como colección de pequeños programas, cada uno diseñado para realizar una tarea específica relacionada con correo. Un daemon maestro de Postfix se ejecuta continuamente y maneja el uso de otros daemons de Postfix, sólo en ejecución cuando se necesitan. Un daemon **bounce** maneja correo que no puede entregar, un daemon **trivial-rewrite** redirige mensajes y el daemon **showq** proporciona información de colas de impresión.

Comandos de Postfix

Varios comandos de Postfix permiten manejar tareas propias de servidor. El comando **sendmail** envía mensajes. Debe utilizar **mailq** para desplegar el estado de sus colas de correo. El comando **newaliases** toma alias de correo, mostradas en listas en archivos de alias y almacena en un archivo de base de datos que puede usarse por Postfix.

El comando **postmap** se usa para mantener varios archivos de base de datos utilizados por Postfix, como el archivo de alias para alias de correo y archivo de acceso restringiendo mensajes recibidos por el servidor. También puede implementar estos archivos de base de datos, como SQL para MySQL, con el fin de permitir una administración más sencilla. La página Man **mysql_table** ofrece información detallada sobre cómo configurar el soporte para bases de datos SQL (revise **pgsql_table** para soporte de base de datos de PostgreSQL). También puede utilizar LDAP en vez de SQL (**ldap_table**).

Además, Postfix brinda herramientas de nivel más bajo, en que todas empiezan con el término **post**, como el comando **postalias**, para mantener la base de datos alias y **postcat**, desplegando archivos de cola de impresión.

Configuración de Postfix: main.cf

La configuración de Postfix se maneja manipulando parámetros de su archivo de configuración, **main.cf**. Un archivo **/etc/postfix/main.cf** predeterminado se instala con Postfix, y casi todos los valores esenciales ya están configurados. Los nombres de parámetros tienden a ser amigables para el usuario. Por ejemplo, las ubicaciones de directorios se especifican con parámetros que finalizan con el término **directory**, como **queue_directory** para ubicación de las colas Postfix y **daemon_directory** para la ubicación de daemons de Postfix. Las opciones predeterminadas ya se encuentran implementadas para casi todos los parámetros. Por ejemplo, las opciones predeterminadas se configuran para controles de recursos particulares, como tamaño de mensaje, límites de tiempo y número de mensajes permitidos por colas. Edite el archivo **main.cf** para cambiar los valores de parámetro con el fin de satisfacer propias necesidades. Después de hacer cualquier cambio, sólo necesita cargar de nuevo la configuración al usar el comando **postfix reload**:

```
postfix reload
```

Parámetros de red

Lo más seguro es que necesite configurar varios parámetros de red. Para facilitar este proceso, Postfix define parámetros que almacenan información de red clave, como **myhostname**, que almacena el nombre de host de su sistema y **mydomain**, para almacenar el nombre de dominio de su red. Por ejemplo, **myhostname** sería el nombre de host **tortuga.mipista.com**, mientras **mydomain** sería solo **mipista.com**. Los parámetros como **myhost** y **mydomain**, por sí solos, se utilizan como valores asignados a otros parámetros. En el siguiente ejemplo, **myhostname** y **mydomain** se configuran al servidor de correo host en ejecución y su dominio de red:

```
myhostname=tortuga.mipista.com
mydomain=mipista.com
```

El parámetro **myorigin** especifica la dirección de origen del correo electrónico enviado al servidor. Como opción predeterminada, se asigna el valor del parámetro **myhostname**, mostrado aquí. Observe que se coloca un signo **\$** antes de la variable **myhostname** para evaluarla.

```
myorigin=$myhostname
```

Si usa un sólo sistema, ligado directamente a Internet, tal vez quiera mantener su configuración, al etiquetar el correo como si fuera enviado por su host. Sin embargo, si su sistema opera como puerta de enlace, su servidor de correo envía correo de diferentes hosts a esa red. Tal vez quiera cambiar la dirección de origen por el nombre de dominio, para que el correo se perciba como enviado por el dominio.

```
myorigin=$mydomain
```

Redes locales

El parámetro **mydestination** almacena una lista de dominios de los que su servidor de correo recibirá correo. Como opción predeterminada, ésta incluye **localhost** como nombre de host del sistema.

```
mydestination = $myhostname localhost.$mydomain
```

Si quiere que el servidor de correo reciba correos para toda una red, necesita especificar su nombre de dominio. De esa forma, el servidor recibirá correo dirigido sólo al dominio, en vez de un host específico.

```
mydestination = $myhostname localhost.$mydomain $mydomain
```

Además, si su host va por otros nombres de host y existen registros DNS identificando a su host por tales nombres, necesita especificar también esos nombres. Por ejemplo, su host también puede ser un servidor Web al cual dirigir el correo. Un host **tortuga.mipista.com** también puede identificarse como el sitio Web **mipista.com**. Ambos nombres tendrían que figurar en la lista del parámetro **mydestination**.

```
mydestination = $myhostname localhost.$mydomain $mydomain www.$mydomain
```

Si su sistema es una puerta de enlace para una o más redes locales, especifíquelas con el parámetro **mynetworks**. Esto permite a su servidor de correo retransmitir el correo dirigido a esas redes. Las redes se especifican mediante sus direcciones IP. El parámetro **relay_domains** permite especificar direcciones de correo de redes, a las que puede retransmitir mensajes. Como opción predeterminada, se establecen en **mydestination**:

```
mynetworks=192.168.0.0
relay_domains=$mydestination
```

Los host en la red local conectada a Internet por la puerta de enlace, necesitan conocer la identidad del host de transmisión. Configure esto con el parámetro **relayhost**. También, **myorigin** debe configurarse sólo a **mydomain**. Si existe un servidor DNS que identifica la puerta de enlace como servidor de correo, configure **relayhost** al valor de **mydomain**. Si no, entonces **relayhost** debe configurarse al nombre de host específico del servidor de puerta de enlace y correo. Si su red local no ejecuta un servidor DNS, asegúrese de configurar **disable_dns_lookups** a **yes**.

```
relay_host=$mydomain
```

Conexiones directas

Si su sistema se conecta directamente a Internet y usa un proveedor de servicio de Internet (ISP, Internet Service Provider) para recibir correo, configure Postfix como cliente nulo que sólo enviará correo. Configure el parámetro **relay_host** sólo a su nombre de dominio. Además, en el archivo **master.cf**, convierta en comentario las entradas del servidor SMTP y el agente de entrega local.

```
relayhost = $mydomain
```


Enmascaramiento

Si su servidor de correo opera en una puerta de enlace para una red local y quiere ocultar los host de esa red, opte por enmascarar el host local, permitiendo parezca que todo el correo proviene del dominio en general, en vez de un host particular. Para configurar esta opción, recurra al parámetro **masquerade_domains**. En el siguiente ejemplo, todo el correo enviado a un host local como **conejo.mipista.com** se dirigirá como procedente de **mipista.com**. De esta manera un mensaje enviado por el usuario **chris@conejo.mipista.com** se envía como si proviniera de **chris@mipista.com**:

```
masquerade_domains = $mydomain
```

El correo recibido no se enmascara como opción predeterminada. Esto permite a Postfix entregar todavía el mensaje recibido a hosts particulares. Si quiere que el correo recibido también se enmascare, debe agregar el parámetro **envelope_recipients** para hacer una lista de valores asignados al parámetro **masquerade_class**. En ese caso, Postfix ya no será capaz de entregar correo recibido.

Dominios y cuentas virtuales

Si su red tiene implementados dominios virtuales, necesitará configurar una tabla de dominio virtual y luego especificar esa tabla con la opción **virtual_maps**. Para configurar una tabla sólo necesita una lista de nombres virtuales y direcciones reales en un archivo de texto como **/etc/postfix/virtual**. Luego use el comando **postmap** para crear una tabla de Postfix:

```
postmap /etc/postfix/virtual
```

En el archivo **main.cf**, especifique la tabla con el parámetro **virtual_maps**. Entonces, Postfix usará la tabla para buscar dominios virtuales

```
virtual_maps = hash:/etc/postfix/virtual
```

NOTA Véase las preguntas más frecuentes de Postfix en **postfix.org** para conocer información detallada acerca de la forma de configurar Postfix para una puerta de enlace, estación de trabajo local o host conectado directamente a Internet (servidor nulo).

En lugar de usar cuentas de correo para usuarios reales en un sistema, puede configurar una cuenta virtual. Las cuentas virtuales se administran en archivos de texto de Postfix estándar, en bases de datos SQL o como entradas LDAP. Se prefieren las bases de datos SQL para administrar gran número de cuentas virtuales. Para soporte SQL, primero cree tablas en una base de datos MySQL para dominios (los dominios virtuales), usuarios (cuentas de usuario) y reenvío (alias). Los archivos de configuración de dominio virtual correspondientes, crearán listas de información como base de datos, tablas y host a ser usadas, como **mysql_virt.cf** para acceso a base de datos SQL y **mysql_users.cf** para acceso a la tabla de usuario. Revise la documentación en **postfix.org** para conocer información detallada.

Postfix Greylisting Policy Server

Postfix también soporta “listas grises” con Postfix Greylisting Policy Server. Estas listas bloquean correo basura con base en sus métodos de correo, en vez del contenido, dependiendo del hecho de que los correo basura no intentarán reenviarse si se rechazan (**greylisting.org**). Se rechazan los mensajes de fuentes nuevas desconocidas, después de lo que un MTA válido hará un reenvío, mientras un correo basura no lo hará. Para dar soporte a Postfix Greylisting Policy Server, Postfix se

configura para delegar la política de acceso a un servidor. En el directorio `/etc/postfix` se utilizan los archivos `postgrey_whitelist`, para excluir direcciones de correo electrónico de listas grises.

El Postfix Greylisting Policy Server se ejecuta como servidor independiente, usando su propia secuencia de comandos de inicio. La página Man de `postgrey` proporciona información detallada acerca de las opciones del servidor.

Control del acceso de usuario y host

Con un archivo de acceso, se controla el acceso de ciertos usuarios, host y dominios. El archivo de acceso trabaja muy similar al utilizado en Sendmail. Las entradas se hacen en un archivo de texto que comienza con el usuario, host o nombre de dominio o dirección, seguido por una acción que habrá de tomarse. Un usuario, host o dominio se acepta, se niega sin un mensaje o se niega con un mensaje. Una vez se hayan incluido las entradas, se instalan en un archivo de base de datos de Postfix con el comando `postmap`:

```
postmap /etc/postfix/access
```

Luego puede usar el archivo de acceso en varias operaciones para controlar clientes, destinatarios y emisores.

El acceso también se controla con sistemas de prevención de abuso en correo (MAPS, Mail Abuse Prevention System), que proporciona el servicio RBL+, colección de direcciones de correo de bases de datos basadas en DNS (mail-abuse.com). Estas bases de datos, como Realtime Blackhole List (RBL), muestran una lista de direcciones de correo que se sabe son usados por abusadores de correo. Un dominio o host se compara contra una lista mantenida por el servicio, a la que se accede en un servidor local o directamente en un sitio en línea. Varias operaciones de Postfix permiten utilizar bases de datos de MAPS para controlar el acceso por parte de clientes, destinatarios o emisores.

Revisiones de encabezado y contenido

Con el parámetro `header_checks`, se especifica una tabla Postfix donde se hace una lista de criterios para negar mensajes. Revise el archivo `/etc/postfix/header_checks` para conocer más detalles. Los criterios son patrones que relacionan encabezados de mensajes. Puede hacer que los mensajes coincidentes se nieguen, se nieguen con una contestación, simplemente se eliminen o registren con una advertencia. Tiene la opción de tomar varias acciones, incluidas REJECT, DISCARD, WARN, HOLD e IGNORE.

```
header_checks = regexp:/etc/postfix/header_checks
```

La base de datos, en este caso `/etc/postfix/header_checks`, tendrá líneas, cada una con una expresión regular y acción correspondiente. La expresión regular es un estándar que se indica con `regexp` en el parámetro `header_checks`, u otra adecuada para Perl Compatible Regular Expression, `prece`.

El parámetro `body_checks` permite revisar el contenido de un mensaje de texto, línea por línea, mediante expresiones regulares y acciones como las usadas por `header_checks` en un archivo `/etc/postfix/body_checks`.

Control de clientes, emisores y destinatarios

Con el parámetro `smtpd_client_restrictions`, se restringe el acceso al servidor de correo por parte de ciertos clientes. Las restricciones que se aplican incluyen `reject_unknown_client_hostname`, rechazará cualquier cliente con direcciones no resueltas; `permit_mynetworks`, permite el acceso a cualquier cliente definido por `mynetworks` y `check_client_access`, revisará una

base de datos de acceso para ver si un cliente debe aceptarse o negarse. Los parámetros **reject_rbl_client** y **reject_rhsbl_client** rechazarán clientes de dominios específicos.

```
smtpd_client_restrictions = permit_mynetworks, \
    reject_unknown_client, check_client_access, reject_maps_rbl
```

La restricción **reject_rbl_client** rechaza direcciones de dominio, de acuerdo con un servicio MAPS específico. Puede ser un sitio en línea o local, configurado para proporcionar servicios. La restricción **reject_rhsbl_client** rechaza direcciones de host.

```
smtpd_client_restrictions = reject_rbl_client relays.mail-abuse.org
```

Para implementar restricciones de un archivo de acceso, se usa la directiva **hash** y el nombre del archivo.

```
smtpd_client_restrictions = hash:/etc/postfix/access
```

El parámetro **smtpd_client_restrictions** correspondiente trabaja de forma muy parecida a su contraparte de cliente, pero controla el acceso de emisores específicos. Tiene muchas de las mismas restricciones, pero agrega **reject_non_fqdn_sender**, que rechazará cualquier encabezado de correo sin nombre de dominio plenamente calificado y **reject_sender_login_mismatch**, requerirá verificación de emisor. La restricción **reject_rhsbl_sender** rechaza direcciones de dominio, de acuerdo con un servicio MAPS especificado.

El parámetro **smtpd_recipient_restrictions** restringirá los destinatarios de los que el servidor aceptará correos. Las restricciones incluyen **permit_auth_destination**, permitiendo mensajes autorizados y **reject_unauth_destination**, rechazando mensajes no autorizados. La restricción **check_recipient_access** revisa redes locales en busca de la dirección de un destinatario. La restricción **reject_unknown_recipient_domain** rechaza direcciones de destinatarios sin entrada DNS. La restricción **reject_rhsbl_recipient** rechaza direcciones de dominio de acuerdo con un servicio MAPS especificado.

Se refinan aun más las restricciones con parámetros como **smtpd_helo_restrictions**, requiriendo un comando HELO de un cliente. Entre los parámetros de restricción se incluyen **reject_invalid_hostname**, para revisar sintaxis incorrectas, **reject_unknown_hostname**, para hosts sin entrada DNS y **reject_non_fqdn_hostname**, para hosts cuyos nombres no estén plenamente calificados. El parámetro **strict_rfc821_envelopes** implementará una conformidad estricta con el protocolo de envoltura.

Sendmail

Sendmail opera como servidor para enviar y recibir mensajes de correo. Sendmail escucha cualquier mensaje de correo recibido de otros hosts, dirigidos a usuarios en hosts de la red a la que sirve y, al mismo tiempo, maneja mensajes que los usuarios envían a usuarios remotos, determinando a qué host se enviarán. Conocerá más acerca de Sendmail en sendmail.org, incluida la documentación en línea y paquetes de software actuales. El grupo de noticias de Sendmail es **comp.mail.sendmail**. También se obtiene una versión comercial en **sendmail.com**.

El servidor de nombres de dominio de su red designa el host ejecutando el servidor Sendmail. Es su host de correo. Los mensajes se envían a este host, cuyo servidor Sendmail envía luego el mensaje al usuario apropiado y sus hosts. En el archivo de configuración de su servidor de nombres de dominio, la entrada del host de correo se especifica con una entrada MX. Para imprimir la cola de mensajes para entregas futuras, utilice **mailq** (o **sendmail -v -q**). Esto ejecuta Sendmail con instrucciones para imprimir la cola de impresión.

El paquete de software Sendmail contiene utilerías para administrar su servidor Sendmail. Estas incluyen mailq, que despliega la cola de mensajes salientes, mailstats, para mostrar estadísticas en el uso del servidor de correo; hostat, que ofrece estadísticas de host remotos conectadas con el servidor de correo y praliases, que imprime la lista de alias de correos en el archivo `/etc/aliases`. Algunas utilerías, como mailq y hoststat, sólo invocan Sendmail.

Sendmail ahora mantiene todos los archivos de configuración y bases de datos en el directorio `/etc/mail`. Aquí encontrará el archivo de configuración de macros de Sendmail, `sendmail.mc`, además de varios archivos de base de datos (véase la tabla 25-2). Muchos han cambiado sus nombres con el lanzamiento de Sendmail 8.10. Por ejemplo, el archivo de ayuda ahora es `/etc/mail/helpfile`, en vez de `/etc/sendmail.ht`. Los archivos especializados proporcionan soporte a ciertas características como `access`, que permite controlar acceso a diferentes host y redes a su servidor de correo y `virtusertable`, para designar host virtuales. Estos archivos tienen versiones de texto y base de datos. La versión de base de datos termina con la extensión `.db` y es el archivo realmente empleado por Sendmail. Haga sus entradas en la versión de texto y después efectúe los cambios al generar una versión de base de datos correspondiente. Las versiones de base de datos se generan usando el comando `makemap` con la opción `hash` y una operación de redirección para el texto y archivo de base de datos. Por ejemplo, para negar el acceso a un host particular, coloque la entrada apropiada para éste en el archivo `/etc/mail/access`, al editar el archivo usando cualquier procesador de texto. Luego, genere la versión `/etc/mail/acces.db` del archivo de acceso, cambie el directorio `/etc/mail` y utilice el siguiente comando:

```
cd /etc/mail
makemap hash access < access
```

Para regenerar todos los archivos de base de datos, sólo utilice el comando `make` en el directorio `/etc/mail`:

```
make
```

Ciertos archivos y directorios se usan para administrar correo recibido y enviado. El correo entrante suele almacenarse en el directorio `/var/spool/mail` y los mensajes salientes se almacenan en el directorio `/var/spool/mqueue`, con subdirectorios para diferentes usuarios. Monitoreo y mensajes de error se registran en el archivo `/var/log/maillog`.

NOTA Si su servidor de correo sirve a varios hosts, necesitará insertarlos en el archivo `/etc/mail/local-host-names`.

Alias y LDAP

Sendmail ahora soporta el protocolo ligero de acceso a directorio (LDAP, Lightweight Directory Access Protocol), permitiendo el uso de un servidor separado para administrar las consultas de Sendmail acerca de direcciones de correo de usuario. En vez de mantener alias y archivos `virtusertable` en diferentes servidores, Sendmail utiliza soporte a LDAP para usar simplemente un servidor LDAP centralizado, para localizar destinatarios. Las direcciones de correo se buscan en el servidor LDAP, en vez de buscar varios alias y archivos `virtusertable` en diferentes servidores. LDAP también ofrece autenticación segura de usuarios, permitiendo el acceso controlado a cuentas de correo. En el siguiente ejemplo se habilita el soporte LDAP en Sendmail en el archivo `sendmail.mc`:

```
FEATURE('enrutamiento_ldap')dnl
LDAPROUTE_DOMAIN('mipista.com')dnl
```

Archivo	Descripción
<code>/etc/mail/sendmail.cf</code>	Archivo de configuración de Sendmail
<code>/etc/mail/sendmail.mc</code>	Archivo de configuración de macros M4 de Sendmail
<code>/etc/mail/submit.cf</code>	Archivo de configuración de Sendmail para modo de remisión, donde Sendmail no se ejecute como servidor, sino sólo envíe correo
<code>/etc/mail/submit.mc</code>	Archivo de configuración M4 de Sendmail para modo de remisión de correo de Sendmail
<code>/etc/aliases</code>	Archivo de alias de Sendmail para listas de correo
<code>/etc/aliases.db</code>	Archivo de base de datos de alias de Sendmail, generado por el comando newaliases al utilizar el archivo de alias
<code>/etc/mail/access</code>	Archivo de texto de acceso de Sendmail, control de acceso para vigilancia o retransmisión de mensajes de diferentes hosts, redes o usuarios; utilizado para generar el archivo access.db
<code>/etc/mail/access.db</code>	Archivo de base de datos de acceso de Sendmail, generado por el archivo de texto de acceso
<code>/etc/mail/local-host-names</code>	Archivo de host local de Sendmail para múltiples hosts usando el mismo servidor de correo (originalmente sendmail.cw)
<code>/etc/mail/trusted-users</code>	Archivo de usuarios confiables de Sendmail (originalmente sendmail.ct)
<code>/etc/mail/error-header</code>	Archivo de encabezados de error de Sendmail (originalmente sendmail.oE)
<code>/etc/mail/helpfile</code>	Archivo de ayuda de Sendmail (originalmente sendmail.ht)
<code>/etc/mail/statistics</code>	Archivo de estadísticas de Sendmail (originalmente sendmail.st)
<code>/etc/mail/virtusertable</code>	Archivo de texto de la tabla de usuario virtual; correlaciona direcciones de dominio de usuario virtual, permitiendo que los dominios virtuales sean hosts en un sólo sistema; haga entradas en este archivo y después utilícelo para generar el archivo virtusertable.db
<code>/etc/mail/virtusertable.db</code>	Base de datos de la tabla de usuarios virtuales de Sendmail, generada a partir del archivo virtusertable
<code>/etc/mail/mailertable</code>	Archivo de texto de la tabla de agentes de correo de Sendmail, empleado para invalidar el enrutamiento a sus dominios
<code>/etc/mail/mailertable.db</code>	Archivo de base de datos de la tabla de agentes de correo de Sendmail, generado a partir del archivo mailertable
<code>/etc/mail/userdb</code>	Archivo de base de datos de usuario de Sendmail
<code>/etc/mail/domaintable</code>	Archivo domaintable de Sendmail, correlaciona un nombre de dominio con otro
<code>/etc/mail/domaintable.db</code>	Archivo de base de datos domaintable de Sendmail, generado a partir del archivo domaintable
<code>/var/spool/mail</code>	Correo entrante
<code>/var/spool/mqueue</code>	Correo saliente
<code>/var/spool/mailllog</code>	Archivo de registro de correo

TABLA 25-2 Archivos y directorios de Sendmail

Como opción, Sendmail todavía permite el uso de alias, ya sea para enviar o recibir correo. Revisa un archivo de base de datos de alias llamado **aliases.db**, para almacenar nombres de alias y direcciones de correo electrónico asociadas. Esto suele utilizarse para correo administrativo, donde el correo tal vez se envíe al usuario root del sistema y después se redirija a la dirección de correo del administrador de sistema real. También se utilizan direcciones de host de alias, que permiten dirigir hosts en su red usando sólo sus alias. Las entradas de alias se mantienen en el archivo **/etc/aliases**. Este archivo consta de registros de alias de una línea asociando alias con direcciones de usuario. Se edita este archivo para agregar nuevas entradas o cambiar entradas antiguas. Después se almacenan para búsqueda en el archivo **aliases.db** mediante el comando **newaliases**, ejecutando Sendmail con instrucciones para actualizar el archivo **aliases.db**.

Los alias permiten asignar nombres diferentes a una dirección de correo electrónico o colección de direcciones de correo electrónico. Una de sus características más útiles es la creación de una lista de correo de usuarios. El correo dirigido a un alias se enviará al usuario o lista de usuarios asociados con el alias. Una entrada de alias consta de un nombre de alias terminando con dos puntos, seguido por un nombre de usuario o lista de usuarios separados por comas. Por ejemplo, en el caso del alias **criticodecine** con el usuario **jorge@conejo.mipista.com**, se utiliza la siguiente entrada:

```
criticodecine:    jorge@conejo.mipista.com
```

Para el alias **cantantes**, con los usuarios locales **alicia** y **larisa**, utilice

```
cantantes:    alicia, larisa
```

Se utilizan también los alias como direcciones de destino, en cuyo caso se expandirán a sus respectivas direcciones de usuario. Por ejemplo, el alias **artistas** se expandirá a través de los alias **criticodecine** y **cantantes**, a los usuarios **jorge@conejo.mipista.com**, **alicia** y **larisa**:

```
artistas:    criticodecine, cantantes
```

Una vez haya hecho sus entradas en el archivo **/etc/mail/aliases**, necesita generar una versión de base de datos mediante el comando **newaliases**:

```
newaliases
```

NOTA Sendmail ahora soporta el modo de remisión de correo en que Sendmail no se ejecuta como servidor junto con privilegios de root correspondientes. En cambio, sólo envía correo. De este modo, los clientes de correo electrónico invocan directamente a Sendmail para enviar correo. El modo de remisión de correo usa el archivo de configuración **/etc/mail/submit.cf**, configurado al utilizar macros en el archivo **/etc/mail.submit.mc**.

Configuración de Sendmail

El archivo de configuración principal de Sendmail es **sendmail.cf**, y ubicado en el directorio **/etc**. Este archivo consta de una lista a veces larga de definiciones de correo estableciendo opciones, designan MTA y definen reglas de reescritura de dirección. Una serie de opciones configura características, como el tamaño máximo de mensajes de correo o nombre de archivo de host. Los MTA son remitentes de correo a través de los que Sendmail enruta mensajes. Las reglas de reescritura “reescriben” una dirección de correo, para enrutar a su destino un mensaje

a través de las conexiones de Internet apropiadas (estas reglas son complejas). Revise el HOWTO y documentación en línea para conocer una explicación detallada de Sendmail.

Las definiciones de **sendmail.cf** son complejas y confusas. Para simplificar el proceso de configuración, Sendmail permite el uso de macros usados para generar el archivo **sendmail.cf** mediante el preprocesador M4 (requiere la instalación del paquete **sendmail-cf**). Las macros se colocan en el archivo **/etc/mail/sendmail.mc**. Aquí, se usan las macros para designar definiciones y características deseadas para Sendmail, después las macros se utilizan para generar definiciones apropiadas y describir las reglas en el archivo **sendmail.cf**. Como parte del paquete de Sendmail, están disponibles varias versiones especializadas del archivo **sendmail.mc** en el directorio **/usr/share/sendmail-cf**. Éstas comienzan con un nombre de sistema y tienen el sufijo **.mc**. En muchas distribuciones, una versión especializada hecha a la medida para su distribución ya está instalada como archivo **/etc/mail/sendmail.mc**.

Una vez se configura el archivo **sendmail.mc**, se utiliza el siguiente comando, ejecutable en el directorio **/etc/mail**, para generar un archivo **sendmail.cf** (asegúrese primero de respaldar su archivo **sendmail.cf** original). Puede cambiar el nombre del archivo **sendmail.mc**, para reflejar la configuración específica y tener todos los archivos diferentes **.mc** que quiera y usarlos para implementar diferentes configuraciones:

```
make -C /etc/mail
```

Como opción, utilice el comando de macro **m4** original:

```
m4 sendmail.mc > /etc/mail/sendmail.cf
```

Entonces necesitará reiniciar el servidor Sendmail para llevar a cabo la configuración.

En el archivo **sendmail.mc**, se configuran diferentes aspectos de Sendmail mediante **define** para configurar el valor de las variables de Sendmail o una macro Sendmail, definida para configurar una característica de Sendmail particular. Por ejemplo, para asignar la variable **PROCMail_PATH** al directorio **/usr/bin/procmail**, utilice lo siguiente:

```
define('PROCMail_MAILER_PATH', '/usr/bin/procmail')
```

De manera similar, si existen variables que no quiere se definan, se eliminan con el comando **undefine**:

```
undefine('UUCP_RELAY')
```

Para especificar el tipo de sistema operativo en que se ejecuta su servidor Sendmail, se utiliza la macro **OSTYPE**. En el siguiente ejemplo se especifica el sistema operativo Linux:

```
OSTYPE('linux')
```

La macro **MAILER** especifica los agentes de entrega de correo (MDA) a ser usados. Puede tener más de uno. Generalmente, necesitará un agente de entrega de correo como procmail para entregar el correo a hosts de su red. Además, Sendmail opera como MDA para recibir mensajes de hosts en su red local, que después enviará a la conexión más grande.

```
MAILER(procmail)
MAILER(smtp)
```


Sendmail también permite un amplio número de características que necesita activar explícitamente. Haga esto con la macro **FEATURE**. Consulte la tabla 25-3 para conocer una lista de características de Sendmail de uso común. En el siguiente ejemplo se activa la característica **redirect**, usada para informar a un emisor que un destinatario ahora está en una dirección diferente:

```
FEATURE(redirect)
```

Característica	Descripción
use_cw_file	Revisa hosts servidos por el archivo /etc/mail/local-host-names de servidor de correo.
use_ct_file	Lee una lista de usuarios del archivo /etc/trusted-users . Son usuarios confiables que cambian el nombre de emisor para sus mensajes.
redirect	Rechaza todas los correos dirigidos a la dirección REDIRECT, que proporciona una dirección de reenvío en lugar del archivo /etc/aliases .
mailertable	Utiliza un archivo de la tabla mailer, /etc/mail/mailertable , para invalidar el enrutamiento de dominios particulares.
domaintable	Utiliza un archivo de tabla de dominio, /etc/mail/domaintabl , para correlacionar un dominio con otro. Es útil si cambia su nombre de dominio.
allmasquerade	Causa que las direcciones de destinatarios también se enmascaren como si fueran del host enmascarado
masquerade_entire_domain	Enmascara todos los hosts en un dominio específico en MASQUERADE_AS .
masquerade_envelope	Enmascara al emisor y destinatario de envoltura junto con encabezados
virtusertable	Para host virtuales; correlaciona direcciones virtuales con reales.
nullclient	Convierte un servidor de sendmail en cliente nulo, que sólo reenvía mensajes de correo a un servidor de correo central para procesarlos.
local_procmail	Utiliza procmail como remitente de correo local.
smrsh	Utiliza Sendmail Restricted Shell (smrsh) para enviar correos.
promiscuous_relay	Permite la retransmisión de un mensaje, aceptando el correo recibido de fuera de su dominio y enviándolo a hosts fuera de su dominio.
relay_entire_domain	Permite que cualquier host de su dominio transmita correo (como opción predeterminada, se limita esto a hosts en la base de datos de acceso)
relay_hosts_only	Revisa el permiso de retransmisión para hosts particulares en vez de dominios.

TABLA 25-3 Características de Sendmail

Característica	Descripción
<code>accept_unqualified_senders</code>	Permite que las direcciones de correo electrónico de remitentes sean nombres de usuario simples en lugar de nombres plenamente calificados incluyendo nombres de dominio.
<code>accept_unresolvable_domains</code>	Permite a Sendmail aceptar nombres de dominio que no se pueden resolver. Es útil para usuarios de una red local bloqueados por la firewall del espacio de nombres DNS completo. Como opción predeterminada, Sendmail impone que los dominios en direcciones se resuelvan con DNS.
<code>access_db</code>	Acepta o rechaza correo de dominios y hosts en la base de datos de acceso.
<code>blacklist_recipients</code>	Bloquea correo a ciertos usuarios, como los que nunca deben recibir correo (usuarios nobody y host).
<code>dnsbl</code>	Rechaza hosts en Realtime Blackhole List. Administrado por el sistema de prevención de abuso de correo (MAPS LLC) y está diseñado para limitar el transporte de correo masivo no deseado (mail-abuse.org).
<code>ldap_routing</code>	Habilita el uso de LDAP.

TABLA 25-3 Características de Sendmail (continuación)

Además, puede configurar ciertas opciones. Estas son variables comenzando con el prefijo **conf** que puede configurar y asignar valores mediante el comando **define**. Existe amplia cantidad de opciones de configuración; casi ninguna necesita cambio. En el siguiente ejemplo se define la opción de configuración **confAUTO_REBUILD**, que reconstruirá automáticamente la base de datos de alias, si es necesario:

```
define('confAUTO_REBUILD')
```

Es necesario colocar ciertas macros y tipos de macros en el archivo **sendmail.mc** en una secuencia particular, como se muestra aquí. Observe que **MAILER** está hacia el final y **OSTYPE** al principio. Las definiciones de macros locales (**define**) y de entradas **FEATURE** siguen tras las entradas **OSTYPE** y **DOMAIN**:

```
VERSIONID
OSTYPE
DOMAIN
define
FEATURE
local macro definitions
MAILER
LOCAL_RULE_*
LOCAL_RULESETS
```

Antes de la entrada **FEATURE**, es necesario insertar definiciones de macros locales y opciones de configuración afectando una característica particular. Por ejemplo, la característica **redirect** utiliza el archivo de alias. Cualquier definición local del archivo alias necesita insertarse antes de la característica **redirect**.

```
define('ALIAS_FILE','/etc/aliases')
FEATURE(redirect)
```

Es necesario que tenga cuidado con la manera en que introduce los comentarios en un archivo **sendmail.mc**. Este archivo se lee como flujo de macros, ignora todos los espacios en blanco, incluidos cambios de línea. No se buscan caracteres de comentarios especiales. En cambio, debe simular indicadores de comentario usando los comandos **dnl** o **divert**. El comando **dnl** da instrucciones para que se ignoren todos los caracteres siguiendo ese comando hasta el nuevo cambio de línea e incluyendo éste. Si coloca un comando **dnl** al principio de una línea de texto en el archivo **sendmail.mc**, tiene por efecto convertir esa línea en un comentario, ignorando todo lo que esté en la línea (incluido su cambio de línea). Incluso las líneas vacías requerirán una entrada **dnl** para ignorar el carácter de nueva línea.

dnl tendrá `/etc/mail/sendmail.cf` al ejecutar esto con la configuración **dnl** de la macro **m4** a través de un preprocesador:

```
dnl
```

Como opción, puede utilizar el comando **divert**. Éste ignorará todos los datos hasta llegar a otro comando **divert**:

```
divert (-1)
Este es el archivo de configuración de macros utilizado para generar
el archivo /etc/mail/sendmail.cf. Si modifica el archivo a regenerar
deberá regenerar /etc/mail/sendmail.cf ejecutando la
macro m4
divert (0)
```

Para que Sendmail funcione, sólo requiere se definan las macros **OSTYPE** y **MAILERS**, junto con las características y opciones necesarias. Aquí se muestra un archivo muy simple de Sendmail:

```
mysendmail.mc
dnl Mi archivo sendmail.mc
OSTYPE('linux')
define('PROCMAIL_MAILER_PATH','/usr/bin/procmail')
FEATURE(redirect)
MAILER(procmail)
MAILER(smtp)
```

Un archivo **sendmail.mc** suele contener muchas entradas más, sobre todo de parámetros y características. Revise el archivo `/etc/mail/sendmail.mc` en su sistema para ver las entradas predeterminadas estándar de Sendmail.

Enmascaramiento de Sendmail

En el caso de un servidor de correo retransmitiendo mensajes de hosts locales a Internet, tal vez quiera enmascarar la fuente de los mensajes. En redes grandes contando con sus propios servidores de correo conectados a Internet, el enmascaramiento de Sendmail hace los mensajes enviados por el host local aparezcan como enviados por el servidor de correo. Su dirección de host se reemplazará con la del servidor de correo. El correo devuelto se envía entonces al servidor de correo y almacena en buzones de correo de servidor POP o IMAP a los que después pueden acceder los usuarios en hosts locales. Además, las entradas en la tabla de usuarios virtuales del servidor pueden reenviar correos correspondientes a usuarios en el host local.

El enmascaramiento suele usarse para ocultar hosts locales tras un nombre de dominio. También se enmascaran todos los subdominios. Este método aplica en situaciones donde un ISP o administrador de red han asignado a su red su propio nombre de dominio. Luego se enmascaran todos los mensajes de correos como si provinieran de su nombre de dominio, en vez de un host particular o desde cualquier subdominio que pueda tener. Por ejemplo, si el nombre oficial de dominio de una red es **mipista.com**, todos los mensajes procedentes del host en la red **mipista.com**, como **conejo.mipista.com** y **tortuga.mipista.com**, se enmascaran para simular que provienen de **mipista.com**. Si la red **mipista.com** tiene una subred cuyo dominio es **miplaya.com**, cualquier mensaje de **miplaya.com** también se enmascara como si viniera de **mipista.com**.

El enmascaramiento se activa con el comando **MASQUERADE_AS**. Esto toma como argumento el nombre que quiera para enmascarar su correo. Por lo general, el nombre usado es sólo el de dominio, sin host de correo. En el siguiente ejemplo, el correo se enmascara como sólo **mipista.com**. El correo enviado desde un host local como **tortuga.mipista.com** aparecerá como si fuera enviado por sólo **mipista.com**:

```
MASQUERADE_AS('mipista.com')dnl
```

También deberá especificar los hosts y dominios en la red local que su servidor Sendmail debe enmascarar. Si ha decidido enmascarar todos los host en su red local, necesita configurar la característica **masquerade_entire_domain**, como en

```
FEATURE('masquerade_entire_domain')dnl
```

Si, en cambio, quiere enmascarar hosts particulares o su dominio tiene varios subdominios que quiere enmascarar, póngalos en una lista en la entrada **MASQUERADE_DOMAIN**. Haga una lista de hosts particulares o dominios completos. Por ejemplo, dada una red local con hosts locales **tortuga.mipista.com** y **conejo.mipista.com**, haga una lista de éstos con **MASQUERADE_DOMAIN** para ser enmascarados. El dominio se enmascara como especifica la entrada **MASQUERADE_AS**.

```
MASQUERADE_DOMAIN('tortuga.mipista.com conejo.mipista.com')dnl
```

Si quiere enmascarar todos los hosts de su red local, basta colocar en la lista el nombre de dominio de red local. Si su red local soporta varios subdominios, haga también una lista de éstos para enmascararlos. Por ejemplo, para enmascarar todos los hosts en el dominio **miplaya.com**, use la siguiente entrada:

```
MASQUERADE_DOMAIN('mipista.com miplaya.com')dnl
```

Si tiene una larga lista de dominios o hosts, o quiere cambiar de manera sencilla aquellos que deben ser enmascarados, se cuelan en un archivo para leerse por Sendmail. Especifique el archivo con el comando **MASQUERADE_DOMAIN_FILE**:

```
MASQUERADE_DOMAIN_FILE('misdominios')dnl
```

Si sólo quiere enmascarar todos los host de su dominio local, use la característica **masquerade_entire_domain**:

```
FEATURE(masquerade_entire_domain)dnl
```

Una configuración común para una red local especifica el nombre de dominio en las entradas **MASQUERADE_AS** y **MASQUERADE_DOMAIN**. Al utilizar el ejemplo **miisp.com** para el dominio, las entradas se ven así:

```
MASQUERADE_AS('miisp.com')dn1
FEATURE(masquerade_entire_domain)dn1
```

Si se quiere enmascarar como dominio de correo de ISP, use el dominio ISP en la entrada **MASQUERADE_AS**, como se muestra aquí:

```
MASQUERADE_AS('miisp.com')dn1
MASQUERADE_DOMAIN('mipista.com')dn1
```

Cuando recibe un correo del exterior llevando sólo la dirección **mipista.com**, su red necesita saber a qué host enviarlo. Este es el host designado como servidor de correo para la red **mipista.com**. Esta información se ofrece en un registro de intercambio de correo (MX), en su configuración DNS que especificará el correo enviado a **mipista.com** lo manejará el servidor de correo (en este caso, **tortuga.mipista.com**):

```
mipista.com. IN MX 0 tortuga.mipista.com
```

También debe estar seguro de que la transmisión MX está habilitada con la característica **relay_based_on_MX**:

```
FEATURE(relay_based_on_MX)dn1
```

Todos los mensajes aparecerán como si se originaran en el host del servidor de correo. Por ejemplo, si su servidor de correo se ejecuta en **tortuga.mipista.com**, el correo enviado de un host local llamado **conejo.mipista.com** también aparecerá como enviado desde **tortuga.mipista.com**.

También se pueden enmascarar direcciones de destinatario, para que el correo enviado a usuarios de su host local se envíe, en cambio, a la dirección enmascarada. Use la característica **allmasquerade** para habilitar el enmascaramiento de destinatario:

```
FEATURE(allmasquerade)dn1
```

Configuración de servidores y clientes de correo

Sendmail puede utilizarse como servidor de correo, administrando correo para varios hosts en una red, o como un cliente de correo, que administra correo para usuarios locales de un host particular. En una configuración de red simple, tiene a todos los hosts ejecutando Sendmail en una configuración de cliente y un host operando como servidor de correo, retransmitiendo correo a los hosts de la red. En el caso de una red local conectada a Internet, su host local ejecutará Sendmail en una configuración de cliente y su puerta de enlace ejecutará Sendmail en una configuración de servidor (aunque el servidor de correo no deba ejecutarse necesariamente en la puerta de enlace). El servidor de correo transmite mensajes del host de la red local a Internet. El servidor de correo también se usa para bloquear acceso no deseado de hosts externos, como los que envían correo basura. Una configuración básica de cliente o servidor de Sendmail requiere sólo unas características en el archivo **/etc/mail/sendmail.mc**. La configuración predeterminada instalada en su sistema permite el uso en un sólo host, administrando mensajes entre usuarios en ese host. Para habilitar el uso de cliente y servidor, necesitará cambiar el archivo **/etc/mail/sendmail.mc**.

Configuración de Sendmail para una red simple

Algunas distribuciones, como Fedora, configuran inicialmente Sendmail para trabajar sólo en el sistema en que se ejecuta, **localhost**. Con el fin de usar Sendmail para enviar mensajes a otros host de una red local, necesita cambiar y agregar configuraciones en archivos **sendmail.mc** y **/etc/mail/access**. Una configuración de red simple tiene a Sendmail ejecutándose en cada host, manejando el envío de correo entre usuarios de ese host y el intercambiado con usuarios de otros hosts. Para cada configuración del servidor Sendmail, debe hacer los cambios descritos en la siguiente sección, en una configuración de red local simple.

En el caso de mensajes enviados entre hosts de su red, sólo necesita ejecutar el servidor Sendmail para cada uno, haciendo cambios a sus configuraciones de Sendmail. Es posible configurar el servidor Sendmail en uno de sus hosts para manejar la tarea de retransmitir mensajes entre hosts. Al utilizar el ejemplo de red descrito antes, los hosts **tortuga**, **conejo** y **lagarto** se ejecutarán en sus propios servidores de Sendmail. El servidor de Sendmail en el host **tortuga** se configurará para transmitir mensajes entre todos los hosts, incluido el suyo.

En cada host de su red, edite el archivo **/etc/mail/sendmail.mc** y haga los siguientes cambios. Convierta en comentarios la línea **DAEMON_OPTIONS** en el archivo **sendmail.mc** predeterminado, al colocar la palabra **dnl** al principio de ésta, como se muestra aquí. Al eliminar esta característica, permitirá recibir mensajes a través de la red local. Esta entrada restringe Sendmail al **localhost** (127.0.0.1):

```
dnl DAEMON_OPTIONS('Port=smtplib,Addr=127.0.0.1, Name=MTA')dnl
```

En el archivo **sendmail.mc** ubicado en el host que manejará la transmisión de mensajes, también necesita agregar la siguiente línea:

```
FEATURE(relay_entire_domain)dnl
```

Ejecute la operación **m4** para instalar la configuración cambiada y después reinicie el servidor.

Ahora puede enviar mensajes de un usuario a otro a través de su red. Por ejemplo, ahora **jorge@tortuga.mipista.com** puede enviar un mensaje de correo electrónico a **larisa@conejo.mipista.com**. Los servidores de Sendmail locales se encargarán de enviar y entregar correo a usuarios en sus hosts y los ubicados en otros host de red.

Configuración de Sendmail para un servidor de correo centralizado

Como opción, puede configurar un servidor de correo central para manejar todo el correo en su red. Los clientes de correo en varios hosts pueden enviar sus mensajes al servidor de correo central, que luego los transmitirá a la red más grande o Internet. Entonces, el servidor de correo central podrá recibir correo; los clientes pueden recuperarlo allí. Existen varias formas de configurar un servidor de correo central. Una de las más simples consiste en ejecutar un servidor de correo central en su host de puerta de enlace y después tener una versión de cliente nulo del servidor Sendmail, ejecutándose en un host local. Cualquier correo enviado desde host locales se reenviará automáticamente al servidor de correo central. El correo recibido sólo se entrega al servidor central, generalmente un servidor POP o IMAP también ejecutándose en el host del servidor central. Entonces los usuarios accederán al servidor POP para recuperar su correo.

Para una configuración centralizada, tiene sentido tratar a los usuarios como si tuvieran su dominio de red como dirección, en lugar de hosts separados en su red. Por tanto, el usuario **cece@conejo.mipista.com** utiliza la dirección **cece@mipista.com**, no **cece@conejo.mipista.com**. Los usuarios tienen el mismo nombre que los de sus respectivos hosts, pero los usuarios correspondientes se configurarían en la puerta de enlace para manejar correo recibido administrado por servidores POP o IMAP.

Un servidor de correo simple y efectivo requiere varios componentes:

- Un servidor de correo central en ejecución en el host de puerta de enlace
- Que cada cliente ejecute Sendmail como cliente nulo
- Enmascaramiento de todo el correo para utilizar sólo la dirección de dominio, no direcciones de host
- Un servidor POP o IMAP ejecutándose en el host de puerta de enlace para manejar correo recibido

Configuración de una estación de trabajo con conexión directa al ISP

Si ejecuta un sistema Linux que no es parte de una red, sino con una conexión directa a Internet a través de un ISP, se utilizan los servidores de correo del ISP para enviar y recibir correo. Usualmente, tendría un servidor de correo SMTP para correo saliente y un servidor POP para correo entrante. Sin embargo, también se configura Sendmail para interactuar con su ISP.

Asegúrese primero de convertir en comentario la opción **DAEMON_OPTIONS**, como se muestra en las secciones previas.

Por lo general, su ISP proporciona un servidor de correo para manejar el correo de sus hosts. Para utilizar el servidor de correo del ISP, defínalo con las opciones de **SMART_HOST**. Luego, el correo se enviará a través del servidor de correo ISP. **SMART_HOST** tiene el formato *tipo:nombredehost*, donde *tipo* es el tipo de servidor de correo utilizado, generalmente SMTP. La opción predeterminada es relay. Defina la opción **SMART_HOST** para usar su ISP con el fin de enviar y recibir correo:

```
define ('SMART_HOST', 'smtp:mail.mi-sp.com')dnl
```

La opción **SMART_HOST** indica un servidor de correo remoto específico que quiere maneje la retransmisión de sus mensajes de red. Puede ser un servidor de correo del ISP, así como cualquier servidor de correo en una red grande.

En el caso de conexiones de marcado telefónico, se usan varias opciones de configuración para controlar su conexión. La opción **confMESSAGE_TIMEOUT** permite controlar cuánto tiempo puede permanecer el correo en la cola de salida, permitiendo que su unidad de correo esté lista para marcar y enviarlo. Al configurar la opción **confDELIVERY_MODE** en **queueonly** se permite enviar correo sólo cuando usted esté listo.

La tabla de agentes de correo

La tabla de agentes de correo permite enrutar mensajes dirigidos a un host o dominio específico a un servidor de correo particular. Use la tabla de agentes de correo para que éste se dirija a un dominio virtual enrutado a un servidor de correo para su red. Para hacer referencia a un dominio completo, coloque un punto antes del nombre de dominio. Al host al que se enruta el correo se le antecede con el agente de correo utilizado, comúnmente **smtp** para Sendmail. La siguiente entrada enrutará el correo dirigido a **.miplaya.com** al servidor de correo **tortuga.mipista.com**:

```
.miplaya.com      smtp:tortuga.mipista.com
```

Las entradas se colocan en el archivo **/etc/mail/mailtable**. Una vez hechas las entradas, genere el archivo de base de datos **mailtable.db** con el comando **make**:

```
make mailertable
```

Dominios virtuales: virtusertable

Tiene la opción de definir dominios virtuales para su red. Su servidor DNS correlaciona éstos con uno o más dominios reales. Sin embargo, puede recibir mensajes con direcciones de correo para usuarios de dominios virtuales. En este caso, necesita correlacionar estas direcciones con usuarios de su dominio real para que el correo se entregue en una ubicación existente. Esta correlación la lleva a cabo la tabla de usuarios virtuales llamada **/etc/mail/virtusertable**. La tabla de usuarios virtual permite correlacionar direcciones de correo de dominios virtuales con usuarios en dominios reales. Una vez haya hecho sus entradas, genere el archivo de base de datos **virtusertable.db**, con el comando **make**:

```
make virtusertable
```

Seguridad

En el caso de la seguridad, Sendmail permite vigilar mensajes específicos, además de proporcionar autenticación y cifrado para transmisiones de Sendmail. Con la versión 8.11, Sendmail incorporó soporte a la capa de conexión segura (SSL, Secure Socket Layer) y la computadora de autenticación y seguridad simples (SASL, Simple Authentication and Security Layer). El soporte a SSL pasa por el comando de Sendmail **STARTTLS**, con “start transport layer security” (iniciar seguridad de capa de transporte). SSL proporciona autenticación, cifrado y revisiones de integridad para operaciones de Sendmail. Debe instalarse primero OpenSSL para permitir el uso de métodos de cifrado y autenticación de SSL.

El comando **AUTH** implementa SASL, al que se conoce como SMTP AUTH. SASL ofrece autenticación para usuarios y servidores de correo. Hace uso de servicios ya instalados de Kerberos para proporcionar autenticación.

Sendmail también brinda la capacidad para vigilar mensajes de dominios, hosts, IP y direcciones de usuarios específicos. Las reglas para realizar esta vigilancia se almacenan en el archivo **/etc/mail/access**. Edite este archivo y agregue sus propias reglas. Una regla consta de una dirección seguida por una acción que habrá de tomarse. (Las acciones permitidas se muestran en la lista de la tabla 25-4). Por ejemplo, para eliminar todos los mensajes del dominio **mianunciomolesto.com**, inserte

```
mianunciomolesto.com DISCARD
```

En el siguiente ejemplo se rechaza cualquier mensaje de **larisa@tortuga.micarro.com** y envía una nota de negación:

```
larisa@tortuga.micarro.com REJECT
```

Acción	Descripción
OK	Acepta el mensaje aunque otras reglas lo rechazarían (hay excepciones a la regla).
DISCARD	Descarta completamente el mensaje.
REJECT	Rechaza el mensaje, se envía una nota de negación al emisor.
RELAY	Retransmite mensajes a un dominio específico.
SMTP- mensajedecódigo	Código y mensaje que se enviará al emisor.

TABLA 25-4 Acciones de acceso

También puede especificar un mensaje de error, que se devolverá como se muestra aquí:

```
cecilia@conejo.mipista.com    ERROR:"Se retiró ayer"
```

Para enviar un mensaje de error a correos basura, se incluye un mensaje como el mostrado aquí. El primer número es un código de error:

```
correobasura.com    ERROR:"550 No aceptamos correo basura"
```

Un archivo **/etc/mail/access** con las entradas anteriores se ve así:

```
mianunciomolesto.com    DISCARD
larisa@tortuga.micarro.com    REJECT
cecilia@conejo.mipista.com    ERROR:"Se retiró ayer"
correobasura.com    ERROR:"550 No aceptamos correo basura"
```

Sendmail lee las reglas de acceso en un archivo de base de datos llamado **access.db**, también ubicado en el directorio **/etc/mail**. Para implementar sus reglas, debe regenerar el archivo **access.db** usando el archivo de acceso. Haga esto con el comando **make** empleando **access** como argumento, como se muestra aquí:

```
make access
```

Es necesario reiniciar Sendmail para leer el nuevo archivo **access.db**.

El archivo de acceso se habilita en el archivo **sendmail.mc** con la característica **access_db**:

```
FEATURE('access_db')dnl
```

El archivo de acceso negará el correo recibido de direcciones en una lista. Sin embargo, asimismo se rechaza cualquier correo enviado a éstos. De manera adicional, también se recibe correo para ciertos host de su red. Se hace esto al habilitar la opción **blacklist_recipients**, en el archivo **sendmail.mc**. Esta opción controla los destinatarios, mientras **access** controla los emisores. Las direcciones de la lista no podrán recibir correo alguno. También utilizan esta característica ciertos usuarios administrativos que nunca deben recibir correo, como **nobody** (el usuario invitado) o **ftp** (el usuario FTP):

```
FEATURE('blacklist_recipients')dnl
```

En el siguiente ejemplo no se permitirán correos enviados a **correobasura.com** (un destinatario) ni se reciben correos para **julio@lagarto.mipista.com**, **proyectosecreto@conejo.mipista.com** o **mitabladesurf.com**:

```
mitabladesurf.com    ERROR:"El dominio no existe"
julio@lagarto.mipista.com    "Se mudó a Cancún"
proyectosecreto@conejo.mipista.com    REJECT
correobasura.com    REJECT
```

Su versión de **smb.conf** de la distribución puede configurar Sendmail para utilizar **access_db** (como el caso de Fedora). El acceso sólo se permite a usuarios de host local. Si su sistema se usa como servidor de correo de una red y no ha habilitado la característica **relay_entire_domain**, deberá permitir el acceso a otros host en su red. En el archivo de acceso, puede colocar una regla **RELAY** para su red. La regla permitirá a otros hosts usar su servidor de correo para enviar fuera

mensajes de otros hosts. Esto suele hacerlo una puerta de enlace que necesita retransmitir mensajes de una red local a Internet. En el siguiente ejemplo se permite el acceso desde la red **mipista.com**:

```
mipista.com    RELAY
```

En el caso de un host específico, coloque la entrada de éste en el archivo de acceso, como se muestra aquí:

```
conejo.mipista.com    RELAY
```

Para que Sendmail sea mucho más seguro, deshabilite **VRFY**. Esta opción permite a usuarios remotos verificar la existencia de una dirección de usuario. Esto se utiliza para adivinar usuarios válidos en su sistema. Esta opción se deshabilita con la característica **noverify**:

```
FEATURE('noverify')dnl
```

Otra posible brecha de seguridad es la opción **EXPn**, que expande listas de correo y alias a sus direcciones reales. Use la característica **noexpn** para desactivarla:

```
FEATURE('noexpn')dnl
```

Como opción predeterminada, Sendmail rechazará correo de cualquier dominio no resuelto. Se invalidará esta restricción con la característica **accept_unresolvable_domains**. Sendmail también rechazará correo cuyas direcciones no tengan nombres de dominio plenamente calificados. Se invalida esta característica con **accept_unqualified_senders**.

Servidor POP e IMAP: Dovecot

El protocolo de acceso a correo de Internet (IMAP) y el protocolo de oficina postal (POP) permitirán que un servidor remoto almacene correos para usuarios que luego obtendrán su correo de éstos cuando estén listos. A diferencia de procmail, que entrega mensajes de correo directamente a una cuenta de usuario en un sistema Linux, los protocolos IMAP y POP almacenan correo, hasta que un usuario accede una cuenta en el servidor IMAP o POP. Después, los servidores transfieren cualquier mensaje recibido a la bandeja de correo local del usuario. Los ISP suelen utilizar estos servidores para proporcionar servicios de correo de Internet a usuarios. En lugar de enviarse directamente a una máquina del usuario, el mail reside en el servidor IMAP o POP hasta ser recuperado. Red Hat Linux y Fedora instalan Dovecot como sus servidores IMAP y POP. Otros servidores IMAP y POP populares disponibles son Qpopper, el servidor POP Qmail y servidores POP e IMAP de la Universidad de Washington y Courier.

Puede acceder al servidor POP desde diferentes hosts; sin embargo, cuando lo hace, todos los mensajes se transfieren a ese host. No se almacenan en el servidor POP (aunque puede configurar una opción para mantenerlos). El servidor POP sólo reenvía sus mensajes al host que lo pide. Cuando accede a sus mensajes desde cierta computadora, se transferirán a esa computadora y se borrarán del servidor POP. Si accede de nuevo al servidor POP de una computadora diferente, esos mensajes previos ya no estarán.

El protocolo de acceso a correo de Internet (IMAP), permite que un servidor remoto almacene el correo de usuarios que luego pueden iniciar sesión para acceder a su correo. A diferencia de servidores POP, los IMAP retienen mensajes de usuarios. Los usuarios incluso pueden guardar correos en el servidor de correo IMAP. Esto tiene la ventaja de mantener el correo de un usuario en

un lugar centralizado, accesible desde cualquier lugar de la red. Los usuarios pueden iniciar sesión en el servidor de correo de cualquier host en la red, leer, enviar y guardar su correo.

A diferencia de POP, IMAP permite a los usuarios configurar varias carpetas en su servidor de correo, en las que pueden organizar sus correos. IMAP también soporta el uso de carpetas compartidas, en las que varios usuarios acceden al correo en un tema dado.

Dovecot

Dovecot es una combinación de servidores IMAP y POP. Al utilizar sus propios métodos de indización, Dovecot puede manejar gran cantidad de tráfico de correo electrónico. Presenta soporte a SSL, junto con métodos de autenticación. El soporte a la base de datos de contraseña incluye contraseñas fantasma, LDAP, PAM y MySQL. El archivo `/etc/dovecot.conf` se configura para utilizar autenticación de contraseña simple con PAM, que utiliza el archivo `passwd`.

Las opciones de configuración se colocan en `/etc/dovecot.conf`. Este archivo contiene configuraciones predeterminadas comentadas con explicaciones detalladas para cada una. Las opciones específicas para **imap** y **pop3** se colocan en sus propias secciones. Estas son configuraciones básicas que se modifican:

- * **protocols** Se configura como **imap** y **pop**, además de **imaps** y **pops** para conexiones encriptadas con SSL.
- * **listen** Se establecen para protocolos IPv3 o IPv6; IPv6 se establece como opción predeterminada. La opción `listen` se configura en su sección de protocolo respectiva, como `protocol imap` o `protocol pop3`.
- * sección **auth default** Almacena opciones de autenticación predeterminadas.
- * **mechanism**, en la sección **auth** simple, como opción predeterminada. Da soporte a digest-MD5 y cran-MD5, pero no se necesitan si utiliza SSL.
- * **passwd** en la sección **auth mail_location** El método y la ubicación del almacenamiento de correo predeterminado.

Dovecot da soporte a formatos de almacenamiento mailbox o maildir (IMAP). El formato mailbox usa archivos de un sólo buzón de correo grande para almacenar varios mensajes. Las actualizaciones consumen mucho tiempo. El formato maildir utiliza un archivo separado para cada mensaje, que hace las actualizaciones mucho más eficientes. Dovecot detectará automáticamente el tipo de almacenamiento, haciendo referencia a la variable de entorno MAIL. Ésta será el archivo **mbox** del usuario en `/var/mail`. Configure Dovecot para usar un formato maildir configurando la opción **mail_location** para usar la configuración maildir, especificando el directorio en uso. El símbolo `%u` se utiliza para representar el nombre de usuario, `%h` para representar el directorio home. Los mensajes se almacenarán en un directorio **maildir** del usuario, en vez de un archivo **mbox**. Asegúrese de crear el directorio **maildir** y darle el acceso de lectura, escritura y ejecución.

```
mail_location=maildir:/var/mail/%lu/%u/maildir
```

Otros servidores POP e IMAP

Muchas distribuciones también incluyen el servidor IMAP Cyrus, que se instala y utiliza en lugar de Dovecot. Además, hay otros servidores IMAP y POP disponibles para utilizarse con Linux:

- Los servidores POP e IMAP de la Universidad of Washington (<ftp.cac.washington.edu/imap>) son parte del paquete RPM **imap** de dicha universidad. Los daemons del servidor POP son **ipop2d** e **ipop3d**. Entonces su sistema Linux se ejecuta como servidor POP2 y POP3 para su red. Estos servidores se ejecutan a través de **xinetd**. El servidor POP3 utiliza el archivo **ipop3** en **/etc/xinetd.d** e IMAP utiliza **imap**.
- El servidor IMAP Cyrus (<asg.web.cmu.edu/cyrus>) presenta controles de seguridad y autenticación mediante una estructura de buzón de correo privado, que se incrementa de manera sencilla. Diseñado para ejecutarse en servidores de correo dedicados, tiene soporte y mantenimiento de Carnegie Mellon. El nombre del daemon del IMAP Cyrus es **imapd**. Habrá un archivo llamado **imap** en el directorio **/etc/xinetd.d**.
- El servidor IMAP Courier (<courier.mta.org>), es pequeño y rápido, y ofrece amplio soporte de autenticación que incluye LDAP y PAM.
- Qpopper es el servidor POP de Berkeley (popper). Qpopper es un software sin soporte, actualmente disponible con Qualcomm, los creadores del software de correo electrónico Eudora. La página Web de Qpopper se ubica en los archivos del sitio de Eudora (eudora.com).

NOTA Los servidores IMAP y POP ofrecen cifrado SSL para transmisiones de correo electrónico seguras. También pueden ejecutarse los servidores IMAP y POP con Stunnel, para proporcionar seguridad similar. Stunnel es una envoltura de SSL para daemons como **imap**, **popd** e incluso **pppd** (conexiones de modem). En la secuencia de comandos **xinetd** del servicio, se invoca al servidor con el comando **stunnel** en vez de ejecutar el servidor directamente.

Correo basura: SpamAssassin

Con SpamAssassin, se filtra correo electrónico enviado y recibido para apartar el correo basura. El filtro examina encabezado y contenido, trazando reglas diseñadas para detectar mensajes comunes de correo basura. Cuando se detectan, entonces etiqueta el mensaje como correo basura, para que el cliente de correo lo descarte. SpamAssassin también reportará mensajes de correo para bases de datos de detección de correo basura. La versión de SpamAssassin se distribuye para Linux como la versión de fuente abierta, desarrollada por el proyecto Apache, ubicado en spamassassin.apache.org. Ahí podrá encontrar documentación detallada, preguntas frecuentes, listas de correo e incluso una lista de pruebas realizadas por SpamAssassin.

Los archivos de reglas de SpamAssassin se ubican en **/usr/share/spamassassin**. Los archivos contienen reglas para ejecutar pruebas como detectar un saludo falso en un encabezado. Los archivos de configuración de SpamAssassin se ubican en **/etc/mail/spamassassin**. El archivo **local.cf** muestra listas de opciones de SpamAssassin para todo el sistema, por ejemplo como rescribir encabezados. El archivo **init.pre** almacena configuraciones de sistema de correo basura. El archivo **spamassassin-spamc.rc** redirigirá todo el correo al cliente **spamc**.

Los usuarios configuran sus propias opciones de SpamAssassin en su archivo **.spamassassin/user_prefs**. Entre las opciones comunes se incluyen **required_score**, configurando un umbral para clasificar un mensaje como SPAM, varias opciones de listas blancas y negras que aceptan y rechazan mensajes de ciertos usuarios y dominios, así como opciones de etiquetado reescribiendo o sólo agregando etiquetas SPAM. Revise la página Mail::SpamAssassin::Conf Man para conocer más detalles.

Con el fin de configurar procmail para utilizar SpamAssassin, necesita hacer que procmail ejecute el archivo `/etc/mail/spamassassin/spamassassin-spamc.rc`. Esto filtrará todo el correo a través de SpamAssassin. El archivo `spamassassin-spamc.rc` usa el daemon `spamd`, que significa debe tener el servicio SpamAssassin en ejecución. El archivo `spamassassin-default.rc` ejecuta una secuencia de comandos menos eficiente para SpamAssassin, en vez del daemon. Si quiere filtración procmail de todo el sistema, utilice el archivo `/etc/procmailrc`; mientras para implementar el filtrado por usuario, utilice el archivo `.procmail` en el directorio home del usuario. En los respectivos archivos de procmail, agregue lo siguiente en la parte superior:

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-spamc.rc
```

Es complicado configurar Postfix para utilizarlo con SpamAssassin. Una herramienta útil para esta tarea es **amavisd-new**, una interfaz entre un agente de transporte de correo como Sendmail o Postfix y revisores de contenido como SpamAssassin y revisores de virus. Consulte ij.s.si/software/amavisd para conocer más detalles.

Servidores de impresión, noticias, búsqueda y bases de datos

Los servidores de impresión se han vuelto parte integral de todo sistema Linux. Estos permiten usar cualquier impresora en su sistema o red. Los servidores de grupos de noticias son más raros; se usan para configurar grupos de noticias para redes locales o dar soporte al servicio Usenet News de Internet. Los servidores de bases de datos se han vuelto más comunes para administrar grandes conexiones de datos en redes locales, además de servicios de Internet.

Servidores de impresión: CUPS

Las impresoras, que alguna vez se consideraron dispositivos conectados directamente a un sistema, ahora son recursos de red administrados por servidores de impresión. En el caso de una sola impresora, conectada directamente al sistema, las características de red se vuelven transparentes y la impresora sólo figura como un dispositivo más. Por otra parte, puede usar de manera sencilla las capacidades de red del servidor de impresora, para permitir que varios sistemas utilicen una sola. Aunque la instalación de la impresora es casi automática en la mayoría de distribuciones de Linux, es útil comprender el proceso fundamental. Varios sitios y recursos de impresión se muestran en la tabla 26-1.

Common Unix Printing System (CUPS) provee servicios de impresión. Está disponible de manera gratuita bajo la licencia pública GNU. Aunque ahora se incluye en casi todas las distribuciones, también pueden descargarse versiones de código fuente más recientes de CUPS desde cups.org, ofreciendo documentación detallada sobre instalación y administración de impresoras. CUPS se basa en el protocolo de impresión de Internet (IPP, Internet Printing Protocol), diseñado para establecer un estándar de impresión para Internet (para conocer más información, consulte pwg.org/ipp). Mientras los sistemas de impresión basados en las antiguas impresoras de línea (LPD) se concentraban, principalmente, en impresoras de línea, un sistema basado en IPP ofrece soporte de red, PostScript y Web. CUPS trabaja como un servidor de Internet y emplea una configuración muy similar a la del servidor Web Apache. Su soporte de red permite a los clientes acceder directamente a impresoras en sistemas remotos, sin configurar las propias impresoras. Sólo es necesario mantener la configuración en los servidores de impresión.

Resource	Description
cups.org	Common Unix Printing System
pwg.org/ipp	Protocolo de impresión de Internet
sourceforge.net/projects/lprng	Servidor de impresión LPRng

TABLA 26-1 Recursos de impresión

CUPS es el principal servidor de impresión en casi todas las distribuciones de Linux. Con **libgnomecups**, GNOME ahora brinda soporte integrado a CUPS, lo que permite que las aplicaciones basadas en GNOME accedan directamente a impresoras basadas en CUPS.

Una vez haya instalado sus impresoras y configurado su servidor de impresión, imprima y administre su cola de impresión mediante clientes de impresión. Existen varios clientes de impresora disponibles para el servidor CUPS, el administrador de impresión de GNOME, la herramienta de configuración de CUPS y varias herramientas de línea para impresión, como **lpq** y **lpc**. Estas herramientas se describen con más detalle en páginas posteriores de este capítulo. La herramienta de configuración CUPS se basa en Web, para administrar impresoras y trabajos de impresión (abra su explorador e inserte la URL **http://localhost:631**). Se despliega una página Web con entradas para manejar trabajos e impresoras y para tareas administrativas. Seleccione la entrada Administrar trabajos, para eliminar o reordenar trabajos enviados.

NOTA *Line Printer, Next Generation (LPRng), fue el servidor de impresora tradicional para sistemas Linux y Unix, pero ya se ha eliminado de muchas distribuciones de Linux. Encontrará más acerca de LPRng en sourceforge.net/projects/lprng.*

Dispositivos y configuración de impresoras

Antes de que utilice cualquier impresora, primero debe instalarla en su sistema Linux en su red. Una impresora local se instala directamente en su propio sistema. Esto requiere crear una entrada para la impresora, en el archivo de configuración definiendo el tipo de impresora que es, junto con otras características, como archivo de dispositivo y directorio spool que utiliza. En CUPS, el archivo de configuración de impresora es **/etc/cups/printers.conf**. La instalación de la impresora es moderadamente sencilla: determine qué archivo de dispositivo utilizar para la impresora y las entradas de configuración para ésta.

SUGERENCIA *Si no encuentra los controladores para su impresora, tal vez pueda descargarlos de la base de datos OpenPrinting en linux-foundation.org/en/OpenPrinting. El sitio mantiene una extensa lista de controladores.*

Archivos de dispositivo de impresoras

Linux crea de manera dinámica los nombres de dispositivo para las impresoras instaladas. En el caso de impresoras paralelas, los nombres de dispositivo serán **lp0**, **lp1** y **lp2**, dependiendo de cuántas impresoras paralelas están conectadas. El número usado para estos nombres corresponde a un puerto paralelo en su PC; **lp0** hace referencia al puerto paralelo LPT1 y **lp1** hace referencia al puerto paralelo LPT2. Las impresoras seriales utilizarán puertos seriales; se hace referencia a éstos mediante archivos de dispositivo como **ttyS0**, **ttyS1**, **ttyS2**, etc. Las impresoras conectadas a USB

tendrán una conexión de dispositivo de capa abstracta de hardware (HAL, Hardware Abstract Layer). HAL está diseñada para dispositivos extraíbles que se acoplan con facilidad a otras conexiones y tienen reconocimiento.

Directorios spool

Cuando su sistema imprime un archivo, usa directorios especiales llamados *directorios spool*. Un *trabajo de impresión* es un archivo que se imprimirá. Cuando envía un archivo a la impresora, se hace una copia de éste y se coloca en el directorio spool, configurado para esa impresora. La ubicación del directorio spool se obtiene de la entrada de la impresora, en su archivo de configuración. En Linux, el directorio spool se ubica en `/var/spool/cups`, bajo un directorio con el nombre de la impresora. Por ejemplo, el directorio spool para la impresora **miepson** se ubicaría en `/var/spool/cups/miepson`. El directorio spool contiene varios archivos para administrar trabajos de impresión. Algunos archivos usan el nombre de la impresora como extensión. Por ejemplo, la impresora **miepson** tiene los archivos **control.miepson**, que otorga control de cola de impresión y **active.miepson**, para el trabajo de impresión activo, además de **log.miepson**, el archivo de registro.

Instalación de impresoras con CUPS

Existen varias herramientas disponibles para instalar impresoras CUPS. El método más sencillo consiste en utilizar las herramientas CUPS de GNOME. Éstas suelen proporcionarse como parte de las herramientas de configuración de la impresora, incluidas con casi todas las distribuciones. Se ejecutan de manera sencilla desde el escritorio de GNOME. Como opción, se usan las herramientas de configuración basadas en explorador Web de CUPS, incluidas con el software de CUPS. Por último, tiene la opción de editar directamente los archivos de configuración de CUPS de la impresora.

Configuración de CUPS en GNOME

GNOME provee soporte para agregar y configurar impresoras CUPS. La herramienta **gnome-cups-add** detectará impresoras conectadas y encontrará el modelo. También puede agregar impresoras de red, además de configurar manualmente una impresora, especificando un puerto particular. Las impresoras USB suelen detectarse de manera automática, mientras las impresoras en serie o paralelas, más antiguas, deben configurarse manualmente. Para cambiar más adelante una configuración de impresora o especificar opciones de impresora, como el tamaño del papel, haga clic con el botón derecho en su icono y seleccione Propiedades. Los paneles Papel y Avanzados configuran especificaciones de impresión, mientras Controlador y Conexión cambian la configuración de su impresora.

Configuración de CUPS en KDE

KDE brinda soporte para agregar y configurar impresoras de CUPS a través del Centro de control de KDE. Seleccione la entrada Impresoras, bajo Periféricos. La herramienta Impresoras de KDE tiene la capacidad de realizar muchos tipos diferentes de impresión, como enviar faxes o guardar en archivos PDF. Las impresoras USB que se detectan automáticamente, se mostrarán en una lista en la ventana Impresoras de KDE. Cuando haga clic en la entrada de la impresora, los paneles Información, Trabajos, Propiedades e Instancias administrarán su impresora y sus trabajos de impresión. El panel Propiedades tiene opciones para controlar el acceso a usuarios, establecer cuotas, seleccionar un anuncio e incluso cambiar su controlador.

Para cambiar las opciones de impresora, como tamaño de la página y resolución, elija la entrada Configurar, en el menú Impresora. Este menú también permite deshabilitar la impresora o probarla. El administrador de impresión configura características generales, como las fuentes disponibles, la

vista previa utilizada o las impresoras que habrán de desplegarse. Un menú emergente para el sistema de impresora utilizado, tendrá CUPS seleccionado como opción predeterminada. Cambie a LPRng, si es necesario. Revise el KDEPrint Handbook, accesible desde el menú Documentación, para conocer información más detallada.

Herramienta de configuración basada en explorador Web de CUPS

Una de las formas más sencillas de configurar e instalar impresoras con CUPS, es utilizar la herramienta de configuración de CUPS, basada en exploradores Web. Para iniciar la interfaz Web, inserte la siguiente URL en su explorador Web:

```
http://localhost:631
```

Esto abre una pantalla de administración donde se administran y agregan impresoras. Primero se le pedirá escribir el nombre de usuario del administrador (habitualmente, **root**) y la contraseña (suele ser la contraseña del usuario root).

Con la herramienta de configuración de CUPS, se instala una impresora en CUPS a través de una serie de páginas Web, cada una pide diferente información. Para instalar una impresora, haga clic en el botón Añadir impresora, para desplegar una página donde se inserta nombre de la impresora y ubicación. La ubicación es el host al que está conectada la impresora.

Las páginas subsecuentes pedirán que ingrese el modelo de su impresora y el controlador, a elegir entre los disponibles de una lista. Una vez haya agregado la impresora, se configura. Al hacer clic en la entrada Administrar impresoras, en la página Administración, se muestra una lista de impresoras instaladas. Luego se hace clic en una impresora, para desplegar una página que le permite controlar la impresora. Aquí detiene la impresora, configura su impresión, modifica la instalación e incluso se elimina la impresora. Al hacer clic en el botón Cambiar opciones de impresora, se despliega una página donde configura su impresora, especificando resolución o tamaño de papel.

La información de configuración para una impresora se almacenará en el archivo **/etc/cups/printers.conf**. Puede examinar este archivo directamente e incluso hacer cambios. Aquí se muestra un ejemplo de una entrada de configuración de impresora. La entrada **DeviceURI** especifica el dispositivo utilizado, en este caso, una impresora USB administrada por HAL. Está actualmente desocupada, sin trabajos:

```
# Printer configuration file for CUPS
# Written by cupsd
<Printer mycannon>
Info Cannon s330
Location
DeviceURI hal:///org/freedesktop/Hal/devices/usb_device_4a9_1074_300HCR_if0_printer_noserial
State Idle
StateTime 1166554036
Accepting Yes
Shared Yes
JobSheets none none
QuotaPeriod 0
PageLimit 0
KLimit 0
OpPolicy default
ErrorPolicy stop-printer
</Printer>
```

NOTA Puede realizar todas las tareas administrativas desde la línea de comandos, con el comando `lpadmin`. Consulte la documentación de CUPS para conocer más detalles.

Configuración de impresoras remotas en CUPS

Para instalar una impresora remota conectada a un sistema de Windows u otro sistema de Linux que ejecuta CUPS, se especifica su ubicación con protocolos URL especiales. Para otra impresora CUPS en un host remoto, el protocolo utilizado es **ipp**, para Internet Printing Protocol, mientras para una impresora en Windows, sería **smb**. Los sistemas de Unix y Linux viejos que utilizan LPRng, manejarían el protocolo **lpd**.

En el caso de una impresora remota, la entrada Device URL, del archivo **cupsd.conf**, tendrá una dirección de Internet, junto con su protocolo, en lugar mostrar el dispositivo. Por ejemplo, una impresora remota en un servidor CUPS (**ipp**) se indicaría como se muestra aquí (una impresora de Windows utilizaría un protocolo **smb**):

```
DeviceURI ipp://miscosas.com/printers/queue1
```

En el caso de una impresora de Windows, primero es necesario instalar, configurar y ejecutar Samba. (CUPS utiliza Samba para acceder a impresoras de Windows.) Cuando se instala la impresora de Windows en CUPS, se especifica su ubicación usando el protocolo URL **smb**. El usuario al que se permite iniciar sesión en la impresora se inserta antes del nombre de host y se separa con un signo **@**. En casi todas las configuraciones, este es el usuario **guest**. Aquí se muestra la entrada de ubicación para una impresora de Windows llamada **mihp**, conectada a un host de Windows llamado **lagarto**. Su referencia para compartir en Samba sería **//lagarto/mihp**:

```
DeviceURI smb://guest@lagarto/mihp
```

Para habilitar CUPS en Samba, también debe asignar **cups** a la opción printing en el archivo **/etc/samba/smb.conf**, como se muestra aquí:

```
printing = cups
printcap name = cups
```

Si desea habilitar CUPS para funcionar con Samba, debe vincular **smbpool** al directorio **pool** **smb** de CUPS:

```
ln -s /usr/bin/smbpool /usr/cups/backend/smb
```

NOTA Si desea configurar una impresora compartida de Linux para que el host de Windows acceda a ella, necesita configurarla como impresora compartida **smb**. Esto se hace con **samba**.

Clases de impresora de CUPS

CUPS tiene una manera de permitir la selección un grupo de impresoras para un trabajo de impresión, en vez de seleccionar sólo una. De esta manera, si una impresora está ocupada o no funciona, otra impresora se elige automáticamente para realizar el trabajo. A estas agrupaciones de impresoras se les llama *clases*. Una vez haya instalado sus impresoras, se agrupan en diferentes clases. Por ejemplo, tal vez quiera agrupar todas las impresoras de inyección de tinta en una clase y las impresoras láser en otra, o incluir un grupo de impresoras conectadas a un servidor de impresora específico, en su propia clase. Para crear una clase, seleccione **Añadir clase**, en la página Administración e inserte el nombre de la clase. Después se agregan las impresoras.

Configuración de CUPS

Los archivos de configuración de CUPS se colocan en el directorio `/etc/cups`. Estos archivos se muestran en la tabla 26-2. La interfaz Web administra los archivos `classes.conf`, `printers.conf` y `client.conf`. El archivo `printers.conf` contiene información de configuración para diferentes impresoras instaladas. Puede editar manualmente cualquiera de estos archivos, si así lo desea.

`cupsd.conf`

El servidor CUPS se configura con el archivo `cupsd.conf` ubicado en `/etc/cups`. Debe editar manualmente las opciones de configuración; el servidor no se configura con la interfaz Web. Su instalación de CUPS instala una versión comentada del archivo `cupsd.conf` con cada opción de la lista, aunque casi todas las opciones estarán convertidas en comentarios. Las líneas comentadas tienen un símbolo `#` antes. Cada opción se documenta en detalle. La configuración del servidor usa una sintaxis de servidor Web Apache, integrado por un conjunto de directivas. Como con Apache, varias de estas directivas agrupan otras directivas en bloques.

Directivas de CUPS

Ciertas directivas permiten colocar controles de acceso en ubicaciones específicas. Se trata de impresoras o recursos, como la herramienta administrativa o directorios spool. Los controles de ubicación se implementan con la directiva `Location`. Las directivas `Allow From` y `Deny From` admiten o niegan el acceso a host específicos. CUPS soporta formas de autenticación Basic y Digest, especificadas en la directiva `AuthType`. La autenticación básica maneja usuario y contraseña. Por ejemplo, para utilizar la interfaz Web, se pide que ingrese el nombre de usuario root y la contraseña de usuario root. La autenticación Digest hace uso de información de usuario y contraseña almacenada en el archivo `/etc/cups/passwd.md5`, que utiliza las versiones MD5 de un nombre de usuario y contraseña para autenticación. La directiva `AuthClass` especifica la clase con acceso permitido. La clase `System` incluye los usuarios root, sys y system. En el siguiente ejemplo se muestra la directiva `Location` para el recurso `/admin`, la herramienta administrativa:

```
<Location /admin>

AuthType Basic
AuthClass System

## Restringe el acceso al dominio local
Order Deny,Allow
Deny From All
Allow From 127.0.0.1

</Location>
```

Nombre de archivo	Descripción
<code>classes.conf</code>	Contiene configuraciones para diferentes clases de impresora local
<code>client.conf</code>	Muestra listas de opciones específicas para clientes específicos
<code>cupsd.conf</code>	Configura el servidor CUPS, cupsd
<code>printers.conf</code>	Contiene configuraciones de impresora para impresoras locales disponibles

TABLA 26-2 Archivos de configuración CUPS

Clientes de impresión de Línea de comandos de CUPS

Una vez haya colocado un trabajo en una cola de impresión, puede utilizar cualquiera de los varios clientes de impresión para administrar los trabajos en su impresora o impresoras, como Klpq, GNOME Print Manager y la herramienta CUPS Printer Configuration. También pueden utilizarse varios clientes de impresión de línea de comandos de CUPS. Entre éstos se incluyen los comandos **lpr**, **lpc**, **lpr** y **lprm**. Printer System Switcher lo mueve de un conjunto a otro. Con estos clientes se imprimen documentos, despliega una lista de la cola de impresión, reordenan y eliminan trabajos de impresión, cancelándolos. En el caso de conexiones de red, CUPS presenta una opción de cifrados para sus comandos, **-E**, para cifrar trabajos de impresión e información de impresión enviada desde la red. En la tabla 26-3 se muestran varios métodos de administración de impresora.

NOTA Los clientes de línea de comandos tienen el mismo nombre y casi la misma sintaxis que los clientes más antiguos de línea de comandos LPR y LPRng, usados en Unix y sistemas Linux más antiguos.

lpr

El cliente **lpr** remite un trabajo, que luego toma **lpd** y lo coloca en la cola de impresión apropiada; **lpr** asume como su argumento el nombre del archivo. Si no se especifica una impresora, entonces se utiliza la impresora predeterminada. La opción **-P** permite especificar una impresora particular. En el siguiente ejemplo, el usuario imprime el archivo **prólogo** y después imprime el archivo **informe**, en la impresora con el nombre **miepson**:

```
$ lpr prólogo
$ lpr -P miepson informe
```

Administración de impresora	Descripción
Administrador de impresión de GNOME	Herramienta de administración de cola de impresión de GNOME (CUPS).
Herramienta de configuración CUPS	Imprime, administra y configura CUPS.
lpr opciones lista-archivo	Imprime un archivo, copia el archivo al directorio spool de la impresora y lo coloca en la cola de impresión para sea impreso. -P impresora imprime el archivo en la impresora especificada.
lpq opciones	Despliega los trabajos de impresión en la cola. -P impresora imprime la cola para la impresora especificada. -l imprime una lista detallada.
lpstat opciones	Despliega el estado de la impresora.
lprm opciones id-trabajodeimpresion o impresora	Elimina un trabajo de la cola de impresión. Identifique un trabajo de impresión por su número, como aparece en lpq . -P impresora elimina todos los trabajos para la impresora especifica.
lpc	Administra sus impresoras. En el indicador de comandos lpc>> , inserte los comandos para revisar el estado de sus impresoras y tomar otras acciones.

TABLA 26-3 Clientes de impresión de CUPS

lpc

Tiene la opción de utilizar **lpc** para habilitar o deshabilitar impresoras, reordenar sus colas de impresión y volver a ejecutar archivos de configuración. Para ejecutar **lpc**, escriba el comando **lpc** en el indicador de comandos de shell. Entonces se le presenta un indicador de comandos **lpc>**, en el que se insertan comandos de **lpc** para administrar sus impresoras y reordenar sus trabajos. Se despliega el comando **status** con información como nombre de la impresora, si la impresora está lista, cuántos trabajos de impresión tiene, etc. Los comandos **stop** y **start** detienen una impresora y la inician de nuevo. Las impresoras que se muestran, dependerán de qué impresoras están configuradas para servidores de impresión particulares. Una impresora configurada en CUPS sólo mostrará si ha cambiado a CUPS.

```
# lpc
lpc> status miepson
miepson:
  printer es on device 'hal' speed -1
  queuing is enabled
  printing is enabled
  1 entry in spool area
```

lpq y lpstat

Puede administrar la cola de impresión con los comandos **lpq** y **lprm**. El comando **lpq** muestra listas de trabajos en la cola de impresión. Con la opción **-P** y el nombre de la impresora, se despliega una lista de trabajos para una impresora particular. Si especifica un nombre de usuario, se muestran los trabajos de impresión para ese usuario. Con la opción **-l**, **lpq** se despliega información detallada acerca de cada trabajo. Si quiere información sobre un trabajo específico, sólo use el número de ID del trabajo con **lpq**. Para revisar el estado de una impresora, utilice **lpstat**.

```
# lpq
miepson is ready and printing
Rank   Owner   Jobs   File(s)           Total Size
active  chris   1      report            1024
```

lprm

El comando **lprm** permite eliminar un trabajo de la cola de impresión, lo que elimina el trabajo antes de imprimirse. El comando **lprm** toma muchas de las mismas opciones que **lpq**. Para eliminar un trabajo específico, utilice **lprm** con el número de trabajo. Para eliminar todos los trabajos de impresión para una impresora particular, utilice la opción **-P**, con el nombre de la impresora. **lprm**, sin opciones, elimina el trabajo en impresión. El siguiente comando elimina el primer trabajo de impresión en la cola (utilice **lpq** para obtener el número de trabajo):

```
# lprm 1
```

Herramientas administrativas de línea de comandos de CUPS

CUPS proporciona herramientas administrativas de línea de comandos como **lpadmin**, **lpoptions**, **lpinfo**, **enable**, **disable**, **accept** y **reject**. Los comandos **enable** y **disable** inician y detienen directamente trabajos de impresión, mientras los comandos **accept** y **reject** inician y detienen trabajos particulares. El comando **lpinfo** ofrece información acerca de impresoras y **lpoptions** configura opciones de impresión. El comando **lpadmin** realiza tareas administrativas, como

Herramientas administrativas	Descripción
<code>lpadmin</code>	Configuración de impresora de CUPS
<code>lpoptions</code>	Configura opciones de impresión
<code>enable</code>	Activa una impresora
<code>disable</code>	Detiene una impresora
<code>accept</code>	Permite que una impresora acepte nuevos trabajos
<code>reject</code>	Evita que una impresora acepte trabajos de impresión
<code>lpinfo</code>	Muestra una lista de dispositivos de CUPS disponibles

TABLA 26-4 Herramientas administrativas de CUPS

agregar impresoras y cambiar configuraciones. Las herramientas administrativas de CUPS se muestran en la tabla 26-4.

lpadmin

Tiene la opción de utilizar el comando **lpadmin** para establecer la impresora predeterminada o configurar varias opciones para la impresora. Se utiliza la opción **-d**, para especificar una impresora particular como destino predeterminado. Aquí se define a **miepson** como impresora predeterminada:

```
lpadmin -d miepson
```

La opción **-p** designa una impresora para la que se configuran varias opciones. En el siguiente ejemplo, se configura la información de descripción de impresora:

```
lpadmin -p miepson -D Epson550
```

Ciertas opciones controlan cuotas por usuario para trabajos de impresión. La opción **job-k-limit** configura el tamaño de un trabajo permitido por usuario, **job-page-limit** configura el límite de páginas para un trabajo y **job-quota-period** limita el número de trabajos, con un marco de tiempo específico. El siguiente comando configura un límite de 100 páginas por usuario:

```
lpadmin -p miepson -o job-page-limit=100
```

El control de acceso de usuario se determina con la opción **-u**, con una lista **allow** y **deny**. Los usuarios a los que se permite el servicio aparecen después de la entrada **allow**; a los que se niega el acceso, aparecen tras la entrada **deny**. Aquí se da acceso a **carlos**, pero se niega a **alicia** y **larisa**.

```
lpadmin -p miepson -u allow:carlos deny:alicia,larisa
```

Utilice **all** o **none**, para permitir o negar el acceso a todos o ningún usuario. Se crean excepciones usando **all** o **none**, en combinación con el acceso específico por usuario. En el siguiente ejemplo se permite el acceso a todos los usuarios, excepto **juan**:

```
lpadmin -p miepson -u allow:all deny:juan
```

lpoptions

El comando **lpoptions** configura opciones y valores predeterminados de impresión que, en esencia, definen la manera en que se imprimirán sus trabajos. Por ejemplo, se configura color o formato de página que se utilizará con una impresora particular. El usuario root mantiene las configuraciones predeterminadas para todos los usuarios en el archivo `/etc/cups/lpoptions`, mientras cada usuario crea sus propias configuraciones, guardadas en sus archivos `.lpoptions`. La opción **-l** despliega una lista con las opciones actuales para una impresora y la opción **-p**, designa a una impresora (también se configura la impresora al utilizarse con la opción **-d**).

```
lpoptions -p miepson -l
```

Las opciones de impresora se configuran con la opción **-o**, junto con el nombre y valor de la opción, **-o opcion=valor**. Puede eliminar la opción de impresora con **-r**. Por ejemplo, para imprimir ambos lados de sus páginas, defina la opción **sides** como **two-sided**:

```
lpoptions -p miepson -o sides=two-sided
```

Para eliminar la opción, utilice **-r**.

```
lpoptions -p miepson -r sides
```

Para desplegar una lista de opciones disponibles, revise las opciones de impresión estándar, en el manual de software de CUPS en cups.org.

enable y disable

El comando **enable** inicia una impresora y **disable** la detiene. Con la opción **-c**, se cancelan todos los trabajos en la cola de impresión y con la opción **-r** se transmite un mensaje explicando por qué se apaga.

```
disable miepson
```

accept y reject

Los comandos **accept** y **reject** controlan el acceso de impresoras específicas a las colas de impresión. El comando **reject** evita que la impresora acepte trabajos, mientras **accept** permite nuevos trabajos de impresión.

```
reject miepson
```

lpinfo

El comando **lpinfo** es una herramienta práctica que permite conocer cuáles dispositivos y controladores de CUPS están disponibles en su sistema. Utilice la opción **-v** para dispositivos y la opción **-m** para controladores.

```
lpinfo -m
```

Servidores de noticias

Los servidores ofrecen servicios de noticias Usenet a los usuarios de Internet. Tienen su propio protocolo TCP/IP, el protocolo de red de transferencia de noticias (NNTP, Network News Transfer Protocol). En casi todos los sistemas Linux, incluido Red Hat, el servidor de noticias InterNetNews

(INN) provee servicios de noticias (isc.org). Además, existen servidores suministrando mejor acceso a recursos de Internet.

Servidores de noticias: INN

El servidor de noticias InterNetNews (INN) accede a alimentadores de noticias de Usenet, ofreciendo clientes de noticias en su red, con un rango completo de grupos de noticias y artículos. Los artículos de grupos de noticias se transfieren usando NNTP, mientras los servidores que dan soporte a este protocolo, se les denomina *servidores NNTP*. INN fue escrito por Rich Salz, que el consorcio Internet Software Consortium (ISC) mantiene y da soporte técnico. Puede descargar las versiones actuales del sitio Web en isc.org. INN también se incluye con casi todas las distribuciones de Linux. El directorio de documentación de INN, en `/usr/share/doc`, contiene gran cantidad de ejemplos. El programa principal para INN es el daemon `innnd`.

Archivos de configuración INN

Varios archivos de configuración de INN se encuentran en `/etc/news`, incluidos `inn.conf`, `storage.conf`, `readers.conf` e `incoming.conf`; `inn.conf` configura opciones para INN y `incoming.conf` almacena los hosts, de donde se reciben alimentaciones de noticias. Se colocan entradas de hosts remotos en el archivo `readers.conf`, para permitir el acceso a su servidor de noticias. Los alimentadores de noticias se administran en los directorios de `/var/spool/news`. Aquí encontrará directorios como `article`, almacenando artículos de grupos de noticias, `outgoing` para artículos publicados por sus usuarios en grupos de noticias y `overview`, almacenando información de

Archivo	Descripción
<code>inn.conf</code>	El archivo de configuración general de INN.
<code>incoming.conf</code>	Especifica host de donde se recibirán alimentadores de noticias.
<code>cycbuff.conf</code>	Configura los búferes utilizados en el formato de almacenamiento cnfs.
<code>storage.conf</code>	Define clases de almacenamiento. Está integrado por un método de almacenamiento y los grupos de noticias que utiliza. Los métodos de almacenamiento son los formatos de almacenamiento: tradspool, timehash, timecaf y cnfs. Un método adicional, trash, desplaza los artículos.
<code>expire.ctl</code>	Configura la política de vencimiento de los artículos en el servidor de noticias.
<code>readers.conf</code>	Designa hosts, cuyos usuarios pueden acceder al servidor de noticias con lectores de noticias.
<code>ovdb.conf</code>	Configura el método de almacenamiento ovdb para resúmenes.
<code>newsfeeds</code>	Define la manera en que su servidor de noticias alimenta artículos a otros servidores de noticias.
<code>moderated</code>	Grupos de noticias con moderador.
<code>active</code>	Grupos de noticias con soporte
<code>history</code>	Registro de artículos publicados.
<code>innfeed.conf</code>	Configura procesos de alimentación de noticias para innfedd.
<code>innreport.conf</code>	Configura la utilidad innreport para generar informes basados en registro.
<code>buffindexed.conf</code>	Configura el búfer de resumen para el método buffindexed.

TABLA 26-5 Archivos de configuración INN

resumen acerca de artículos. La configuración correcta de INN es un proceso complejo que lleva mucho tiempo, asegúrese de consultar las referencias y recursos en línea, como documentos. Cuando cambie las configuraciones, asegúrese de reiniciar el servidor INN. Una secuencia de comandos **innnd** se encuentra en el directorio **/etc/rc.d/init.d**, con argumentos similares a los de la secuencia de comandos Web **httpd**. Utilice **start**, **restart** y **stop**, con la secuencia de comandos **innnd** para iniciar, reiniciar y detener el servidor INN.

***SUGERENCIA** Existe una página Man para cada archivo de configuración en INN, suministrando información detallada sobre la manera de configurar sus características.*

inn.conf

En muchas distribuciones, un archivo **inn.conf** básico ya está configurado con opciones predeterminadas. Será necesario que configure varios de los parámetros iniciales, como **domain**, que almacena el nombre de dominio de su servidor; **pathhost**, en el que se especifica el nombre de su lector de noticias, tal como quiere que aparezca en el encabezado de campo Path, para los artículos de noticias que publique y **server**, en el que se establece el IP de su lector de noticias o la dirección del nombre de dominio plenamente calificado, como **misnoticias.mipista.com**. Ya se han configurado diferentes opciones de Path diferentes para definir la ubicación de diferentes directorios INN, como **pathartiches**, configurado en artículos **/var/spool/news**, que almacenan sus artículos de grupos de noticias y **pathetc**, configurado en **/etc/news**, para sus archivos de configuración.

Formatos de almacenamiento

Los formatos de almacenamiento para el vasto número de artículos descargados y a los que se acceden con frecuencia, son la preocupación central de un servidor de noticias de escala completa como INN. INN permite seleccionar entre cuatro formatos de almacenamiento posibles: tradspool, timehash, timecaf y cnfs. El formato tradspool es el método tradicional. a partir del que los artículos se ordenan en una estructura de directorio simple, de acuerdo con sus grupos de noticias. Se sabe que consume mucho tiempo acceder a él y almacenarlo. timehash almacena artículos en directorios organizados por la hora en que fueron recibidos, facilitando la eliminación de artículos antiguos. timecaf es similar a timehash, pero los artículos recibidos en cierta hora, se colocan en el mismo archivo, haciendo que el acceso sea mucho más rápido. cnfs almacena artículos en archivos de búfer, previamente configurados. Cuando un archivo de búfer está completo, los artículos más viejos se sobrescriben con nuevos conforme llegan. Se trata de un método muy rápido, porque no se crean nuevos archivos. No hay necesidad de establecer límites de artículo máximos, pero tampoco existe control sobre cuánto tiempo se retendrá un artículo. En el archivo **storage.conf**, los formatos de almacenamiento se asignan como métodos de almacenamiento a diferentes grupos de noticias.

Acceso al lector de noticias

Los usuarios acceden a su servidor de grupo de noticias mediante lectores de noticias. Puede colocar controles a usuarios con opciones, en el archivo **readers.conf**. El control se especifica con dos componentes: autenticación y definiciones de acceso. La definición de autenticación crea una categoría de usuario, el host y sus herramientas de autenticación para usuarios. La definición de acceso aplica restricciones a categorías de usuario, como grupos de noticias a los que accede y si permite publicar artículos.

Resúmenes

INN también soporta resúmenes. Se trata de resúmenes de artículos que los lectores pueden revisar, en lugar de descargar todo el artículo para ver de qué trata. Los resúmenes tiene sus propios

métodos de almacenamiento: tradindexed, buffindexed y ovdb. Se especifica el que quiere utilizar en la característica ovmethod, en **inn.conf**. tradindexed es rápido para los lectores pero difícil de generar para el servidor. buffindexed es rápido para los servidores de noticias, pero lento para los lectores. ovdb utiliza los archivos de base de datos DB de Berkeley y es muy rápido para ambos, pero utiliza más espacio en disco. Si elige ovdb, se establecen parámetros de configuración para éste en **ovdb.conf**.

Implementación de INN

En muchas distribuciones, ya está creado un usuario **news** con un grupo de noticias, para usarlo con su daemon INN, además de configurarse directorios de noticias en **/var/spool/news**. El software INN también instala secuencias de comandos **cron**, usadas para actualizar su servidor de noticias, que elimina artículos antiguos y busca nuevos. Estos suelen colocarse en el directorio **/etc/cron.daily**, aunque pueden residir en cualquier lugar. **inn-cron-expire** elimina artículos antiguos e **inn-cron-rnews** busca nuevos. **inn-cron-nntpsend** envía artículos publicados de su sistema a otro servidor de noticias.

INN también incluye programas de soporte para mantenimiento y recuperación de fallas, además de realizar análisis estadístico sobre rendimiento y uso del servidor. cleanfeed implementa protección contra correo basura e innreport genera informes INN, basados en registros. INN también presenta un sistema de filtro muy poderoso, para filtrar artículos no deseados.

***NOTA** Leafnode es un servidor de noticias NNTP, diseñado para pequeñas redes que pueden tener conexión lenta a Internet. Obtenga el paquete de software Leafnode, junto con la documentación del sitio Web en **leafnode.org**. Además del servidor NNTP de Leafnode, el paquete de software incluye utilidades como Fetchnews, Texpire y Newsq, que envían, eliminan y despliegan nuevos artículos. slrnpull es una versión simple de Leafnode de un solo usuario, usada sólo con el lector de noticias slrn.*

Servidores de base de datos: MySQL y PostgreSQL

Dos servidores de bases de datos, totalmente funcionales, se incluyen con casi todas las distribuciones de Linux: MySQL y PostgreSQL. MySQL es, con gran margen, el más popular de los dos, aunque PostgreSQL se distingue por ofrecer más características. Recientemente, el proyecto AB de MySQL agregó MaxDB, originalmente SAP DB, suministrando capacidades comparables con las de muchos sistemas de administración de bases de datos profesionales. En este capítulo se cubrirá la forma de configurar y administrar una base de datos MySQL, asimismo se ofrecerá una breve introducción a PostgreSQL. Conocerá más acerca de estos productos en los sitios mostrados en la tabla 26-6.

Base de datos	Recurso
MySQL	mysql.com
PostgreSQL	postgresql.org
MaxDB	mysql.com

TABLA 26-6 Recursos de base de datos

Estructura de base de datos relacional

MySQL y PostgreSQL manejan una estructura de base de datos relacional. En esencia, esto significa que los datos se colocan en tablas, con identificadores de campos utilizados para relacionar los datos con entradas en otras tablas. Cada fila de la tabla es un registro, cada uno con un identificador único, como un número de registro. Las conexiones entre registros, en diferentes tablas, se implementan con tablas especiales, asociando los identificadores únicos de registros de una tabla con los de otra. La teoría e implementación de las bases de datos relacionales son temas fuera del alcance de este capítulo.

Una base de datos simple, de una sola tabla, no tiene un identificador único. Una libreta de direcciones simple, mostrando nombres y direcciones, es un ejemplo de una base de datos con una sola tabla. Sin embargo, casi todas las bases de datos acceden información compleja de diferentes tipos, relacionada de varias formas. En lugar de grandes registros con información repetida, se dividen los datos en diferentes tablas; cada una de ellas almacenando una instancia única de datos. De esta forma, los datos no se repiten; sólo se tiene una tabla que almacena un solo registro para el nombre de la persona, en lugar de repetir el nombre de la persona, cada vez que los datos hacen referencia a éste. La organización relacional se encarga entonces de la tarea de relacionar una pieza de datos con otra. De esta forma, se almacena gran cantidad de información, con archivos de base de datos relativamente pequeños.

Aunque existen muchas formas de implementar una base de datos relacional, una simple regla empírica consiste en organizar los datos en tablas, donde cada elemento tiene una instancia. A cada registro se da un identificador único, generalmente un número. Para asociar los registros de una tabla con los de otra, se crean tablas asociando sus identificadores.

SQL

El lenguaje de consulta SQL, es el utilizado en casi todos los sistemas de administración de base de datos relacionales (RDBMS, Relational DataBase Management System), incluidos MySQL y PostgreSQL. Aunque muchos RDBMS utilizan herramientas administrativas para administrar bases de datos, como MySQL y PostgreSQL en Linux, todavía debe utilizar directamente comandos SQL. Los comandos SQL comunes que puede utilizar se muestran en la tabla 26-7. Por convención, los comandos suelen escribirse en mayúsculas, aunque también pueden usarse en minúsculas.

Comando	Descripción
CREATE DATABASE <i>nombre</i>	Crea una base de datos.
CREATE TABLE <i>nombre (campos, ..)</i>	Crea una tabla dentro de una base de datos, que especifica campos.
INSERT INTO <i>nombre-tabla</i> VALUES (<i>lista de valores</i>)	Crea e inserta un registro en una tabla.
INSERT INTO <i>nombre-tabla</i> VALUES (<i>lista de valores</i>), (<i>lista de valores</i>), ...	Inserta varios registros a la vez.
SELECT <i>campo</i> FROM <i>nombre-tabla</i> WHERE <i>valor</i>	Operación de búsqueda, selecciona ciertos registros en una tabla basada en un valor de un campo especificado.
USE <i>basededatos</i>	Utiliza una base de datos particular; que sigue los comandos en donde operará.

TABLA 26-7 Comandos SQL

Al utilizar la base de datos relacional ya descrita, con el siguiente comando se creará la base de datos:

```
CREATE DATABASE misfotos
```

Antes de realizar cualquier operación en una base de datos, primero debe acceder a esta con el comando USE.

```
USE misfotos
```

Las tablas se crean con el comando CREATE TABLE; los campos de cada tabla se incluyen entre paréntesis, tras el nombre de la tabla. Para cada campo, necesita especificar un nombre, tipo de datos y otras opciones, como la presencia de un valor nulo.

```
CREATE TABLE names (
    iddepersona INT(5) UNSIGNED NOT NULL.
    nombre VARCHAR(20) NOT NULL.
    calle VARCHAR(30) NOT NULL.
    teléfono CHAR(8)
);
```

Para insertar un registro en una tabla, se utiliza el comando INSERT INTO, aunque muchas bases de datos permiten el uso de archivos de datos que se leen de una sola vez. Para agregar registros, debe utilizar el comando INSERT INTO, con el nombre de la tabla seguido por la opción VALUES, seguida a su vez por una lista de valores delimitados por comas, uno para cada campo. Los valores de carácter se incluyen entre comillas sencillas. La lista se encierra entre paréntesis. Si no ha hecho eso antes, acceda a la base de datos con el comando USE.

```
INSERT INTO nombres VALUES (1, 'juan', '111 mordor', '55-85-75-43');
```

Una vez que los valores se agreguen a las tablas, se buscan con el comando SELECT, especificando campo, nombre de tabla y valor que se buscará.

```
SELECT telefono FROM nombres WHERE telefono='55-85-75-43';
```

MySQL

MySQL está estructurado de acuerdo con un modelo cliente/servidor, con un daemon de servidor (**mysqld**) para responder solicitudes de programas de clientes. MySQL está diseñado para ser veloz, confiable y ser de uso sencillo. Se creó para constituir un sistema de administración de base de datos rápido, para bases de datos grandes y, al mismo tiempo, resulta confiable y adecuado para uso intensivo.

Para crear bases de datos, se maneja el lenguaje de programación SQL estándar. El acceso de usuario se controla mediante la asignación de privilegios.

Configuración de MySQL

MySQL soporta tres archivos de configuración diferentes, uno para configuraciones globales, otro para configuraciones de servidor específicas y una opcional, para configuraciones personalizadas por el usuario.

- El archivo de configuración **/etc/my.cnf** se utiliza para configuraciones globales aplicadas a clientes y servidores. El archivo **/etc/my.cnf** ofrece información, como las ubicaciones del

directorio de datos (**/var/lib/mysql**) y un archivo de registro (**/var/log/mysql.log**). Además del directorio base del servidor (**/var/lib**).

- El archivo **/var/lib/mysql/my.cnf** se utiliza sólo para configuraciones de servidor.
- El archivo **.my.cnf** permite a los usuarios personalizar su acceso a MySQL. Está ubicado en el directorio de inicio del usuario. Observe que es un archivo tipo punto.

En el directorio **mysql-server**, de **/usr/share/doc**, se encuentran archivos de configuración de ejemplo **my.cnf**. El directorio **mysql-server** presenta configuraciones para implementaciones pequeñas, medianas, grandes y enormes. El manual administrativo se ubica en el directorio **mysql**, de **/usr/share/doc**. Está en formato info. Utilice **info mysql** para iniciarlo y las teclas de flechas y ENTER para recorrer los menús. Aquí se encuentra más información acerca de diferentes opciones.

Configuración global: **/etc/my.cnf**

MySQL especifica opciones, de acuerdo con diferentes grupos, generalmente los nombres de herramientas de servidor. Las opciones se ordenan en segmentos de grupo. El nombre de grupo se coloca dentro de llaves y se aplican opciones tras éste. Aquí se muestra el archivo predeterminado **/etc/my.cnf**:

```
[mysqld]
datadir=/var/lib/mysql

socket=/var/lib/mysql/mysql.sock

[mysql.server]
user=mysql
basedir=/var/lib

[safe_mysqld]
err-log=/var/log/mysqld.log
pid_file=/var/run/mysqld/mysqld.pid
```

Las opciones globales de MySQL se muestran en el archivo **/etc/my.cnf**. Las opciones se configuran de acuerdo con los grupos que controlan diferentes comportamientos del servidor MySQL: **mysqld** para el daemon, **mysql.server** para opciones del servidor y **safe_mysqld** para la secuencia de comandos de inicio de MySQL. El directorio de **datadir**, **/var/lib/mysql**, es donde se colocarán sus archivos de base de datos. Las herramientas y daemons del servidor se ubican en el directorio de **basedir**, **/var/lib** y el usuario bajo el que se ejecutará MySQL, tiene por nombre **mysql**, como se especifica en la opción **user**.

Un grupo de clientes configurará opciones para enviar a clientes, como puerto y conector usados para acceder la base de datos MySQL.

```
[client]
port=3306
socket=/var/lib/mysql/mysql.sock
```

Para ver las opciones configuradas actualmente para cliente y servidor, ejecute **mysqld** directamente con la opción **--help**.

```
/usr/libexec/mysqld --help
```

Configuración de usuario: .my.cnf

Los usuarios que acceden a la base de datos del servidor, tendrán su propio archivo de configuración en su directorio de inicio: **.my.cnf**. Aquí, el usuario especifica las opciones de conexión como contraseña utilizada para acceder a la base de datos y los tiempos de espera de la conexión.

```
[client]
password=micontraseña

[mysql]
no-auto-rehash
set-variable = connect_timeout=2

[mysql-hotcopy]
interactive-timeout
```

Herremientas de MySQL

MySQL proporciona diversas herramientas (como se muestra en la tabla 26-8), incluidos servidor, cliente y herramientas administrativas. Las copias de seguridad se manejan con el comando **mysqldump**. El comando **mysqlshow** desplegará una base de datos, como lo haría al enviar el comando SQL **SELECT ***, y **mysqlimport** importa archivos de texto, como **LOAD INFILE**.

Administración de MySQL con mysql y mysqladmin

Para administrar su base de datos MySQL, utilice **mysql** como usuario **root**. El cliente **mysql** inicia el monitor de MySQL. Como usuario **root**, puede insertar comandos administrativos para crear bases de datos y tablas de base de datos, agregar o eliminar entradas, además de llevar a cabo tareas de cliente estándar, como desplegar datos.

Debe iniciar sesión como usuario **root** y abrir una ventana de terminal. Luego, ingrese el comando **mysql**. Esto iniciará una shell de monitor MySQL, con un indicador de comandos **mysql>**. Asegúrese de terminar sus comandos con un punto y coma; de otra forma, el monitor proporcionará un indicador de comandos de flecha con sangría, esperando a que se agreguen argumentos. En el monitor, el punto y coma, no la tecla **ENTER**, termina el comando.

```
# mysql -u root -p
mysql>
```

Si ha configurado un usuario **root** de MySQL, puede usar **-u root** con la opción **-p**. Se le pedirá una contraseña.

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

Comando	Descripción
mysqld	Servidor MySQL
mysql	Cliente MySQL
mysqladmin	Crea y administra bases de datos
mysqldump	Crea copias de seguridad de bases de datos
mysqlimport	Importa archivos de texto
mysqlshow	Despliega bases de datos

TABLA 26-8 Comandos de MySQL

Una vez haya iniciado el cliente **mysql**, use el comando **status**, para revisar el estado de su servidor y **show databases**, para mostrar una lista de bases de datos actuales.

```
mysql> status;
mysql> show databases;
```

Al principio, se despliegan dos bases de datos configuradas por MySQL para su propia administración: **mysql** y **test**. La base de datos **mysql** almacena información de usuario de MySQL, mientras la base de datos **test** se maneja para probar el servidor.

PostgreSQL

PostgreSQL se basa en el sistema de administración de base de datos POSTGRES, aunque usa SQL como lenguaje de consulta. POSTGRES es un prototipo de investigación de la siguiente generación, desarrollado en la Universidad de California, en Berkeley. Encontrará más información sobre éste, en el sitio Web de PostgreSQL, en **postgresql.org**. PostgreSQL es un proyecto de fuente abierta, desarrollado bajo GPL.

PostgreSQL se utiliza a menudo para dar soporte de base de datos a servidores de Internet con grandes demandas, como servidores Web. Con unos cuantos comandos simples, se crean tablas de base de datos relacionales. Utilice el comando **createuser** para generar un usuario de PostgreSQL, con el que después puede iniciar sesión en el servidor. Luego puede crear una base de datos con el comando **createdb** y construir tablas relacionales con la directiva **create table**. Con el comando **insert**, se agregarán más registros y después se ven con el comando **select**. El acceso al servidor, por parte de usuarios remotos, se controla con las entradas en el archivo **pg_hba.conf**, ubicado en el directorio de PostgreSQL, que suele ser **/var/lib/pgsql**.

La edición de PostgreSQL de Red Hat Linux incluye herramientas de Red Hat Database Graphical, para manejar y acceder de manera sencilla bases de datos de PostgreSQL. Con la herramienta de administrador, se exploran y administran bases de datos; la herramienta Visual Explain analiza procesos de consulta y el Centro de control permite administrar bases de datos en servidores.

NOTA El servidor de búsqueda e indización [ht://DIG](http://DIG) habilita búsquedas de documentos de sitios Web y FTP (htdig.org). Con éste, se indizan documentos y llevan a cabo solicitudes de búsquedas complejas.

VII

PARTE

Administración del sistema

CAPÍTULO 27

Administración básica del sistema

CAPÍTULO 28

Administración de usuarios

CAPÍTULO 29

Sistemas de archivo

CAPÍTULO 30

RAID y LVM

CAPÍTULO 31

Dispositivos y módulos

CAPÍTULO 32

Administración del Kernel

CAPÍTULO 33

Administración de copias de seguridad

Administración básica del sistema

Linux está diseñado para servir a muchos usuarios al unísono, brindando una interfaz entre usuarios y sistema con recursos, servicios y dispositivos. Los usuarios tienen shell propias mediante las que interactúan con el sistema operativo, pero quizás necesite configurar el propio sistema operativo de diferentes formas. Tal vez necesite agregar nuevos usuarios, dispositivos como impresoras y escáneres e incluso sistemas de archivos. Estas operaciones caen bajo el título de administración del sistema. A la persona ejerciendo dichas tareas se le conoce como *administrador del sistema* o *superusuario*. En este sentido, existen dos tipos de interacción con Linux: de usuarios regulares y superusuario, efectuando tareas de administración del sistema. En los capítulos de esta sección del libro se cubren operaciones como cambiar niveles de ejecución del sistema, administrar usuarios, configurar impresoras, agregar sistemas de archivos y compilar el kernel. Casi todas ellas realizadas de manera esporádica, como agregar una nueva impresora o montar un sistema de archivos. Entre otras tareas, agregar o quitar usuarios, se hacen de manera regular. Una administración básica del sistema cubre temas como acceso de superusuarios al sistema, la selección del nivel de ejecución con que se inicia, archivos de configuración del sistema y monitoreo de rendimiento.

Con Linux, tiene la capacidad para cargar diferentes versiones del kernel de Linux, además de otros sistemas operativos instalados en su sistema. La tarea de elegir un sistema operativo o kernel en el arranque se maneja con una utilidad de administración de arranque, Grand Unified Bootloader (GRUB). Es una herramienta versátil para cargar sistemas operativos compartiendo el mismo disco duro, además de seleccionar diferentes kernels, si están instalados en el mismo sistema Linux.

Control de superusuario: el usuario root

Para realizar operaciones de administración de sistema, primero debe tener derechos de acceso, como la contraseña correcta, para iniciar sesión como usuario root, que lo hace un superusuario. Debido a que el superusuario tiene el poder de cambiar casi lo que sea en el sistema, esta contraseña suele mantenerse en secreto, se cambia con frecuencia y sólo se da a quienes administran el sistema. Con la contraseña correcta, puede iniciar sesión en el sistema como administrador del mismo y

configurar el sistema de diferentes formas. Puede iniciar y apagar el sistema, al igual que cambiar a un modo operativo diferente (como al modo de un solo usuario). También puede agregar o quitar usuarios, añadir o quitar sistemas de archivos completos, respaldar y restaurar archivos e incluso designar nombre y dirección del sistema.

NOTA Si SELinux está habilitado, el acceso a superusuario se controlará con las reglas de SELinux.

Para convertirse en superusuario, inicie sesión en la *cuenta de usuario root*. Se trata de una cuenta especial reservada para operaciones de administración del sistema con acceso irrestricto a todos los componentes de su sistema operativo Linux. Se inicia sesión como usuario root, sea en la pantalla de inicio de sesión GUI (interfaz de usuario gráfica) o en el indicador de comandos de inicio de sesión, de la línea de comandos. Entonces accederá a todas las herramientas administrativas. Usando una interfaz GUI como GNOME, el usuario root accede a varias herramientas administrativas GUI de la distribución. Si inicia sesión desde la interfaz de línea de comandos, puede ejecutar comandos administrativos como **rpm** para instalar paquetes o **useradd** para agregar un nuevo usuario. Desde su escritorio GUI, también se ejecutan herramientas administrativas de línea de comandos al usar una ventana de terminal. La interfaz de línea de comandos para el usuario root usa un indicador de comandos especial, el signo de número, #. En el siguiente ejemplo, el usuario inicia sesión en el sistema como usuario root y recibe el indicador de comandos #.

```
login: root
password:
#
```

Contraseña de usuario root

Como usuario root, puede utilizar el comando **passwd** para cambiar la contraseña del inicio de sesión root, al igual que cualquier otro usuario del sistema. El comando **passwd** revisará su contraseña con módulos de autenticación insertables (PAM, Pluggable Authentication Modules), para ver si ha seleccionado una contraseña que se puede descubrir.

```
# passwd root
New password:
Re-enter new password:
#
```

Debe tomar precauciones para proteger su contraseña de usuario root. Cualquier persona que obtenga acceso como usuario root tendrá control completo de su sistema. El manual en línea para el comando **passwd** proporciona recomendaciones detalladas para el manejo y selección de su contraseña. Por ejemplo, nunca guarde su contraseña en un archivo de su sistema, y jamás seleccione una basada en información accesible, como su número de teléfono o fecha de nacimiento. Una directriz básica consiste en hacer que la contraseña sea lo más compleja posible, al utilizar una frase de varias palabras con números y letras mayúsculas y minúsculas, pero que, a pesar de ello, sea algo que todavía se recuerde fácilmente para no tener que escribirlo. Accederá a la página del manual en línea de **passwd** con el comando

```
# man passwd
```

Acceso de usuario root: su

Mientras tenga una sesión iniciada con una cuenta de usuario regular, tal vez necesite iniciar sesión como root y volverse superusuario. En general, primero debería salir de su cuenta de usuario y

Comando	Descripción
<code>su root</code>	Inicia sesión como superusuario desde una sesión de usuario; el superusuario regresa a la sesión original con <code>ctrl-d</code> .
<code>sudo comando</code>	Restringe el acceso administrativo a usuarios especificados.
<code>passwd nombre-iniciodesesión</code>	Configura una nueva contraseña para el nombre de inicio de sesión.
<code>telinit niveldeejecución</code>	Cambia el nivel de ejecución del sistema.
<code>shutdown opciones hora</code>	Apaga el sistema.
<code>date</code>	Configura la fecha y hora del sistema.

TABLA 27-1 Herramientas básicas de administración del sistema

después iniciar sesión como usuario `root`. En cambio, puede utilizar el comando `su` (Switch User, cambiar usuario) para iniciar sesión directamente como `root`, mientras se mantiene en su sesión de cuenta de usuario. Si está utiliza un escritorio GUI como GNOME, inserte el comando `su` en una ventana de terminal, o utilice `ALT-CTRL-F1` para cambiar a la interfaz de línea de comandos (`ALT-CTRL-F10` lo regresa a la interfaz GUI).

Un comando `exit` o `CTRL-D` lo regresa a su propio inicio de sesión de usuario. Cuando inició sesión como usuario `root`, puede utilizar `su` para iniciar sesión como cualquier usuario, sin proporcionar la contraseña. En el siguiente ejemplo, el usuario ya ha iniciado sesión. El comando `su` inicia entonces una sesión como usuario `root`, que convierte de usuario en un superusuario. Algunos comandos básicos de superusuario se muestran en la tabla 27-1.

```
$ pwd
/home/carlos
$ su
password:
# cd
# pwd
/root
# exit
$
```

PRECAUCIÓN Por razones de seguridad, las distribuciones de Linux no permiten el uso de `su` en una sesión Telnet para acceder al usuario `root`. En el caso de sistemas con SSH o Kerberos habilitado, el acceso de inicio de sesión seguro se ofrece al utilizar `slogin` (SSH) y `rlogin` (versión Kerberos).

Acceso administrativo controlado: sudo

Con la herramienta `sudo` se permite que usuarios ordinarios tengan acceso administrativo en el nivel de usuario `root` para ciertas tareas. Esto permite a otros usuarios realizar operaciones específicas de superusuario sin tener control completo en el nivel `root`. Encontrará más información acerca de `sudo` en [sudo.ws](https://www.sudo.ws). Para utilizar `sudo` con el fin de ejecutar un comando administrativo, el usuario debe ingresar `sudo` antes del comando. Se le envía al usuario un boleto sensible al tiempo para permitir acceso.

```
sudo date
```

La primera vez que envíe un comando `sudo` durante un inicio de sesión, se le pedirá escriba su contraseña administrativa.

El acceso se controla con el archivo `/etc/sudoers`. Este archivo incluye usuarios y comandos en ejecución, junto con la contraseña para el acceso. Si la opción **NOPASSWD** está configurada, los usuarios no necesitarán contraseña. **ALL**, dependiendo del contexto, hace referencia a todos los host de su red, comandos en nivel root y usuarios.

***NOTA** Algunas distribuciones como Ubuntu niegan, como opción predeterminada, el acceso directo a **root** y sólo permiten el acceso administrativo mediante comandos con **sudo**.*

Para hacer cambios o agregar entradas, debe editar el archivo con el comando de edición especial de sudo **visudo**. Esto invoca el editor Vi para editar el archivo `/etc/sudoers`. A diferencia de un editor estándar, **visudo** bloqueará el archivo `/etc/sudoers` y revisará la sintaxis de sus entradas. No se permite guardar cambios a menos que la sintaxis sea correcta. Si quiere utilizar un editor diferente, asígnelo a la variable EDITOR de shell.

Una entrada **sudoers** tiene la siguiente sintaxis:

```
user    host= command
```

El host es un host de su red. Especifique todos los hosts con el término **ALL**. El comando puede ser una lista de comandos, todos o algunos calificados por opciones indicando, por ejemplo, si se requiere contraseña. Para especificar todos los comandos, también se utiliza el comando **ALL**. En el siguiente ejemplo se da a **jorge** acceso completo en el nivel root a todos los comandos en todos los hosts:

```
jorge  ALL = ALL
```

Además, puede permitir que un usuario se ejecute como otro en un host determinado. Tales usuarios alternos se colocan entre paréntesis antes del comando. Por ejemplo, si quiere dar a **jorge** acceso al host **playa**, como usuario **midns**, utilice el siguiente comando:

```
jorge playa = (midns) ALL
```

Como opción predeterminada, sudo negará el acceso a todos los usuarios, incluido root. Por esta razón, el archivo `/etc/sudoers` predeterminado configura acceso completo para el usuario root a todos los comandos. La entrada **ALL=(ALL) ALL** permite el acceso para el usuario root a todos los hosts y los usuarios a todos los comandos.

```
root  ALL=(ALL) ALL
```

Para especificar un nombre de grupo, se antepone un signo % al grupo, como en **%migrupo**. De esta forma, se da el mismo acceso a un grupo de usuarios. El archivo `/etc/sudoers` contiene ejemplos para un grupo **%wheel**.

Para dar acceso a **roberto** en todos los host al comando **date**, utilice

```
robert ALL=/usr/bin/date
```

Si un usuario quiere ver qué comandos puede ejecutar, debe utilizar el comando **sudo** con la opción **-l**.

```
sudo -l
```

Hora y fecha del sistema

Es probable que haya establecido hora y fecha cuando instaló por primera vez su sistema, sin necesidad de hacerlo de nuevo. Sin embargo, si escribió la hora incorrecta o se movió a una zona horaria diferente, podrá configurar hora y fecha del sistema usando el comando de shell **date** o

herramientas de escritorio como Fecha y hora de KDE, o las opciones Fecha y hora de GNOME (time-admin).

Además, muchas distribuciones de Linux ofrecen sus propias utilidades de fecha y hora. Las herramientas del escritorio y distribución brindan una interfaz GUI fácil de usar.

Puede utilizar el comando **date** en su línea de comandos de usuario root para configurar fecha y hora del sistema. Como un argumento para **date**, haga una lista (sin delimitadores) de mes, día, hora y año. En el siguiente ejemplo, se establece la fecha 2:59 P.M., 6 de abril de 2008 (04 para abril, 06 para el día, 1459 para la hora y 08 para el año 2008):

```
# date 0406145908
Sun Mar 6 02:59:27 PST 2008-04-25
```

Casi todas las herramientas de fecha y hora también cuentan con una opción para usar servidores de hora y configurar la hora automáticamente. El protocolo de hora de red (NTP, Network Time Protocol) permite a un servidor remoto configurar fecha y hora, en vez de configuraciones locales. NTP permite la sincronización más exacta de su reloj del sistema. Se utiliza a menudo para administrar hora y fecha de sistemas en red, liberando al administrador de sincronizar los relojes manualmente. Puede descargar la documentación y software de NTP actualizados del sitio ntp.org.

Una herramienta de fecha y hora permitirá decidir si habilita NTP o seleccionar el servidor que habrá de utilizarse. Los servidores NTP operan a través de almacenes que seleccionarán, de manera aleatoria, un servidor disponible para incrementar la eficiencia. Es probable que su distribución ya tenga instalado un conjunto de almacenes designados. Si el acceso con un almacén es lento, cambie a otro. Los servidores de almacén pool.ntp.org acceden en todo el mundo. Los almacenes para ubicaciones geográficas específicas se encuentran en el sitio NTP Public Services Project (vínculo Time Servers), en ntp.isc.org. Es probable que un servidor cercano sea más rápido.

Programación de tareas: cron

La programación de tareas regulares de mantenimiento, como la creación de copias de seguridad, se administra en Linux con el servicio **cron**, implementado por un daemon **cron**. Un daemon es un servidor en ejecución continua y que hace revisiones constantes, en espera de ciertas acciones que habrán de emprenderse. Estas tareas se incluyen en el archivo **crontab**. El daemon **cron** revisa constantemente el archivo **crontab** del usuario, para ver si es hora de emprender tales acciones. Cualquier usuario puede configurar un archivo **crontab** por su cuenta. El usuario root puede configurar un archivo **crontab** para emprender acciones administrativas relacionadas con el sistema, como copias de seguridad de archivos a cierta hora, cada semana o mes.

La forma más sencilla de programar tareas es usar la herramienta de escritorio **cron**. En el caso de KDE, se usa la herramienta KCRON y, en el caso de GNOME, GNOME Schedule. Ambos proporcionan paneles para seleccionar mes, fecha y hora de un proceso, aunque deberá escribir el comando que quiera ejecutar manualmente, como si estuviera en la línea de comandos. Una lista de entradas de **cron** permite modificar o eliminar tareas. Si tiene una operación abierta, asegúrese también de programar un comando para apagarla.

El nombre del daemon **cron** es **crond**. En general, se inicia automáticamente junto con su sistema. Configure esta característica usando un servicio de administración como **chkconfig** (Fedora y SUSE), **services-admin** (GNOME) o **sysv-rc-conf**. También es posible iniciar y detener manualmente los servicios de **crond**, que tal vez quiera realizar en caso de mantenimiento de emergencia o durante actualizaciones.

Entradas crontab

Una entrada **crontab** tiene seis campos: los primeros cinco se utilizan para especificar la hora en que se realizará una acción, mientras el último campo es la acción misma. El primer campo especifica los minutos (0-59); el segundo especifica la hora (0-23); el tercero, el día del mes (1-31); el cuarto, el mes del año (1-12, o prefijos de mes como *Jan* y *Sep*); y el quinto, el día de la semana (0-6, o prefijos como *Wed* y *Fri*), empieza con 0 como domingo. En cada campo de hora, se especifica rango, conjunto de valores o usa el asterisco para indicar todos los valores. Por ejemplo, **1-5** para el campo del día de la semana especifica de lunes a viernes. En el campo de hora, **8, 12, 17** especificaría 8 A.M., medio día y 5 P.M. Un ***** en el campo mes del año indica todos los meses. El formato del campo **crontab** es:

```
minuto hora día-del-mes mes día(s)-de-la-semana tarea
```

En el siguiente ejemplo se crea una copia de seguridad del directorio **proyectos** a las 2:00 A.M., todos los días:

```
0 2 * * 1-5 tar cf /home/backp /home/proyectos
```

La misma entrada se presenta aquí nuevamente utilizando prefijos para mes y día(s)-de-la-semana:

```
0 2 * * Mon-Fri tar cf /home/backp /home/proyectos
```

Para especificar meses, días, semanas u horas particulares, haga una lista individual de éstos, separados por comas. Por ejemplo, para realizar la tarea previa en domingo, miércoles y viernes, use **0, 3, 5** en el campo de día(s)-de-la-semana, o sus prefijos equivalentes, **Sun, Wed, Fri**.

```
0 2 * * 0,3,5 tar cf /home/backp /home/proyectos
```

cron también permite comentarios. Un comentario es una línea comenzando con el signo **#**.

```
# Respaldo semanal para los proyectos de Chris
0 2 * * Mon-Fri tar cf /home/backp /home/proyectos
```

Variables de entorno para cron

El servicio **cron** también permite definir variables de entorno para utilizarlas con tareas realizadas. Linux define variables para **SHELL**, **PATH**, **HOME** y **MAILTO**. **SHELL** designa la shell que utilizarán las tareas, en este caso la shell BASH. **PATH** muestra los directorios en que se encuentran programas y secuencias de comandos. En este ejemplo se muestran los directorios estándar, **/usr/bin** y **/bin**, además de los directorios del sistema reservados para aplicaciones del sistema, **/usr/sbin** y **/sbin**. **MAILTO** designa a quién se enviará el correo con el resultado de la tarea. Como opción predeterminada, se envía el correo al usuario que lo programa, pero también puede hacer que se envíe a un usuario específico (por ejemplo, la dirección de correo electrónico del administrador o una cuenta de otro sistema en una red). **HOME** es el directorio de inicio para una tarea, en este caso el directorio más alto.

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
```

El directorio cron.d

En un sistema usado en exceso, el archivo **/etc/crontab** se llenará fácilmente. También pueden existir instancias donde ciertas entradas requieren diferentes variables. Por ejemplo, tal vez necesite

ejecutar tareas bajo una shell diferente. Para ayudar a organizar mejor sus tareas de **crontabs**, se colocan las entradas **crontab** en archivos en el directorio **cron.d**. Todos los archivos de este directorio contienen entradas **crontab**, del mismo formato que **/etc/crontab**. Se les puede dar cualquier nombre. Se tratan como archivos adicionales de **crontab**, que **cron** revisa para ver qué tareas se deben ejecutar. Por ejemplo, Linux instala el archivo **sysstat** en **cron.d** conteniendo entradas de **crontab** para ejecutar herramientas permitiendo recopilar estadísticas del sistema.

El comando crontab

Se utiliza el comando **crontab** para instalar sus entradas en un archivo **crontab**. Para esto, primero debe crear un archivo de texto y escribir sus entrada de **crontab**. Guarde este archivo con cualquier nombre que quiera, como **miarchivocron**. Después, para instalar estas entradas, inserte **crontab** y el nombre del archivo de texto. El comando **crontab** toma su contenido del archivo de texto y crea un archivo **crontab** en el directorio **/var/spool/cron**, agregando el nombre del usuario que empleó el comando. En el siguiente ejemplo, el usuario **root** instala el contenido de **miarchivocron** como el archivo **crontab** del usuario **root**. Esto crea un archivo llamado **/var/spool/cron/juan**. Se controla el uso del comando **crontab** a usuarios regulares con el archivo **/etc/cron.allow**. Sólo los usuarios con sus nombres incluidos pueden crear archivos **crontab** propios. En sentido opuesto, el archivo **/etc/cron.deny** incluye usuarios a los que se niega el uso de la herramienta **cron**, lo que evita programen tareas. Si no existe uno de los archivos, el acceso se niega a todos los usuarios. Si un usuario no está en un archivo **/etc/cron.allow**, se niega el acceso. Sin embargo, si el archivo **/etc/cron.allow** no existe, y sí existe el archivo **/etc/cron.deny**, entonces todos los usuarios que no están en **/etc/cron.deny** se les permite el acceso automáticamente.

```
# crontab miarchivocron
```

Edición en cron

Nunca intente editar su archivo **crontab** directamente. En cambio, utilice el comando **crontab** con la opción **-e**. Esto abre el archivo **crontab** en el directorio **/var/spool/cron**, con un editor de texto estándar, como **Vi** (**crontab** utiliza el editor predeterminado, como especifica la variable de entorno de shell **EDITOR**). Para usar un editor diferente para **crontab**, cambie el editor predeterminado al incluir el nombre del programa del editor en la variable **EDITOR** y exportar esa variable. Generalmente, la variable **EDITOR** se configura en la secuencia de comandos **/etc/profile**. Al ejecutar **crontab** con la opción **-l**, despliega el contenido de su archivo **crontab** y la opción **-r** elimina todo archivo. Al invocar **crontab** con otro archivo de texto de entradas de **crontab** se sobrescribe el archivo **crontab** actual, reemplazándolo con el contenido del archivo de texto.

Organización de tareas programadas

Tiene la opción de organizar tareas administrativas de **cron** en dos grupos generales: tareas administrativas comunes ejecutándose a intervalos regulares o tareas especializadas que necesitan ejecutarse en una hora única. Las tareas únicas se ejecutan como entradas en el archivo **/etc/crontab**, como se describe en la siguiente sección. Las tareas administrativas comunes, aunque se ejecuten desde el archivo **/etc/crontab**, están mejor organizadas en directorios **cron** especializados. En estos directorios, cada tarea se coloca en su propia secuencia de comandos shell invocando la tarea cuando se ejecuta. Por ejemplo, pueden existir varias tareas administrativas que necesitan ejecutarse cada semana, el mismo día, como cuando se programa el mantenimiento del sistema el domingo por la mañana. Para este tipo de tareas, **cron** ofrece varios directorios especializados para tareas automáticas diarias, semanales, mensuales o anuales. Cada una contiene un prefijo **cron** y un sufijo

Comandos y herramientas de cron	Descripción
crontab <i>opciones</i> <i>nombredearchivo</i>	Con <i>nombredearchivo</i> como argumento, se instalan las entradas de crontab en el archivo crontab ; estas entradas son operaciones ejecutándose en horas específicas -e Edita el archivo crontab -l Despliega el contenido del archivo crontab -r Elimina el archivo crontab
Kcron	Herramienta de administración de cron de la interfaz GUI de KDE
Schedule	Herramienta de administración de cron de la interfaz GUI de GNOME
Archivos y directorios de cron	
/etc/crontab	Archivo crontab del sistema, sólo tiene acceso a él el usuario root
/etc/cron.d	Directorio conteniendo varios archivos crontab , sólo root tiene acceso a él
/etc/cron.hourly	Directorio para tareas realizadas cada hora
/etc/cron.daily	Directorio para tareas realizadas a diario
/etc/cron.weekly	Directorio para tareas realizadas semanalmente
/etc/cron.monthly	Directorio para tareas realizadas mensualmente
/etc/cron.yearly	Directorio para tareas realizadas anualmente
/etc/cron.allow	Usuarios a los que se permite enviar tareas de cron
/etc/cron.deny	Usuarios a los no que se permite el acceso a cron

TABLA 27-2 Comandos, herramientas, archivos y directorios de cron

para el intervalo. El directorio **/etc/cron.daily** se usa para tareas realizadas a diario, mientras las tareas semanales se colocan en el directorio **/etc/cron.weekly**. Los directorios de **cron** se muestran en la tabla 27-2.

Ejecución de secuencias de comandos de directorio de cron

Cada directorio contiene secuencias de comandos y todas se ejecutan simultáneamente. La programación para cada grupo se determina con una entrada en el archivo **/etc/crontab**. La ejecución real de secuencias de comandos se realiza por secuencia de comandos **/usr/bin/run-parts**, que ejecuta todas las secuencias de comandos y programas de un directorio determinado. La programación de todas las tareas de un directorio específico se maneja con una entrada en el archivo **/etc/crontab**. Linux ofrece entradas con horas designadas, que puede cambiar de acuerdo con sus necesidades. Aquí se muestra el archivo **crontab**, con horas para ejecutar las secuencias de comandos en diferentes directorios **cron**. Aquí se ve que la mayoría de secuencias de comandos se ejecutan a las 4 A.M., ya sea diario (4:02), el domingo (4:22) o el primer día de cada mes (4:42). Las secuencias de comandos por hora se ejecutan un minuto después de la hora.

```
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
```

SUGERENCIA Las secuencias de comandos en el directorio **cron** se ejecutan alfabéticamente. Si necesita que cierta secuencia de comandos se ejecute antes que otras, quizás deba modificar su nombre. Un método consiste en poner un número antes del nombre. Por ejemplo, en el directorio **/etc/cron.weekly**, la secuencia de comandos **anacron** cambiaría por **0anacron** para ejecutarse antes que las demás.

Tenga en cuenta que se trata simplemente de directorios conteniendo archivos ejecutables. La programación real se realiza mediante entradas en el archivo **/etc/crontab**. Por ejemplo, si el campo de semana en la entrada **cron.weekly crontab** cambia a ***** en vez de **0**, y el campo mes se cambia a **1** (**22 4 1 * *** en vez de **22 4 * * 0**), las tareas del archivo **cron.weekly** se ejecutarán mensual y no semanalmente.

```
* 12 * * 3 root run-parts /etc/cron.misdocs
```

Anacron

Para un sistema que se apagará, generalmente, durante las horas en que es probable **cron** se esté ejecutando, tal vez quiera complementar **cron** con **anacron**, que sólo se activa cuando necesitan ejecutarse tareas de programación. Por ejemplo, si un sistema se apaga en un fin de semana, cuando los trabajos de **cron** están programados, entonces éstos no se realizarán; sin embargo, **anacron**, revisa si hay trabajos que deben realizarse cuando el sistema se enciende de nuevo y después los ejecuta. Está diseñado sólo para trabajos de ejecución diaria o semanal.

En el caso de trabajos de **anacron**, coloque las entradas de **crontab** en el archivo **/etc/anacrontab**. Para cada tarea programada, se especifica el número de días intermedios entre ejecuciones (7 es semanal, 30 es mensual), la hora del día en que se ejecuta (enumerado en minutos), una descripción de la tarea y el comando a ejecutar. En el caso de copias de seguridad, el comando usado es la operación **tar**.

Niveles de ejecución de sistema: telinit, initab y shutdown

Un sistema de Linux se ejecuta en diferentes niveles, dependiendo de las capacidades que quiera darle. Por ejemplo, para ejecutar su sistema a nivel administrativo, bloqueando cualquier acceso de usuario. Las operaciones completas normales se activan sólo al ejecutar su sistema en cierto nivel de capacidad operacional, como el soporte a acceso de varios usuarios o interfaces gráficas. A estos niveles (también conocidos como estados o modos) se les denomina *niveles de ejecución*, representando el nivel de soporte que se está ejecutando en su sistema.

Niveles de ejecución

Un sistema Linux tiene varios niveles de ejecución, numerados de 0 a 6. Cuando enciende su sistema, ingresa en el nivel de ejecución predeterminado. 0, 1 y 6 son niveles de ejecución especiales realizando funciones específicas. El nivel de ejecución 0 es el estado de apagado y se invoca con el comando **halt**, para apagar el sistema. El nivel de ejecución 6 es el estado de reinicio (apaga el

sistema y lo reinicia). El nivel de ejecución 1 es el estado de un solo usuario, que sólo permite el acceso al superusuario y no ejecuta servicio de red. Le permite, como administrador, realizar acciones administrativas sin que otros interfieran.

Otros niveles de ejecución reflejan cómo quiere se utilice el sistema. Éstos pueden diferir, dependiendo de la distribución que tenga. Los niveles de ejecución descritos aquí se utilizan en Red Hat, Fedora, SUSE y distribuciones similares. Debian y Ubuntu tienen una configuración de niveles de ejecución un poco diferente; utilizan el nivel de ejecución 2, para inicios de sesión gráficos y el resto como definidos por el usuario.

En Red Hat, Fedora y distribuciones similares, el nivel de ejecución 2 es un estado multiusuario parcial, permitiendo el acceso a varios usuarios, pero sin servicios de red como NFS o **xinetd** (el daemon de servicios extendidos de Internet). Este nivel resulta útil para un sistema individual que no es parte de una red. Los niveles 3 y 5 ejecutan un sistema de Linux totalmente operacional, con soporte a multiusuarios y acceso a archivos remotos compartidos. La diferencia entre ellos está en la interfaz que ejecutan. Los niveles de ejecución 3 inician su sistema con la interfaz de línea de comandos (también conocida como interfaz de modo de texto). El nivel de ejecución 5 inicia su sistema con una sesión X, ejecutando el servidor X Window System e invocando un inicio de sesión gráfico, que utiliza administradores de despliegue como **gdm** o **xdm**. Si selecciona se usen inicios de sesión gráficos durante la instalación, el nivel de ejecución 5 será el predeterminado. Linux proporciona dos secuencias de teclado permitiéndole cambiar entre ambas en una sesión: CTRL-ALT-F1 cambia de la interfaz gráfica (nivel de ejecución 5), a la de línea de comandos (nivel de ejecución 3) y CTRL-ALT-F7 cambia de la interfaz de línea de comandos a la gráfica. Los niveles de ejecución se muestran en la tabla 27-3.

Niveles de ejecución (estados)	Descripción
0	Halt (no configure este nivel como opción predeterminada), apaga el sistema por completo.
1	Modo de un solo usuario administrativo; niega el acceso a otros usuarios al sistema pero permite el acceso de root a todo el sistema de archivos multiusuario. No se ejecutan las secuencias de comandos de inicio. (Utilice s o S para entrar en el modo de un solo usuario con las secuencias de comandos de inicio en ejecución).
2	Multiusuario, sin servicios de red como NFS, xinetd y NIS (igual a 3 pero sin red).
3	Modo multiusuario completo, con inicio de sesión en la interfaz de línea de comandos; permite a los usuarios remotos compartir con otros sistemas en su red. También conocido como <i>estado de modo de texto</i> .
4	No se usa.
5	Modo multiusuario completo que inicia en una sesión X, empezando un inicio de sesión gráfico; permite compartir archivos remotos con otros sistemas de su red (igual a 3, pero con inicio de sesión gráfico). En Ubuntu, éste es el nivel de ejecución 2.
6	Reinicio; apaga y reinicia el sistema (no configure este nivel como predeterminado).

TABLA 27-3 Niveles de ejecución de sistema en Red Hat, Fedora, SUSE y distribuciones similares

Resulta útil cambiar los niveles de ejecución, si tiene problemas en un nivel de ejecución particular. Por ejemplo, si su tarjeta de video no se instaló apropiadamente, entonces fallará cualquier intento de iniciar en un nivel de ejecución 5, porque este inicia de inmediato la interfaz gráfica. En cambio, debe utilizar la interfaz de línea de comandos, nivel de ejecución 3, para corregir la instalación de su tarjeta de video.

SUGERENCIA Tiene la opción de utilizar el nivel de ejecución de un solo usuario (1), como un estado de modo de recuperación, que permite iniciar su sistema sin ejecutar secuencias de comandos de inicio para servicios como DNS. Esto resulta útil si su sistema tiene problemas cuando intenta iniciar esos servicios. Las conexiones de red están deshabilitadas, al igual que cualquier acceso de multiusuario. También se utiliza **linux -s** como indicador de comandos de inicio, para insertar el nivel de ejecución 1. Si quiere ingresar en el estado de un solo usuario y también ejecutar las secuencias de comandos de inicio, recurra al nivel de ejecución **s** o **S**.

Niveles de ejecución en initab

Cuando su sistema inicia sesión, utiliza el nivel de ejecución predeterminado como se especifica en la entrada **init** predeterminada, en el archivo **/etc/inittab**. Por ejemplo, si su nivel de ejecución **init** predeterminado es 5 (el inicio de sesión gráfico), la entrada predeterminada **init** en el archivo **/etc/inittab** será

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

Puede cambiar el nivel de ejecución predeterminado al editar el archivo **/etc/inittab** y cambiar la entrada predeterminada **init**. Resulta peligroso editar el archivo **/etc/inittab**. Debe hacerlo con gran cuidado. Como ejemplo, si el nivel de ejecución predeterminado es 3 (la línea de comandos), la entrada para su nivel de ejecución predeterminado en el archivo **/etc/inittab** deberá verse así:

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

Puede cambiar de 3 a 5 para ir de su nivel de ejecución predeterminado, de interfaz de línea de comandos (3), al de inicio de sesión gráfico (5). Cambie sólo este número y nada más.

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

SUGERENCIA Si su archivo **letclinitab** se corrompe, reinicie e inserte **linux single** en el indicador de comandos de inicio para arrancar su sistema, omitiendo el archivo **inittab**. Luego podrá editar el archivo para corregirlo.

Cambio de niveles de ejecución con telinit

Sin importar el nivel de ejecución en que inicie, puede cambiar de un nivel de ejecución a otro con el comando **telinit**. Si su nivel predeterminado es 3, se inicia en el nivel de ejecución 3, pero puede cambiar al nivel de ejecución 5, por ejemplo, con **telinit 5**. El comando **telinit 0** apaga su sistema. En el siguiente ejemplo, el comando **telinit** cambia al nivel de ejecución 1, el estado administrativo:

```
# telinit 1
```

En Red Hat, Fedora, SUSE y distribuciones similares, un uso común de **telinit** es cambiar niveles de ejecución cuando necesita instalar un paquete de software requiriendo que el servidor X se apague. Este es el caso de controladores gráficos obtenidos directamente de Nvidia o ATI. Primero

debe cambiar al nivel de ejecución 3 con el comando **telinit 3**, apagar el servidor X y después instalar el controlador gráfico.

```
telinit 3
```

Después de la instalación, se regresa al servidor X y su interfaz GUI con el comando **telinit 5**.

```
telinit 5
```

Tenga en cuenta que es preferible usar paquetes de distribución para Nvidia y ATI, que los obtenidos directamente de estos dos comercializadores.

NOTA En Debian, Ubuntu y distribuciones similares, la versión de escritorio invoca el servidor X en todos los niveles de ejecución principales. El uso de **telinit** para cambiar a otro nivel de ejecución no le permitirá apagar el servidor X. Esto requiere apagar los administradores de despliegue (su pantalla de inicio de sesión), al ejecutar la secuencia de comandos **gdm** (GNOME) o **kdm** (KDE) con la opción **stop**. Utilice el comando **sudo letclinit.d/gdm stop** para apagar el servidor X.

El comando **telinit** es, en realidad, un vínculo simbólico (otro nombre para un comando) con el comando **init**. Este comando realiza las operaciones de inicio reales y se invoca automáticamente al iniciar su sistema. Aunque puede utilizar **init** para cambiar los niveles de ejecución, es mejor utilizar **telinit**. Cuando se invoca **telinit**, **init** simplemente cambia los niveles de ejecución.

Para utilizar **init**, inserte el comando **init** y el número de nivel de ejecución en una línea de comandos. Si está en el nivel de ejecución 3 (la línea de comandos), el comando a continuación lo coloca en el 5 (la interfaz gráfica).

```
init 5
```

El comando runlevel

Utilice el comando **runlevel** para ver en que estado se está ejecutando. Despliega el estado previo, seguido por el actual. Si no ha cambiado de estado, el previo se mostrará como N, indicando que no existe estado previo. Es así para el estado en que inició. En el siguiente ejemplo, el sistema se ejecuta en el estado 3, sin cambio de estado previo.

```
# runlevel
N 3
```

Apagado

Aunque puede apagar el sistema con el comando **telinit** y el estado 0, también puede usar el comando **shutdown**. Éste tiene un argumento de hora que da a los usuarios del sistema una advertencia antes de apagar. Tiene la opción de especificar la hora exacta en que se apagará o un periodo de minutos a partir de la hora actual. La hora exacta se especifica con *hh:mm* para horas y minutos. El periodo se indica con **+** y el número de minutos. El comando **shutdown** toma varias opciones con las que especifica cómo quiere que el sistema se apague. La opción **-h**, que significa halt (detener), sólo apaga el sistema, mientras la opción **-r** apaga el sistema y reinicia. En el siguiente ejemplo, el sistema se apaga después de diez minutos:

```
# shutdown -h +10
```

Para apagar el sistema de inmediato, utilice **+0** o la palabra **now**. En el siguiente ejemplo se apaga el sistema inmediatamente y después se reinicia:

```
# shutdown -r now
```

Con el comando **shutdown**, se incluye un mensaje de advertencia enviado a todos los usuarios que hayan iniciado, dándoles tiempo para terminar lo que hacen antes de apagar. He aquí un ejemplo

```
# shutdown -h +5 "El sistema necesita descansar"
```

Si no especifica la opción **-h** o **-r**, el comando **shutdown** apaga el modo multiusuario y lo cambia al modo administrativo de un solo usuario. En efecto, su estado de sistema cambia de 3 (estado multiusuario) a 1 (estado de un solo usuario administrativo). Sólo el usuario root queda activo, que permite al usuario root realizar cualquier operación administrativa del sistema con la que otros usuarios tal vez interferirían.

SUGERENCIA También puede apagar el sistema desde los escritorios de GNOME o KDE.

Las opciones de shutdown se muestran en la tabla 27-4.

Comando	Descripción
shutdown [-rkhncft] hora [mensaje-advertencia]	Apaga el sistema después del periodo especificado, envía advertencias a los usuarios; se especifica un mensaje de advertencia propio después del argumento hora; si no se especifica -h o -r para apagar el sistema, el sistema establece el modo administrativo, estado de nivel de ejecución 1.
Argumento	
Hora	Tiene dos formatos posibles: una hora absoluta en el formato <i>hh:mm</i> , con <i>hh</i> como la hora (uno o dos dígitos) y <i>mm</i> como los minutos (en dos dígitos), o el formato <i>+m</i> , con <i>m</i> como el número de minutos que habrá de esperar; la palabra now es un alias de +0.
Opciones	
-t seg	Le indica a init espere seg segundos entre el envío de la advertencia y el apagado de las señales, antes de cambiar a otro nivel de ejecución.
-k	No apaga realmente el sistema; sólo envía el mensaje de advertencia a todos.
-r	Reinicia después de apagarse, nivel de ejecución 6.
-h	Se suspende tras apagarse, nivel de ejecución 0.
-n	No llama a init para apagar; usted lo hace.
-f	Omite la revisión del sistema de archivos (fsck) al reiniciar.
-c	Cancela un apagado en ejecución; no hay argumento de hora.

TABLA 27-4 Opciones de apagado de sistema

NOTA Puede seleccionar ciertos servicios para su ejecución y el nivel en que se ejecutarán. Casi todos los servicios son servidores, como un servidor Web o proxy. Otros servicios proporcionan seguridad, como SSH o Kerberos. La mayor parte de distribuciones tienen sus propias herramientas para administrar servidores. Dos de las más conocidas son **chkconfig** (Fedora y SUSE) y **sysv-rc-conf**. Algunas herramientas como **services-admin** y **rcconf** activan o desactivan servicios para niveles de ejecución predeterminados.

Directorios del sistema

Su sistema de archivos de Linux se organiza en directorios cuyos archivos se utilizan para diferentes funciones del sistema (consulte la tabla 27-5). Para una administración básica del sistema, debe familiarizarse con directorios de programas del sistema donde se almacenan las aplicaciones, el directorio de configuración del sistema (**/etc**), donde se colocan casi todos los archivos de configuración y el directorio de registros del sistema (**/var/log**), almacenando los registros del sistema que indican la actividad de su sistema. Otros directorios del sistema se cubren en sus capítulos respectivos, muchos se analizan en el capítulo 29.

Directorio	Descripción
/bin	Programas relacionados con el sistema.
/sbin	Programas de sistema para tareas especializadas.
/lib	Bibliotecas de sistema y aplicaciones.
/etc	Archivos de configuración para servicios y aplicaciones de la red y el sistema.
/home	Directorios de inicio y de datos de servidor del usuario, como archivos de sitios Web y FTP.
/mnt	Donde se montan los sistemas de archivos de CD-ROM y discos flexibles (capítulo 29).
/var	Directorios de sistema cuyos archivos cambian continuamente, como registros, archivos spool de impresora y de bloqueo (capítulo 29).
/usr	Programas y archivos relacionados con el usuario; incluye varios subdirectorios clave, como /usr/bin , /usr/X11 y /usr/doc .
/usr/bin	Programas para usuarios.
/dev	Directorio generado dinámicamente para archivos de dispositivo (capítulo 31).
/etc/X11	Archivos de configuración de X Window System.
/usr/share	Archivos compartidos.
/usr/share/doc	Documentación para aplicaciones.
/tmp	Directorio de archivos temporales del sistema.
/var/log	Directorio de registros.
/var/log/	Registros del sistema generados por syslogd .
/var/log/audit	Registros de audit generados por auditd .

TABLA 27-5 Directorios del sistema

Directorios de programas

Los directorios con “bin” en el nombre, se emplean para almacenar programas. El directorio **/bin** almacena programas básicos de usuario, como las shell de inicio de sesión (BASH, TCSH, y ZSH) y comandos de archivos (**cp**, **mv**, **rm**, **ln**, etc.). El directorio **/sbin** almacena programas de sistema especializados para tareas como administración de sistemas de archivos (**fsck**, **fdisk**, **mkfs**) y operaciones de sistema como apagar y reiniciar (**init**). El directorio **/usr/bin** almacena archivos de programa diseñados para tareas de usuario. El directorio **/usr/sbin** almacena operaciones del sistema relacionadas con usuarios, como **useradd** para agregar nuevos usuarios. El directorio **/lib** almacena todas las bibliotecas utilizadas en su sistema, incluida la principal de Linux, **libc** y los subdirectorios como **modules**, almacenando todos los módulos de kernel actuales.

Directorios y archivos configuración

Cuando se configuran diferentes elementos de su sistema, como usuarios, aplicaciones, servidores o conexiones de red, se utilizan archivos de configuración mantenidos en ciertos directorios del sistema. Los archivos de configuración se colocan en el directorio **/etc**.

Archivos de configuración: **/etc**

El directorio **/etc** almacena archivos de configuración de su sistema, red, servidor y aplicaciones. Aquí se encuentra un archivo **fstab** incluyendo sus sistemas de archivos, el archivo **hosts**, con las direcciones IP de los hosts de su sistema y **/etc/profile**, el archivo de configuración predeterminado de la shell de BASH para todo el sistema. Este directorio incluye varios subdirectorios, como **/etc/apache** para los archivos de configuración del servidor Web Apache, **/etc/X11** para los archivos de configuración de X Window System y el administrador de ventanas y **/etc/udev**, para generar reglas de archivos de dispositivos en **/dev**. Tiene la opción de configurar muchas aplicaciones y servicios editando directamente sus archivos de configuración, aunque sea mejor utilizar una herramienta de administración correspondiente. En la tabla 27-6 se muestra una lista de varios archivos de configuración de uso común en el directorio **/etc**.

Registros de sistema: **/var/log** y **syslogd**

Varios registros de sistema relacionados con tareas realizadas en su sistema se almacenan en el directorio **/var/log**. Aquí se encuentran registros para correo, noticias y todas las demás operaciones de sistema, como los registros del servidor Web. El archivo **/var/log/messages** es un registro de todas las tareas de sistema cubiertas por otros registros. Esto suele incluir tareas de inicio, como cargar controladores y montar sistemas de archivos. Si falla la carga del controlador de una tarjeta al inicio, encontrará allí el mensaje de error correspondiente. Los inicios de sesión también se registran en este archivo, que muestra quién trató de iniciar sesión y en qué cuenta. El archivo **/var/log/maillog** registra transmisiones de mensajes de correo y transferencias de noticias.

NOTA Para ver los registros, utilice el GNOME System Log Viewer.

syslogd y **syslog.conf**

El daemon **syslogd** administra todos los registros de su sistema y se coordina con cualquier operación de inicio de sesión de otros sistemas de su red. La información de configuración para **syslogd** se almacena en el archivo **/etc/syslog.conf**, conteniendo nombres y ubicaciones de sus archivos de registro de sistema. Aquí se encuentran entradas para **/var/log/messages** y **/var/log/**

Archivo	Descripción
/etc/bashrc	El archivo de configuración predeterminado de la shell Bash
/etc/group	Una lista de grupos con configuraciones para cada uno
/etc/fstab	Monta sistemas de archivos automáticamente cuando inicia su sistema
/boot/grub/menu.lst	El archivo de configuración GRUB para el Gestor de arranque de GRUB (vinculado a /etc/grub.conf en Red Hat)
/etc/inittab	Configura el estado predeterminado, además de las conexiones de terminales
/etc/profile	Archivo de configuración predeterminado de la shell para usuarios
/etc/modprobe.conf	Módulos de su sistema que se cargarán automáticamente
/etc/motd	Mensaje del día del administrador del sistema
/etc/mtab	Sistemas de archivos montados actualmente
/etc/passwd	Configuraciones de contraseña e inicio de sesión de usuario
/etc/services	Servicios ejecutándose en el sistema y puertos que utilizan
/etc/shadow	Contraseñas cifradas de usuario
/etc/shells	Shells instaladas en su sistema, utilizadas por los usuarios
/etc/sudoers	Configuración sudo para controlar el acceso administrativo
/etc/termcap	Lista de especificaciones de tipo de terminal para terminales que pueden estar conectadas al sistema
/etc/xinetd.conf	Configuración de servidor Xinetd
Directorio	
/etc/cron	Secuencias de comandos de cron
/etc/cups	Archivos de configuración de impresora CUPS
/etc/init.d	Secuencias de comandos de servicio para distribuciones soportando secuencias de comandos de SysV Init
/etc/mail	Archivos de configuración de sendmail
/etc/openldap	Configuración para el servidor Open LDAP
/etc/rc.d	Secuencias de comandos de inicio para diferentes niveles de ejecución
/etc/skel	Versiones de archivos de inicialización, como .bash_profile , copiadas a directorios de inicio del nuevo usuario
/etc/X11	Archivos de configuración de X Window System
/etc/xinetd.d	Secuencias de configuración para servicios administrados por el servidor xinetd
/etc/udev	Reglas para generar dispositivos (capítulo 31)
/etc/hal	Reglas para generar dispositivos extraíbles (capítulo 31)

TABLA 27-6 Archivos y directorios de configuración de sistema comunes

maillog, entre otras. Siempre que haga cambios en el archivo **syslog.conf**, necesita reiniciar el daemon **syslogd**.

Entradas en syslog.conf

Una entrada en **syslog.conf** consta de dos campos: un *selector* y una *acción*. El selector es el tipo de servicios registrados, como correo o noticias, mientras la acción es la ubicación donde se coloca el mensaje. La acción suele ser un archivo de registro, pero también es un host remoto o canalización a otro programa. Al tipo de servicio se le conoce como *utilería*. El daemon **syslogd** tiene varios términos usados para especificar ciertos tipos de servicios (consulte la tabla 27-7). Una utilidad

Utilerías	Descripción
authpriv	Mensajes de autorización y seguridad (privados)
cron	Mensajes de daemon de reloj (cron y en)
daemon	Otros mensajes del daemon del sistema
kern	Mensajes del kernel
lpr	Mensajes del subsistema de impresora de línea
mail	Mensajes del subsistema de correo
mark	Sólo uso interno
news	Mensajes del subsistema de noticias Usenet
syslog	Mensajes internos de syslog
user	Mensajes genéricos en el nivel de usuario
uucp	Mensajes de subsistema UUCP
local0 a local7	Reservados para uso local
Prioridades	Descripción
debug	7, depuración de mensajes, la prioridad más baja
info	6, mensajes informativos
notice	55, notificaciones, normal, pero con condición significativa
warning	4, advertencias
err	3, mensajes de error
crit	2, condiciones críticas
alert	1, alertas, deben tomarse acciones inmediatas
emerg	0, mensajes de emergencia, el sistema no se puede usar, la prioridad más alta
Operadores	Descripción
*	Relaciona todas las utilerías o prioridades de un sector
=	Restringe a una prioridad específica
!	Excluye la prioridad especificada y las más altas
/	Un archivo en que se guardan mensajes
@	Un host al que se envían mensajes
 	Una canalización FIFO a la que se envían mensajes

TABLA 27-7 Utilerías, prioridades y operadores de syslogd

puede calificarse aún más por prioridad. Una *prioridad* especifica el tipo de mensaje que genera la utilidad; **syslogd** usa varios términos designados para indicar diferentes propiedades. Un *sector* está integrado por una utilidad y una prioridad, separados por un punto. Por ejemplo, para guardar mensajes de error generados por sistemas de correo, se usa un sector constando de la utilidad **mail** y la prioridad **err**, como se muestra aquí:

```
mail.err
```

Para guardar estos mensajes en el archivo **/var/log/maillog**, especifique ese archivo como la acción, dando la siguiente entrada:

```
mail.err /var/log/maillog
```

El daemon **syslogd** también soporta el uso de ***** como carácter para buscar coincidencias en todas las instalaciones o todas las prioridades en un sector: **cron.*** busca coincidencias en todos los mensajes de **cron**, sin importar la prioridad; ***.err** busca coincidencias en mensajes de error de todas las instalaciones y ***.*** las busca en todos los mensajes. En el siguiente ejemplo se guardan todos los mensajes en el archivo **/var/log/maillog** y todos los mensajes críticos en el archivo **/var/log/mycritical**:

```
mail.* /var/log/maillog
*.crit /var/log/mycritical
```

Prioridades

Cuando se especifica la prioridad de una utilidad, también se incluyen todos los mensajes con mayor prioridad. De esta manera, la prioridad **err** también incluye prioridades **crit**, **alert** y **emerg**. Si sólo quiere seleccionar el mensaje de una prioridad específica, califique la prioridad con el operador **=**. Por ejemplo, **mail.=err** elegirá sólo mensajes de error, no mensajes **crit**, **alert** y **emerg**. También se restringen prioridades con el operador **!**. Esto eliminará mensajes con la prioridad especificada y mayor. Por ejemplo, **mail.!crit** excluirá mensajes **crit**, además de mensajes más elevados **alert** y **emerg**. Para ejecutar específicamente todos los mensajes para una utilidad completa, se usa la prioridad **none**; por ejemplo, **mail.none** excluye todos los mensajes de correo. Esto suele utilizarse cuando está definiendo varios sectores en la misma entrada.

Puede desplegar varias prioridades o utilidades en un sector dado, al separarlas con comas. También se pueden tener varios sectores en la misma entrada, separándolos con puntos y comas. En el primer ejemplo se guardan en el archivo **/var/log/messages** todos los mensajes con la prioridad **info**, excluyendo todos los mensajes de correo y autenticación (**authpriv**). En el segundo se guardan todos los mensajes **crit** y mayores para utilidades **uucp** y **news** en el archivo **/var/log/spooler**:

```
*.info;mail.none;news.none;authpriv.none /var/log/messages
uucp,news.crit /var/logspooler
```

Acciones y usuarios

En el campo de acciones, se especifican archivos, sistemas remotos, usuarios o canalizaciones. Una entrada de acción para un archivo siempre debe comenzar con **/** y debe especificarse su nombre de ruta completo, como **/var/log/messages**. En el caso de mensajes de registro de un host remoto, sólo especifique el nombre de host precedido con un signo **@**. En el siguiente ejemplo se guardan todos los mensajes del kernel **conejo.pista.com**:

```
kern.* @conejo.pista.com
```

Para enviar mensajes a los usuarios, se incluyen sus nombres de inicio de sesión. En el siguiente ejemplo, se enviarán mensajes de noticias críticas a las consolas de los usuarios **carlos** y **alicia**:

```
news.=crit carlos,alicia
```

También puede dar salida a mensajes hacia una canalización con nombre (FIFO). La entrada de canalización para el campo de acción comienza con una `|`. En el siguiente ejemplo se canalizan los mensajes de depuración de kernel a la canalización con nombre `|/usr/adm/debug`:

```
kern.=debug |/usr/adm/debug
```

Un ejemplo para `/etc/syslog.conf`

Aquí se muestra el archivo `/etc/syslog.conf` predeterminado. Los mensajes se registran en varios archivos en el directorio `/var/log`.

`/etc/syslog.conf`

```
# Log all kernel messages to the console.
#kern.*                                /dev/console
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;news.none;authpriv.none;cron.none    /var/log/messages

# The authpriv file has restricted access.
authpriv.*        /var/log/secure
# Log all the mail messages in one place.
mail.*            /var/log/maillog
# Log cron stuff.
cron.*            /var/log/cron
# Everybody gets emergency messages
*.emerg           *
# Save mail and news errors of level err and higher in a special file.
uucp,news.crit    /var/log/spooler
# Save boot messages also to boot.log
local7.*          /var/log/boot.log
# INN
news.=crit        /var/log/news/news.crit
news.=err         /var/log/news/news.err
news.notice       /var/log/news/news.notice
```

Sistema de auditoría de Linux: `auditd`

El sistema de auditoría de Linux proporciona una auditoría de llamada al sistema. Un servidor llamado **audit** realiza la auditoría y los registros se guardan en el directorio `/var/log/audit`. Está diseñado para ser un complemento de SELinux, guardando sus mensajes en el registro **auditd** del archivo `/var/log/audit/audit.log`. El servicio de registro de audit ofrece registros especializados de servicios, como SELinux. Los registros se ubican en `/var/log/audit`. Para refinar la auditoría, se crean reglas audit para revisar ciertas llamadas a sistemas, como las generadas por usuarios o grupos específicos.

La configuración para **auditd** se ubica en los archivos `/etc/auditd.conf` y `/etc/sysconfig/auditd`. La configuración principal se maneja con `/etc/auditd.conf`, almacenando opciones como nombre de

archivo de registro, formato de registro, tamaño máximo de archivos de registro y acciones que se toman cuando el espacio en disco disminuye. Consulte la página Man de **auditd.conf** para conocer información más detallada acerca de todas las opciones. El archivo **/etc/sysconfig/auditd** configura opciones de inicio de servidor y ubicaciones locales como **en_US**.

El paquete de auditoría incluye el servidor **auditd** y tres comandos: **autrace**, **ausearch** y **auditctl**. Se usa **ausearch** para consultar registros de audit. Puede buscar por varios ID; por proceso, usuario, grupo o evento, también por nombre de archivo o incluso hora o fecha. Revise las páginas Man de **ausearch** para conocer una lista completa. **autrace** es una herramienta especializada que permite rastrear procesos específicos. Opera de manera similar a **strace**, registrando las llamadas de sistema y acciones de procesos particulares.

Tiene la opción de controlar el comportamiento del servidor **auditd** con la herramienta **auditctl**. Con ella, se activa o desactiva la auditoría, se revisa el estado y agregan reglas de auditoría para eventos específicos. Revise la página Man de **auditctl** para conocer una descripción detallada.

Las reglas de auditoría se organizan en listas predeterminadas con un conjunto específico de acciones para llamadas al sistema. Actualmente hay tres listas: **task**, **entry** y **exit**, además de tres acciones: **never**, **always** y **possible**. Cuando agrega una regla, lista y acción se ponen en pares, separados por una coma, como en:

```
exit,always
```

Para agregar una regla se usa la opción **-a**. Con la opción **-S** se especifica una llamada a un sistema particular y con **-F** se especifica un campo. Existen varios campos posibles, como **loginuid** (ID de inicio de sesión de usuario), **pid** (ID de proceso) y **exit** (valor de salida de la llamada al sistema). En el caso de un campo, se especifica un valor, como **loginuid=510** para el usuario con un ID de inicio de sesión de usuario de 510. La siguiente regla, como se describe en la documentación, revisa todos los archivos abiertos en busca de un usuario particular:

```
auditctl -a exit,always -S open -F loginuid=510
```

Coloque las reglas que quiera se carguen automáticamente en el archivo **/etc/auditd.rules**. El archivo **sample.rules** en el directorio **/usr/share/doc/auditd***, presenta ejemplos de reglas. También se crea un archivo específico de reglas de auditoría y se utiliza **auditctl** con la opción **-R** para leerlas.

Herramientas y procesos de análisis de rendimiento

Linux trata cada tarea realizada en su sistema como un proceso, al que se asignan nombre y número. Tiene la opción de examinar estos procesos e incluso detenerlos. Linux ofrece varias herramientas para examinar procesos, además del rendimiento de su sistema. El Monitor del sistema de GNOME proporciona un monitoreo rápido. Hay otras herramientas disponibles, como **GKrellM** y **KSysguard**.

Varias utilerías de su sistema ofrecen información detallada de sus procesos, junto con otra información del sistema como uso de CPU y disco (consulte la tabla 27-8). Aunque estas herramientas fueron diseñadas para utilizarse en la línea de comandos de shell, que desplegaba la salida en líneas de texto, ahora varias tienen versiones para KDE y GNOME, brindando una interfaz GUI para desplegar resultados y administrar procesos.

Herramienta de rendimiento	Descripción
vmstat	Rendimiento de componentes de sistema
top	Lista de los procesos que consumen más CPU
free	Lista de memoria RAM libre
sar	Información de actividad de sistema
iostat	Uso de disco
Monitor del sistema de GNOME	Monitor de sistema para procesos y monitoreo de uso
GKrellM	Herramienta de monitoreo apilable, flexible y expansible desplegando información de gran variedad de operaciones de sistema, red y almacenamiento, además servicios; se configura fácilmente con temas.
Administrador de tareas y monitor de rendimiento de KDE	Monitor de sistema de KDE para procesos y monitoreo de uso
Frysk	Herramienta de monitoreo para procesos de sistema
System Tap	Herramienta para analizar cuellos de botella de rendimiento
Gestor de energía de GNOME	Administra características de eficiencia del uso de energía en su sistema
cpuspeed	Implementa la reducción de velocidad de la CPU durante momentos de inactividad (Cool y Quiet de AMD).

TABLA 27-8 Herramientas de rendimiento

Monitor del sistema de GNOME

El Monitor del sistema de GNOME despliega información del sistema y monitorea procesos de éste. Existen cuatro paneles, Sistema, Procesos, Recursos y Sistemas de archivos (véase la figura 27-1). El panel Recursos despliega gráficas para CPU, memoria y memoria de intercambio, además de uso de la red. Su panel Sistemas de archivos despliega sistemas de archivo, dónde están montados y sus tipos, además de la cantidad de espacio en disco utilizado y libre. El panel Procesos muestra la actividad, lo que permite ordenar y buscar procesos. Se utilizan botones de campo para ordenar por nombre, ID de proceso, usuario y memoria. El menú emergente Ver permite seleccionar todos los procesos, mis procesos o procesos activos. Se puede detener cualquier proceso de manera sencilla con sólo seleccionarlo y hacer clic en el botón Finalizar proceso. Al hacer clic con el botón derecho en un elemento, se despliegan las acciones que puede emprender con ese proceso, como detenerlo u ocultarlo. Mapas de memoria, que selecciona en el menú Ver, muestra información en memoria virtual, inodos y marcas.

El comando ps

En la línea de comandos, puede utilizar el comando **ps** para ver una lista de procesos. Con la opción **-aux**, se despliegan todos los procesos. Canalizar la salida a un comando **grep** con un patrón permite buscar un proceso en particular. Una canalización entunela la salida de un comando anterior, como entrada del siguiente comando. Este comando muestra una lista de todos los procesos de X Window System:

```
ps -aux | grep 'X'
```

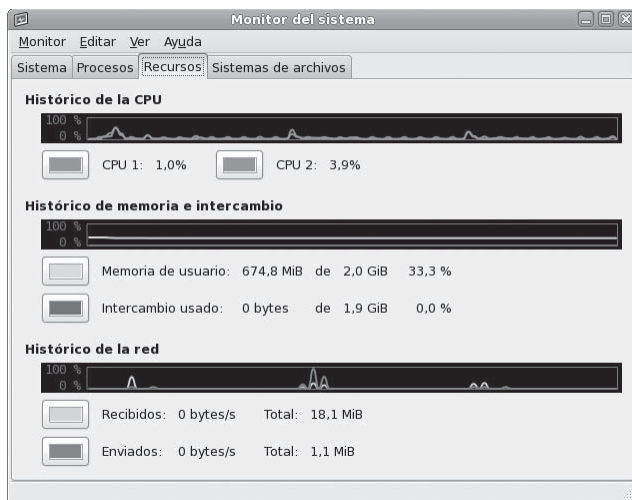


FIGURA 27-1 Monitor del sistema de GNOME

vmstat, top, free, Xload, iostat y sar

El comando **vmstat** da salida a una lista detallada indicando el rendimiento de diferentes componentes del sistema, incluidos CPU, memoria, E/S y operaciones de intercambio. Se genera un informe como una línea con campos para diferentes componentes. Si proporciona un periodo como argumento, se repite en un intervalo especificado (generalmente, unos cuantos segundos). El comando **top** arroja una lista de procesos en su sistema indicando los que consumen más CPU, mostrando los procesos que utilizan más recursos. La lista está en tiempo real y se actualiza cada pocos segundos. Se proporcionan comandos para cambiar el estado de un proceso, como su prioridad.

El comando **free** despliega la cantidad de memoria RAM libre en su sistema, mostrando cuánta se utiliza y cuánta está libre, además de la que se utiliza para memoria de búferes e intercambio. **Xload** es una herramienta de X Window System mostrando carga, CPU y memoria; **iostat** despliega uso del disco y **sar** muestra información de actividad de sistema.

System Tap

System Tap es una herramienta de diagnóstico ofreciendo información acerca de implementaciones complejas del sistema. En esencia, analiza cuellos de botella de rendimiento, para ver dónde está ubicado el problema. System Tap depende de Kprobes (Kernel Dynamic Probes), que permite que los módulos de kernel realicen sondeos simples.

Frysk

Frysk es una herramienta compleja especializada en monitoreo para procesos de sistema para configurar tareas específicas de monitoreo, enfocadas en aplicaciones particulares y elegibles de un conjunto de procesos de observador para ofrecer información acerca de notificación de salida, llamadas al sistema y ejecución. También puede crear sus propios observadores personalizados para procesos. Encontrará más información acerca de Frysk en sourceware.redhat.com/frysk.

Gestor de energía de GNOME

El gestor de energía de GNOME está diseñado para aprovechar todas las características de eficiencia disponibles en computadoras portátiles y de escritorio. Da soporte a tareas como reducción de la frecuencia de la CPU, oscurecimiento de pantalla, apagado de discos duros que no están en uso y apagado o suspensión automáticos. Consulte gnome.org/projects/gnome-power-manager/index.html para conocer una descripción detallada. El gestor de energía de GNOME está integrado con la capa de abstracción de hardware (HAL, Hardware Abstraction Layer) y Dbus para detectar estados de hardware y emitir notificaciones. Éstas se generan mediante iconos para dispositivos, como el de batería. Los iconos de notificación se ubican en el panel. Un recuadro con información sobre herramientas, en el icono de la batería, mostrará cuánto tiempo le queda.

Las preferencias de administración de energía para computadoras portátiles permiten configurar opciones para inactividad, brillo y acción relacionados con la batería y corriente alterna (AC). En el caso de equipos de escritorio, se configura el tiempo de inactividad para apagar la pantalla o suspender el sistema. También se accede a las opciones en el elemento Sistema | Preferencias | Más preferencias | Gestor de energía.

Características como Cool y Quit para controles de frecuencia de las CPU Athlon y Pentium M se manejan por separado con el servicio cpuspeed. Las herramientas de informe de frecuencia de CPU se proporcionan con los paquetes cpufreq y gkrellm-freq. El paquete cpufreq instala dos aplicaciones, cpu-info y cpu-set, que necesitarán un conjunto de controladores de frecuencia de CPU estén establecidos en la configuración del kernel y puedan compilarse como módulos.

GKrellM

GKrellM es un conjunto de monitores pequeños apilables, basados en GTK para varias operaciones de sistema, red y dispositivos. Una barra de título en la parte superior de la pila desplegará el nombre host de su sistema. Como opción predeterminada, GKrellM despliega nombre de host, hora del sistema, carga de CPU, gráfica de procesos, acceso a discos, dispositivos de red como **eth0**, uso de memoria y revisión del correo. Puede cambiar el despliegue de la gráfica de un monitor, su altura, por ejemplo, al hacer clic con el botón derecho para mostrar un panel de opciones de despliegue.

Cada monitor tendrá una barra de título, mostrando, por ejemplo, CPU para la carga de la CPU, Disk para el acceso a disco y Mem para memoria. Para configurar el monitor, haga clic con el botón derecho en su barra de título. Esto desplegará los paneles de configuración para esa tarea. Por ejemplo, la configuración Disk permitirá seleccionar discos duros y particiones para ser monitoreados. Se desplegará la configuración de ventana completa, mostrando una barra lateral con menús de configuración, con el menú integrado expandido para el monitor seleccionado.

Consulte la página Man de **gkrellm** para conocer una descripción detallada de todas las opciones de configuración de monitor. El sitio de GKrellM, gkrellm.net, ofrece recursos para documentación, soporte a programas y temas. GKrellM se instala con los plug-ins integrados y el plug-in WiFi. Para incluir plug-ins y temas adicionales, tiene la opción de descargar e instalar otros paquetes. Estos proporcionarán un extenso conjunto de plug-ins y temas, incluidos controles de radio, monitor LED de teclado y una larga lista de temas como marble, Gotham City y shiny metal blue. Existen varios paquetes de plug-ins: media, misc y utils. Los plug-ins se descargan directamente de gkrellm.net y los temas de muhri.net, además de depósitos de software de distribuciones afiliadas. Los temas de usuario se colocan en un directorio **.gkrellm2/themes** del usuario.

Configuración de GKrellM

Tiene la opción de abrir directamente la ventana de configuración al hacer clic en cualquier monitor y oprimir F1. Como opción, puede abrir el menú principal y seleccionar la entrada Configuración. Para abrir el menú principal, oprima F2 o haga clic con el botón derecho en el monitor superior. Este

mismo menú se utiliza para recorrer los temas o salir de GKrellM. La ventana de configuración muestra una barra lateral para entradas de configuración como General, Integrados, Plugins y Temas. Los paneles se encuentran en la parte derecha, le permiten configurar opciones. El panel General muestra opciones globales, como despliegue de nombre host, tamaño general de la ventana y prioridad de ésta.

La entrada Integrados se expandirá para mostrar una lista de todos los monitores desplegados por GKrellM. Esta lista es extensa, incluye monitores tan variados como sensores de ventilador y calor, reloj, carga del CPU, conexiones de Internet, notificaciones de correo y uso de batería.

La entrada Plugins se expandirá para mostrar una lista de plug-ins instalados. Haga clic en la entrada Plugins directamente, para ver una lista de plug-ins disponibles con casillas de verificación. Para instalar un plug-in, marque su casilla de verificación. Conforme instale otros paquetes de plug-in, aparecerán más en la lista. Cuando se instala, un plug-in aparecerá en el menú de plug-ins expandido. Aquí puede seleccionar un plug-in para desplegar sus paneles de configuración.

Los plug-ins a usarse en todo el sistema se ubican en el directorio `/usr/lib/gkrellm2/plugins` y los temas se ubican en `/usr/share/gkrellm2/themes`. Los archivos de información de configuración y soporte de GKrellM se almacenan en el directorio `.gkrellm2`. Aquí encontrará subdirectorios para temas y plug-ins de usuario. El archivo `user_config` almacena opciones de configuración de usuario, mostrando una lista del monitor, sus opciones y configuraciones en cada línea.

Servidor GKrellM

Se utiliza GKrellM para monitorear hosts de forma remota al utilizar el servidor GKrellM, `gkrellmd`. Debe ejecutar el servidor en el sistema que quiere monitorear, al permitir el acceso a sistemas remotos para usar clientes `gkrellm` para obtener y desplegar sus estadísticas de monitoreo. Si desea ejecutar `gkrellm` como cliente para obtener y desplegar información de otro sistema ejecutando el servidor `gkrellmd`, se usa la opción `-s` y el nombre de host del servidor. El servidor tiene que configurarse para permitir se conecte el host remoto. En el siguiente ejemplo, un host remoto se conecta al servidor `gkrellmd` en ejecución en `tortuga.mipista.com`, para desplegar información acerca del host tortuga:

```
gkrellm -s tortuga.mipista.com
```

La configuración para el servidor GKrellM se maneja por el archivo de configuración `/etc/gkrellmd.conf`. Aquí especifica el monitor de host así como opciones globales como frecuencia de actualizaciones, puerto que escuchará y número máximo de clientes simultáneos. Las opciones se documentan en detalle. Revise la página Man de `gkrellmd` para ver una lista completa.

Administrador de tareas y monitor de rendimiento de KDE (KSysguard)

Se accede al administrador de tareas y monitor de rendimiento de KDE, KSysguard, en el escritorio de KDE. Esta herramienta permite monitorear rendimiento de su propio sistema, además de sistemas remotos. KSysguard proporciona valores simples o tablas detalladas de varios parámetros. Un panel Carga del sistema proporciona información gráfica acerca del uso de CPU y memoria, así como Tabla de procesos muestra una lista de procesos actuales, al utilizar un formato de árbol para mostrar dependencias. Tiene la opción de designar sus propios paneles de monitoreo con hojas de trabajo, mostrando diferentes tipos de valores que quiere desplegar y la forma en que le interesa se desplieguen, como una gráfica de barras o medidor digital. El panel Navegador de sensores es un árbol de sensores expansible para proporcionar información como Carga de CPU o Memoria usada. Existe una entrada en la parte superior para cada host al que se conecta, incluyendo su propio host local. Para designar su propio monitor, cree una hoja de trabajo, arrastre y suelte un sensor en ésta.

Grand Unified Bootloader (GRUB)

Grand Unified Bootloader (GRUB) es un cargador de arranque usado en casi todas las distribuciones de Linux. Con GRUB, los usuarios pueden seleccionar sistemas operativos que habrán de ejecutarse a partir de una interfaz de menús desplegada cuando el sistema arranca. Use las teclas de flechas para moverse a una entrada específica y oprima ENTER. Escriba **e** para editar un comando, que le permite cambiar argumentos del kernel o especificar un kernel diferente. El comando **c** lo coloca en una interfaz de línea de comandos. Suponiendo que su sistema BIOS soporta discos muy grandes, GRUB arranca de cualquier lugar en éstos. Los sistemas operativos de Linux y Unix son conocidos como “de arranque múltiple”, tomando argumentos que se les pasan en tiempo de arranque. Revise la página Man de GRUB para conocer más opciones de GRUB. Éste es un proyecto GNU con página de inicio en www.gnu.org/software/grub y su Wiki en grub.enbug.org.

NOTA Algunas distribuciones ofrecen herramientas de configuración de arranque para seleccionar su sistema o kernel predeterminado, además de establecer el límite de tiempo de espera.

Existen dos versiones de Grub, Legacy Grub y Grub 2. La primera se usa en casi todas las distribuciones, como se describe en detalle aquí. Grub2 está todavía bajo desarrollo, aunque ya se encuentra disponible. Con el tiempo Grub2 remplazará a Grub Legacy, que ya no se desarrolla.

Oficialmente, las opciones de configuración de Grub se almacenan en el archivo **/boot/grub/menu.lst** (Debian y Ubuntu). En Red Hat, Fedora y distribuciones similares, la configuración de GRUB se almacena en el archivo **/boot/grubconf** (**menu.lst** es un vínculo a este archivo). Sólo necesita crear entradas en el archivo de configuración de grub y GRUB las leerá automáticamente cuando reinicie su equipo. Es posible configurar varias opciones, como periodo de espera e imagen de fondo que habrá de utilizarse. Revise la documentación info de GRUB para conocer una descripción más detallada, **info grub**. Puede especificar un sistema de arranque al crear un título de entrada para éste, comenzando con el término **title**. Luego debe especificar dónde se ubica el kernel o programa del sistema operativo, qué disco duro y partición de éste se utilizará. Esta información se incluye entre paréntesis tras la opción **root**. La numeración empieza en 0, no en 1, y los discos duros se indican con el prefijo **hd**, aunque sean discos duros IDE o SCSI. Por tanto, **root (hd0, 2)** hace referencia al primer disco duro (**hda**) y la tercera partición de ese disco (**hda3**). En el caso de sistemas Linux, también deberá utilizar la opción **kernel** para indicar el programa que habrá de ejecutarse, especificando el nombre de ruta completo y cualquier opción que el kernel necesite. El disco RAM se indica con la opción **initrd**.

```
title Fedora Linux (2.6.21-1)
  root (hd0,2)
  kernel /boot/vmlinuz-2.6.21-1 ro root=/dev/hda3
  initrd /boot/initrd-2.6.21-1.img
```

La opción **kernel** indica cuál kernel se ejecutará. El kernel se ubica en el directorio **/boot** y tiene por nombre **vmlinuz** con el número de versión del kernel. Se tienen varios kernels en el directorio **/boot** y se usa GRUB para seleccionar el que se usará. Después del programa del kernel puede especificar cualquier opción requerida para el kernel. Esto incluye una opción **ro**, iniciando el kernel como sólo lectura. La opción **root** se utiliza para especificar el dispositivo en que instaló el sistema, su directorio raíz. En el ejemplo anterior, el sistema se instaló en el dispositivo **/dev/hda3**. Si el directorio raíz es un volumen lógico, donde se instala normalmente, el nombre del dispositivo sería algo como **/dev/VolGroup00/LogVol01**.

Si instaló la configuración de estación de trabajo estándar, su directorio raíz se instalará en un volumen lógico. Su opción `root` refiere el volumen lógico, especificando el grupo de volumen y el volumen lógico, como se muestra aquí.

```
kernel /vmlinuz-2.6.21-1 ro root=/dev/VolGroup00/LogVol100
```

Grub2 utiliza una sintaxis diferente, dependiente de un formato como el de la shell Bash. La configuración se mantiene en un archivo **grub.cfg**. Los elementos de menú se definen con un comando **menuentry**, conteniendo las opciones de configuración de menú. Los comandos son los mismos en Grub Legacy. Es importante observar que la numeración de la partición en Grub2 inicia en 1, no en 0, como Grub Legacy. Los discos duros todavía inician en 0. Así que la tercera partición del primer disco duro es **hd0,3**. En la siguiente entrada se muestra el ejemplo anterior **title** empleando una sintaxis de Grub2.

```
menuentry "Fedora Linux (2.6.21-1)" {
    set root (hd0,3)
    linux /boot/vmlinuz ro root=/dev/hda3
    initrd /boot/initrd-2.6.21-1.img
}
```

En el caso de otro sistema operativo como Windows, se utiliza la opción **rootnoverify**, para especificar dónde está instalado Windows. Esta opción instruye a GRUB para no montar la partición. Utilice la opción **chainloader+1** para permitir que GRUB acceda a éste. La opción **chainloader** indica a GRUB use otro programa para ese sistema operativo. El número indica el sector de partición donde se ubica el programa de arranque (por ejemplo, **+1** indica el primer sector).

```
title Windows XP
    rootnoverify (hd0,0)
    chainloader +1
```

Todos los sistemas de Windows querrán iniciar desde la primera partición del primer disco. Esto se vuelve un problema si quiere instalar varias versiones de Windows en diferentes particiones o instalar Windows en una partición que no sea la primera. En el caso de particiones de Windows en el mismo disco, GRUB le deja trabajar con esto porque permite ocultar otras particiones en línea y después mostrar la deseada, haciendo que aparezca como si fuera la primera partición. En este ejemplo, se oculta la primera partición y muestra la segunda. Para esto se supone que hay una partición de Windows en la segunda partición del primer disco duro (**hd0,1**). Ahora que la primera partición está oculta, la segunda aparece como la primera partición.

```
hide (hd0,0)
unhide (hd0,1)
rootnoverify (hd0,1)
```

En el caso de sistemas con varios discos duros, tal vez quiera instalar Windows en un disco que no sea el primero. GRUB numera los discos duros a partir de 0, **hd1** hace referencia al segundo disco duro y **hd0** al primero. Windows siempre querrá arrancar de la primera partición del primer disco duro. En el caso de una versión de Windows instalada en un disco duro que no sea el primero, GRUB le deja trabajar con éste permitiéndole cambiar la numeración de sus discos con el comando **map**. El primer disco se reenumera como otro disco y luego ese disco se vuelve a asignar como primer disco duro. En este ejemplo, el primer disco se correlaciona de nuevo como el segundo disco y el segundo

disco se correlaciona como el primero. Este ejemplo supone que existe un sistema de Windows en la primera partición del segundo disco duro (**hd1, 0**). Una vez que el primer disco se correlaciona de nuevo como el segundo, éste opera como el primero. Sin embargo, la operación **chainloader** todavía detecta la ubicación actual de ese sistema operativo de Windows en el segundo disco duro, (**hd1, 0**)+1, que en este ejemplo se encuentra en la primera partición. Luego GRUB arrancará la partición de Windows en el segundo disco duro, como si estuviera ubicado en el primer disco duro.

```
map (hd0) (hd1)
map (hd1) (hd0)
chainloader (hd1,0)+1
```

SUGERENCIA Si tiene problemas para arrancar en Linux y puede corregir el problema editando el archivo de configuración de Grub (por ejemplo, al cambiar los números de disco duro), puede arrancar con su CD/DVD de instalación de Linux y escribir **linux rescue** en el indicador de comandos de arranque. Siga los indicadores para arrancar su sistema con la interfaz de línea de comandos. Utilice el comando **chroot /mnt/sysimage**. Luego cambie el directorio **/boot/grub** y edite su archivo de configuración de Grub con un editor como **vi**.

Aquí se muestra un ejemplo de un archivo de configuración de Grub (**menu.lst**) con entradas para Linux y Windows. Se utilizan los ejemplos del kernel de Fedora. Observe que los parámetros del kernel se incluyen en la opción **kernel** como argumentos del kernel. El directorio raíz se instala en el volumen lógico, **/dev/VolGroup00/LogVol01**.

```
# grub.conf generado por anaconda
#
# Observe que no necesita volver a ejecutar grub después de hacer cambios a este
# archivo
# OBSERVE: Tiene una partición /boot. Esto significa que
#           todas las rutas de kernel e initrd son relativas a /boot/, eg.
#           root (hd0,1)
#           kernel /vmlinuz-version ro root=/dev/VolGroup00/LogVol01
#           initrd /initrd-versión.img
#boot=/dev/sda
default=1
timeout=5
splashimage=(hd0,1)/grub/splash.xpm.gz
hiddenmenu
title Fedora Linux (2.6.21-1)
    root (hd0,1)
    kernel /boot/vmlinuz-2.6.21-1 ro root=/dev/VolGroup00/LogVol01
    initrd /boot/initrd-2.6.21-1.img
title Windows XP
    rootnoverify (hd0,0)
    chainloader +1
```

Las entradas **title** de ejemplos previos usando la sintaxis Grub2 tendrían el siguiente aspecto. Tenga en cuenta que la numeración de la partición empieza desde 1 en Grub2, no desde 0. La primera partición en el primer disco es **hd0,1**

```
menuentry "Fedora Linux (2.6.21-1)" {
    set root (hd0,3)
```

```
        linux /boot/vmlinuz ro root=/dev/hda3
        initrd /boot/initrd-2.6.21-1.img
    }
menuentry "Windows XP" {
    rootnoverify (hd0,1)
    chainloader +1
}
```

NOTA Algunas distribuciones de Linux proporcionan Linux Loader (LILO) como administrador de arranque. Realiza los mismos tipos de tareas que GRUB. Modifique su configuración de LILO, ya sea con una herramienta de administración como Boot Manager (LILO-config) o editando directamente el archivo de configuración *letcllilo.conf*. También puede configurar LILO con la herramienta de configuración LILO de KDE (*lilo-config*).

Administración de usuarios

Como administrador de sistema, debe administrar a los usuarios de su sistema. Tiene la opción de agregar o eliminar usuarios, además de añadir y eliminar grupos, y de modificar los derechos y permisos de usuarios y grupos. También tiene acceso a los archivos de inicialización que se utilizan para configurar todas las shell de usuario, y tiene control sobre los archivos de inicialización predeterminados copiados en una cuenta de usuario cuando se crea por primera vez. Tiene la opción de decidir la manera en que se deben configurar inicialmente nuevas cuentas de usuario mediante la modificación de estos archivos.

Herramientas GUI de administración de usuario: user-admin y KUser

Los usuarios se administran de manera más sencilla con herramientas GUI de administración de usuario como user-admin de GNOME y KUser de KDE. La herramienta user-admin de GNOME es parte del paquete de herramientas de sistema de GNOME. Aunque algunas distribuciones como Red Hat todavía utilizan sus propios diseños personalizados de herramientas, otras distribuciones como Ubuntu usan la herramienta user-admin de GNOME. La herramienta KUser siempre ha estado disponible en todas las distribuciones con el escritorio de KDE.

La herramienta **users-admin** de GNOME proporciona una interfaz simple para agregar, modificar y eliminar usuarios y grupos. Se abre con la ventana Opciones de los usuarios, que muestra a los usuarios con su nombre completo, su nombre de inicio de sesión y su directorio de inicio. Un botón Añadir usuario, al lado, abrirá una nueva ventana Cuenta de usuario, con paneles Cuenta, Privilegios del usuario y Avanzado. En el panel Cuenta, se inserta el nombre de inicio de sesión y la contraseña. Un menú desplegable de perfiles le permite especificar si quiere que el usuario sea normal o un administrador. También se agrega información de contacto.

El menú Privilegios de usuario le permite controlar lo que hace un usuario, sobre todo si se le da acceso administrativo. El panel Avanzado le permite especificar las configuraciones de cuenta de usuario, como el directorio de inicio, la shell de inicio de sesión que se usa y el grupo al que pertenece. Las entradas predeterminadas ya están configuradas.

Para cambiar configuraciones más adelante, seleccione un usuario en la ventana Opciones de los usuarios, y haga clic en el botón Propiedades. Una ventana Propiedades de la cuenta se abre con los mismos paneles Cuenta, Privilegios del usuario y Avanzado. Para eliminar un usuario, selecciónelo y haga clic en el botón Borrar.

Para administrar grupos, haga clic en el botón Gestionar grupos. Esto abre una ventana Configuración de los grupos, que muestra todos los grupos. Para agregar usuarios a un grupo, selecciónelo y haga clic en Propiedades. En la ventana Propiedades de la cuenta, se desplegarán los usuarios y usted tendrá opción de seleccionar los que quiere agregar. Para agregar un nuevo grupo, haga clic en el botón Añadir grupo, para abrir la ventana Grupo nuevo, donde se especifica el nombre del grupo, su ID y se seleccionan usuarios para agregar al grupo.

En KDE, el Gestor de usuarios KDE (KUser) le permite administrar usuarios y grupos. La ventana Gestor de usuarios KDE despliega dos paneles, uno para usuarios y otro para grupos. El panel Usuarios muestra una lista de cuentas de inicio de sesión de usuario, nombre completo, el directorio de inicio y la shell de inicio de sesión, junto con el ID de usuario. En la barra de herramientas hay botones para Añadir, Editar y Eliminar usuarios y grupos, además de menús Usuario y Grupo con las entradas correspondientes. Al principio, también se desplegarán todos los usuarios y grupos de sistema. Seleccione Ocultar usuarios/grupos del sistema del menú Preferencias, para desplegar sólo usuarios y grupos normales.

Cuando se agregan nuevos usuarios, primero se pide un nombre de usuario, después se abre la ventana de propiedades de KUser con paneles para Información de usuario, Gestión de la contraseña y Grupos. Las entradas predeterminadas ya están configuradas. En realidad sólo se necesita insertar el nombre completo del usuario y la contraseña (haga clic en Introducir contraseña). El panel Contraseña le permite configurar opciones de expiración, y el panel Grupos le permite seleccionar grupos a los que quiere que pertenezca un usuario.

Puede establecer configuraciones predeterminadas para usuarios con la ventana de configuración de KUser, que se accede desde el menú Preferencias. Aquí se establece un cifrado de contraseña y políticas de expiración junto con el directorio de inicio y la shell de inicio de sesión predeterminada. También se proporciona soporte a LDAP.

SUGERENCIA Cada archivo pertenece a un usuario que controla el acceso a éste. Los archivos de sistema pertenecen al usuario root y sólo éste tiene acceso a él. Los servicios como FTP son excepciones a esta regla. Aunque el usuario root tiene acceso a ellos, los archivos del servicio pertenecen a su propio usuario especial. Por ejemplo, los archivos FTP pertenecen al usuario **ftp**. Esto proporciona usuarios con acceso a los archivos del servicio sin que deban tener también acceso de usuario root.

Directorio y archivos	Descripción
/home	El directorio de inicio propio del usuario
/etc/skel	Los archivos de inicialización predeterminados para la shell de inicio de sesión, configuración de usuario y archivos como .kde para KDE y Desktop para GNOME
/etc/shells	Las shell de inicio de sesión, como BASH o TCSH
/etc/passwd	La contraseña para un usuario
/etc/group	El grupo al que pertenece el usuario
/etc/shadow	Archivo de contraseña cifrada
/etc/gshadow	Archivo de contraseña cifrada para grupos
/etc/login.defs	Definiciones de cuentas de inicio de sesión predeterminadas para usuarios

TABLA 28-1 Rutas para archivos de configuración de usuarios

Archivos de configuración de usuario

Cualquier utilidad para administrar un usuario, como `users-admin` de GNOME o `KUser` de KDE, utiliza ciertos archivos predeterminados, denominados *archivos de configuración*, y directorios para configurar la nueva cuenta. Se utiliza un conjunto de nombres de ruta para localizar estos archivos predeterminados o para indicar dónde crear ciertos directorios de usuario. Por ejemplo, `/etc/skel` almacena archivos de inicialización para un nuevo usuario. Un directorio de inicio de un nuevo usuario se crea en el directorio `/home`. En la tabla 28-1 se muestra una lista de nombres de ruta.

SUGERENCIA Conozca más acerca de los usuarios que han iniciado sesión con el comando `w` o `who`. El comando `w` despliega información detallada acerca de cada usuario conectado, como desde dónde iniciaron sesión y cuánto tiempo han estado inactivos, y la hora y fecha del inicio de sesión. El comando `who` proporciona datos menos detallados.

Archivos de contraseñas

Un usuario obtiene acceso a una cuenta al proporcionar un inicio de sesión y una contraseña correctos. El sistema mantiene contraseñas en archivos de contraseñas, junto con información de inicio de sesión como nombre de usuario e ID. Herramientas como el comando `passwd` permiten a los usuarios cambiar sus contraseñas al modificar estos archivos; `/etc/passwd` es el archivo en que tradicionalmente se almacenan las contraseñas de usuario, aunque en forma cifrada. Sin embargo, a todos los usuarios se les permite leer el archivo `/etc/passwd`, que da acceso a las contraseñas cifradas. Para una mejor seguridad, las entradas de contraseña se almacenan en el archivo `/etc/shadow`, que está restringido al usuario `root`.

`/etc/passwd`

Cuando se agrega un usuario, una entrada para ese usuario se crea en el archivo `/etc/passwd`, al que suele conocerse como *archivo de contraseñas*. Cada entrada ocupa una línea que tiene varios campos separados por puntos. Los campos son los siguientes:

- **Nombre de usuario** Nombre de inicio de sesión del usuario.
- **Contraseña** Contraseña cifrada para la cuenta del usuario.
- **ID de usuario** Número único asignado por el sistema.
- **ID de grupo** Número utilizado para identificar el grupo al que pertenece el usuario.
- **Comentario** Cualquier información de usuario, como su nombre completo.
- **Directorio de inicio** El directorio de inicio del usuario.
- **Shell de inicio de sesión** Shell que se ejecuta cuando el usuario inicia sesión; es la shell predeterminada, que suele ser `/bin/bash`.

Dependiendo del hecho de que esté o no utilizando contraseñas de sombra, el campo de contraseña (el segundo campo) será una `x` o una forma cifrada de la contraseña del usuario. Linux implementa contraseñas de sombra, como opción predeterminada, así que estas entradas tendrán una `x` para sus contraseñas. En el siguiente ejemplo se muestra una entrada de `/etc/passwd`. Para tales entradas, debe utilizar el comando `passwd` para crear una contraseña. Observe también que el ID de usuario en este sistema particular empieza en 500, con incrementos de uno en uno. El grupo dado no es el grupo genérico `User`, sino uno que consta sólo de ese usuario. Por ejemplo, el usuario **daniel** pertenece al grupo llamado **Daniel**, no al grupo genérico **User**.


```
daniel:x:500:500:Daniel:/home/daniel:/bin/bash
chris:x:501:501:Chris:/home/chris:/bin/bash
```

SUGERENCIA Si desactiva el soporte a contraseñas de sombra, las entradas en su archivo **passwd** desplegarán las contraseñas cifradas. Debido a que cualquier usuario lee el archivo **/etc/passwd**, los intrusos también pueden acceder a estas contraseñas cifradas, y posiblemente descubrirlas.

SUGERENCIA Aunque técnicamente es posible editar las entradas del archivo **/etc/passwd** de manera directa, no se recomienda. En particular, el hecho de borrar una entrada no elimina ninguna otra información, permisos y datos asociados con un usuario, lo que abre una posible brecha de seguridad donde un intruso puede tomar el ID del usuario o el espacio en disco.

/etc/shadow y /etc/gshadow

El archivo **/etc/passwd** es un archivo de texto simple y es vulnerable a brechas de seguridad. Cualquiera que obtenga acceso al archivo **/etc/passwd** podrá descifrar o descubrir la contraseña cifrada con un ataque de fuerza bruta. La *suite* de aplicaciones de shadow implementa un nivel más elevado de seguridad. Incluye **useradd**, **groupadd**, y sus programas de actualización y eliminación correspondientes. Casi todas las demás herramientas de configuración dan soporte a medidas de seguridad de shadow. Con seguridad de shadow, las contraseñas ya no se almacenan en el archivo **/etc/passwd**. En cambio, se almacenan en un archivo separado llamado **/etc/shadow**. El acceso se restringe al usuario root.

En el siguiente ejemplo se muestra una entrada en **/etc/passwd** para un usuario.

```
chris:x:501:501:Chris:/home/chris:/bin/bash
```

También se mantiene un archivo de contraseña correspondiente, llamado **/etc/gshadow**, para grupos que requieren contraseñas.

Herramientas de contraseña

Para cambiar cualquier campo particular de un usuario dado, debe utilizar las herramientas de administración de usuario proporcionadas, como el comando **passwd**, **adduser**, **usermod**, **useradd** y **chage**, que se discuten en este capítulo. El comando **passwd** le permite cambiar sólo la contraseña. Otras herramientas no sólo hacen entradas en el archivo **/etc/passwd**, sino también crean el directorio home para el usuario e instalan archivos de inicialización en el directorio home del usuario.

Estas herramientas también le permiten controlar el acceso de los usuarios a sus cuentas. Puede establecer fechas de vencimiento para usuarios o bloquearlos para que no ingresen en sus cuentas. En el archivo **/etc/shadow**, las contraseñas de los usuarios bloqueados tendrán su contraseña antecedida con la cadena de invalidación, **!!**. Al desbloquear la cuenta se eliminan estos prefijos.

Administración de entornos de usuario

Cada vez que un usuario inicia sesión, se crean dos secuencias de comandos de perfil, una de sistema, que es la misma para cada usuario, y una de inicio de sesión que se personaliza de acuerdo con las necesidades de cada usuario. Cuando el usuario sale de su sesión, se ejecuta una secuencia de comandos de salida de sesión de usuario. Además, cada vez que se genera una shell, incluida la shell de inicio de sesión, se ejecuta una secuencia de comandos de shell de usuario. Existen diferentes tipos de secuencias de comandos utilizadas para diferentes shells. La shell

predeterminada que se utiliza de manera común es la shell BASH. Como opción, los usuarios utilizan diferentes shells, como TCSH o Z.

Secuencias de comandos de perfil

Para la shell BASH, cada usuario tiene su propia secuencia de comandos de perfil de inicio de sesión de BASH llamada **.bash_profile**, en el directorio de inicio del usuario. La secuencia de comandos de perfil se ubica en el directorio **/etc** y se llama **profile** sin un punto antes. La secuencia de comandos de shell de usuario de la shell BASH se llama **.bashrc**. El archivo **.bashrc** también ejecuta el archivo **/etc/bashrc** para implementar cualquier definición global, como las variables **PS1** y **TERM**. El archivo **/etc/bashrc** también ejecuta cualquier archivo de inicialización especializado en el directorio **/etc/profile.d**, como los que se utilizan para KDE y GNOME. El archivo **.bash_profile** ejecuta el archivo **.bashrc** y, a través de éste, el archivo **/etc/bashrc**, que implementa definiciones globales.

Como superusuario, puede editar cualquiera de estas secuencias de comandos de perfil y shell y colocar cualquier comando que quiera que se ejecute para cada usuario cuando ese usuario inicia sesión. Por ejemplo, tal vez quiera definir una ruta predeterminada para comandos, en caso de que el usuario no lo haya hecho. O tal vez quiera mostrar ante el usuario noticias recientes relacionadas con el sistema o cambios de cuenta.

/etc/skel

Cuando agrega por primera vez un usuario al sistema, debe proporcionarle versiones de esqueleto de sus archivos de inicialización de inicio de sesión, shell y salida de sesión. Para la shell BASH, se trata de los archivos **.bash_profile**, **.bashrc** y **.bash_logout**. El comando **useradd** y otras herramientas de administraciones agregan estos archivos automáticamente, al copiar cualquier archivo en el directorio **/etc/skel** del nuevo directorio de inicio del usuario. El directorio **/etc/skel** contiene archivos de inicialización para **.bash_profile**, **.bashrc** y **.bash_logout** o, si está utilizando TCSH como su shell de inicio de sesión, los archivos **.login**, **.tcshrc** y **.logout**. El directorio **/etc/skel** también contiene archivos y directorios predeterminados para su escritorio. Entre éstos se incluyen el archivo **.screenrc** para X Window System, un directorio **.kde** para el escritorio KDE y el directorio **Desktop** que contiene archivos de configuración predeterminados para el escritorio de GNOME.

Como superusuario, puede configurar el archivo **.bash_profile** o **.bashrc** en el directorio **/etc/skel** de la forma que quiera. Por lo general, se incluyen asignaciones básicas de variables del sistema que definen nombres de ruta para comandos y alias de comandos. Las variables **PATH** y **BASH_ENV** se definen en **.bash_profile**. Una vez que los usuarios tienen sus propios archivos **.bash_profile** o **.bashrc**, éstos redefinen las variables o agregan nuevos comandos a medida que los seleccionan.

/etc/login.defs

Los valores de todo el sistema utilizados por las utilerías de creación de usuarios y grupos como **useradd** y **usergroup** se almacenan en el archivo **/etc/login.defs**. Aquí encontrará el rango de ID de grupo y usuarios posibles. **UID_MIN** almacena el número mínimo de ID de usuario y **UID_MAX** el número máximo. Varias opciones de contraseña rigen los controles de contraseñas (como **PASS_MIN_LEN**, que determina el número mínimo de caracteres que se permiten en una contraseña). Opciones como **CREATE_HOME** se establecen para indicarles a herramientas de usuario como **useradd** que creen directorios de inicio para nuevas cuentas, como opción predeterminada. Aquí se muestran ejemplos de estas entradas:

```
MAIL_DIR /var/spool/mail
PASS_MIN_LEN 5
CREATE_HOME yes
```

/etc/login.access

Puede controlar el acceso de inicio de sesión por parte de usuarios remotos a su sistema con el archivo **/etc/login.access**. Está integrado por entradas que incluyen usuarios, si se les permite el acceso y desde dónde pueden acceder al sistema. Un registro en este archivo consta de tres campos delimitados por dos puntos: un signo más (+) o menos (–) que indica si a los usuarios se les permite el acceso, los nombres de inicio de sesión de los usuarios a los que se les permite el acceso y el sistema remoto (host) o la terminal (dispositivo tty) desde donde están tratando de iniciar sesión. En el siguiente ejemplo se habilita al usuario **carlos** para que acceda desde el sistema remoto **conejo.mipista.com**:

```
+ :carlos:conejo.mipista.com
```

Puede incluir más de un usuario o ubicación, o utilizar la opción **ALL** en lugar de usuarios o ubicaciones para permitir el acceso a todos los usuarios y ubicaciones. La opción **ALL** se califica con la opción **EXCEPT** para permitir el acceso a todos los usuarios, excepto los que se especifiquen. En la siguiente entrada se permite a cualquier usuario válido iniciar sesión en el sistema empleando la consola, excepto a los usuarios **larisa** y **alicia**:

```
+ :ALL EXCEPT larisa alicia:console
```

Otros archivos de control de acceso se utilizan para controlar el acceso a servicios específicos, como los archivos **host.deny** y **host.allows** que se utilizan con el daemon **tcpd** para servidores que dan soporte a **xinetd**.

Control de contraseñas de usuario

Una vez que ha creado una cuenta de usuario, puede controlar el acceso del usuario a ésta. La herramienta **passwd** le permite bloquear y desbloquear cuentas de usuarios. Se utiliza el comando **passwd** con la opción **-l** para bloquear una cuenta, invalidando su contraseña, y se utiliza la opción **-u** para desbloquearla.

También puede obligar al usuario a cambiar su contraseña a intervalos definidos al configurar una fecha de vencimiento para una contraseña. El comando **chage** le permite especificar un límite de vencimiento para la contraseña de un usuario. A un usuario se le pide que cambie su contraseña cada mes, cada semana o cada fecha determinada. Una vez que la contraseña expira, se le pide que inserte una nueva. Se envía un aviso de antemano, que le indica al usuario cuánto tiempo le queda antes de que la contraseña expire. Si existe una cuenta que quiere cerrar, puede hacer que la contraseña expire permanentemente. Incluso puede apagar cuentas que han estado inactivas por mucho tiempo. En el siguiente ejemplo, la contraseña para la cuenta de **carlos** sólo será válida por siete días. La opción **-M** con el número de días configura el tiempo máximo que una contraseña es válida.

```
chage -M 7 carlos
```

Para configurar una fecha particular para que la cuenta expire, utilice la opción **-E** con la fecha especificada en mm/dd/aaaa.

```
chage -E 07/30/2003 carlos
```

Para saber los valores de vencimiento actuales de una cuenta, utilice la opción **-l**.

```
chage -l carlos
```

También puede combinar sus opciones en un comando.

```
chage -M 7 -E 07/30/2003 carlos
```

Una lista de opciones de **chage** se muestra en la tabla 28-2.

Opción	Descripción
-m	Número mínimo de días que debe esperar un usuario antes de poder cambiar su contraseña.
-M	Número máximo de días que debe pasar un usuario sin cambiar su contraseña.
-d	El último día en que fue cambiada la contraseña.
-E	Fecha de vencimiento específica de una contraseña; debe estar en formato aaaa-mm-dd o en un formato de uso común como mm/dd/aaaa.
-I	Periodo de inactividad permitido para la cuenta (en días), después del cual expirará la contraseña.
-W	Periodo de advertencia, número de días antes del vencimiento, cuando se le enviará al usuario un mensaje de advertencia.
-l	Despliega controles de vencimiento de la contraseña actual.

TABLA 28-2 Opciones del comando `chage`

Adición y eliminación de usuarios con `useradd`, `usermod` y `userdel`

Linux también proporciona los comandos `useradd`, `usermod` y `userdel` para administrar cuentas de usuario. Todos estos comandos toman su información como opciones en la línea de comandos. Si no se especifica una opción, se utilizan los valores predeterminados. Se trata de operaciones de línea de comandos. Para utilizarlas en su escritorio primero necesita abrir una ventana de terminal (haga clic con el botón derecho en el escritorio y seleccione Abrir terminal) e insertar los comandos en el indicador de comandos de shell.

Si está utilizando una interfaz de escritorio, debe utilizar las herramientas GUI para administrar cuentas de usuario. Cada distribución de Linux suele proporcionar una herramienta para administrar usuarios. Además, puede utilizar la herramienta KUser de K Desktop u Opciones de los usuarios de las herramientas de sistema de GNOME. Consulte la tabla 28-3 para conocer una lista de herramientas de administración de usuarios.

Herramienta	Descripción
KUser	Herramienta de K Desktop para agregar, eliminar y modificar usuarios y grupos.
Opciones de los usuarios de GNOME	Herramienta del escritorio GNOME para agregar, eliminar y modificar usuarios y grupos.
<code>useradd nombredeusuario opciones</code>	Agrega un usuario.
<code>userdel nombredeusuario</code>	Elimina un usuario.
<code>usermod nombredeusuario opciones</code>	Modifica las propiedades de un usuario.
<code>groupadd nombredeusuario opciones</code>	Agrega un grupo.
<code>groupdel nombredegrupo</code>	Elimina un grupo.
<code>groupmod nombredegrupo opciones</code>	Modifica un nombre de grupo.

TABLA 28-3 Herramientas de administración de usuarios y grupos

useradd

Con el comando **useradd**, se insertan valores como opciones en la línea de comandos, como el nombre de un usuario, para crear una cuenta de usuario. Luego se crea un nuevo inicio de sesión y directorio para ese nombre al utilizar todas las características predeterminadas para una nueva cuenta.

```
# useradd carlos
```

La utilidad **useradd** busca primero en el archivo **/etc/login.defs** valores predeterminados para crear una nueva cuenta. Para los valores predeterminados no definidos en el archivo **/etc/login.defs**, **useradd** suministra sus propios valores. Se despliegan estas opciones predeterminadas al utilizar el comando **useradd** con la opción **-D**. Los valores predeterminados incluyen el nombre de grupo, el ID de usuario, el directorio de inicio, el directorio **skel**, y la shell de inicio de sesión. Los valores que el usuario inserta en la línea de comandos invalidarán las opciones predeterminadas correspondientes. El nombre de grupo se refiere a aquel en donde se coloca la nueva cuenta. Como opción predeterminada, es **other**, que significa que la nueva cuenta no pertenece a ningún grupo. El ID de usuario es un número que identifica la cuenta de usuario. **skel** es el directorio de sistema que almacena copias de archivos de inicialización. Estos archivos se copian en el nuevo directorio de inicio del usuario cuando se crea. La shell de inicio de sesión es el nombre de ruta de una shell particular que el usuario planea utilizar.

Opción	Descripción
-d <i>dir</i>	Configura el directorio de inicio del nuevo usuario.
-D	Displays defaults for all settings. Can also be used to reset default settings for the home directory (-b), group (-g), shell (-s), expiration date (-e), and password expirations (-f).
-e <i>mm/dd/yy</i>	Despliega opciones predeterminadas para todas las configuraciones. También se utiliza para reiniciar configuraciones predeterminadas para el directorio home (-b), grupo (-g), shell (-s), fecha de vencimiento (-e) y vencimiento de contraseña (-f).
-f <i>días</i>	Configura el número de días que una cuenta permanece inactiva después de que su contraseña expira.
-g <i>grupo</i>	Configura un grupo.
-m	Crea un directorio de inicio del usuario, si no existe.
-m -k <i>skel-dir</i>	Configura el directorio de esqueleto que almacena archivos de esqueleto, como .profile que se copian en el directorio de inicio del usuario automáticamente cuando se crean; el directorio predeterminado es /etc/skel .
-M	No crea el directorio de inicio del usuario.
-p <i>contraseña</i>	Proporciona una contraseña cifrada (crypt o MD5). Sin argumentos, la cuenta se deshabilita inmediatamente.
-s <i>shell</i>	Configura la shell de inicio de sesión del usuario nuevo. Es /bin/bash como opción predeterminada, la shell de BASH.
-u <i>iddeusuario</i>	Configura el ID de usuario del nuevo usuario. La opción predeterminada es el incremento del número mayor utilizado hasta el momento.

TABLA 28-4 Opciones para useradd y usermod

El comando **useradd** tiene opciones que corresponden a cada valor predeterminado. En la tabla 28-4 se muestra una lista de todas las opciones que se utilizan con el comando **useradd**. Se utilizan valores específicos en lugar de cualquiera de estas opciones predeterminadas cuando se crea una cuenta particular. No se puede acceder al inicio de sesión hasta que se habilite. En el siguiente ejemplo, el nombre de grupo para la cuenta **carlos** es **intro1** y el ID de usuario es 578:

```
# useradd carlos -g intro1 -u 578
```

Una vez que agregue una nueva cuenta de inicio de sesión de usuario, necesita darle una contraseña. Las entradas de contraseña se colocan en los archivos **/etc/passwd** y **/etc/shadow**. Utilice el comando **passwd** para crear una nueva contraseña para el usuario, como se muestra aquí. La contraseña que inserta no aparecerá en su pantalla. Se le pedirá que repita la contraseña. Se enviará un mensaje indicando que la contraseña se cambió correctamente.

```
# passwd carlos
Changing password for user carlos
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully
#
```

usermod

El comando **usermod** le permite cambiar los valores de cualquiera de estas características. Se cambia el directorio de inicio o el ID de usuario. Incluso se cambia el nombre de usuario para una cuenta. El comando **usermod** toma las mismas opciones que **useradd**, que se muestran en la tabla 28-4.

userdel

Cuando se quiere eliminar a un usuario del sistema, se utiliza el comando **userdel** para eliminar el inicio de sesión del usuario. Con la opción **-r**, también se eliminará el directorio de inicio del usuario. En el siguiente ejemplo, el usuario **carlos** se elimina del sistema:

```
# userdel -r carlos
```

Administración de grupos

Tiene la opción de administrar grupos al utilizar comandos de shell o utilerías de GUI. Los grupos son una forma efectiva de administrar el acceso y los permisos, permitiéndole controlar varios usuarios únicamente con su nombre de grupo.

```
/etc/group y /etc/gshadow
```

El archivo de sistema que almacena entradas de grupo es **/etc/group**. El archivo consta de un grupo de registros, con un registro por línea y sus campos separados por dos puntos. Un registro de grupo tiene cuatro campos: un nombre de grupo, una contraseña, su ID y los usuarios que son parte del grupo. El campo Contraseña puede estar en blanco. Los campos de un registro de grupo son los siguientes:

- **Nombre de grupo** El nombre del grupo, que debe ser único.
- **Contraseña** Con seguridad de sombra implementada, este campo es una **x**, con la contraseña indicada en el archivo **/etc/gshadow**.

- **ID de grupo** El número asignado por el sistema para identificar este grupo
- **Usuarios** La lista de usuarios que pertenecen a ese grupo, separados por comas.

Aquí se muestra un ejemplo de una entrada en el archivo `/etc/group`. El grupo se llama **motores**, la contraseña se administra con seguridad de sombra, el ID de grupo es 100, y los usuarios que son parte de este grupo son **carlos**, **roberto**, **valeria** y **alicia**:

```
motores:x:100:carlos,roberto,valeria,alicia
```

Como en el caso del archivo `/etc/passwd`, es mejor cambiar las entradas de grupo al utilizar una utilidad de administración de grupo como **groupmod** o **groupadd**. Todos los usuarios tienen acceso de lectura al archivo `/etc/group`. Con seguridad de sombra, los datos seguros del grupo, como contraseñas, se almacenan en el archivo `/etc/gshadow`, al que sólo el usuario root tiene acceso.

Grupos privados de usuarios

Un nuevo usuario se asigna a un grupo especial configurado sólo para ese usuario y se le da el nombre del usuario. Por tanto, al nuevo usuario **daniel** se le da un grupo predeterminado llamado **daniel**. El grupo **daniel** también se mostrará en la lista de grupos. A este método de asignación de grupos de usuario predeterminado se le denomina esquema de grupo privado de usuarios (UPG, User Private Group). Los grupos complementarios son grupos adicionales a los que el usuario tal vez pertenezca. Por lo general, se asignaban todos los usuarios a un grupo llamado **users** que sometía a todos los usuarios a los controles de permisos del grupo **users**. Con UPG, cada usuario tiene su propio grupo, con sus propios permisos.

Directorios de grupos

Al igual que para los usuarios, puede crear un directorio de inicio para un grupo. Para ello, sólo cree un directorio para el grupo en `/home` y cambie su grupo de inicio a ese grupo y permita el acceso a cualquier miembro del grupo. En el siguiente ejemplo se crea un directorio llamado **motores** y cambia su grupo a **motores**:

```
mkdir /home/motores
chgrp engines /home/motores
```

Entonces los permisos de lectura, escritura y ejecución para el nivel de grupo deben configurarse con el comando **chmod**, que se analiza más adelante en este capítulo:

```
chmod g+rxw /home/motores
```

Ahora cualquier miembro del grupo **motores** puede acceder al directorio `/home/motores` y a cualquier archivo compartido que haya ahí. Éste se vuelve un directorio compartido para un grupo. En realidad, se utiliza el mismo procedimiento para que otros directorios se vuelvan compartidos en cualquier ubicación en el sistema de archivos.

Los archivos que se encuentran dentro del directorio compartido también deben tener sus permisos configurados para permitir el acceso a otros usuarios del grupo. Cuando un usuario coloca un archivo en un directorio compartido, necesita configurar los permisos en ese archivo para permitir que otros miembros del grupo accedan a éste. Un permiso de lectura permitirá a otros desplegarlo, uno de escritura les permitirá cambiarlo y uno de ejecución les permitirá ejecutarlo (se utiliza para programas y secuencias de comandos). En el siguiente ejemplo, primero se cambia el

archivo **mimodelo** al grupo **motores**. Después se copia el archivo **mimodelo** al directorio **/home/motores** y se configuran permisos de escritura y lectura para el grupo **motores**:

```
$ chgrp motores mimodelo
$ cp mimodelo /home/motores
$ chmod g+rw /home/motores/mimodelo
```

Administración de grupos al utilizar **groupadd**, **groupmod** y **groupdel**

También se administran grupos con los comandos **groupadd**, **groupmod** y **groupdel**. Estas operaciones de línea de comandos le permiten administrar de manera más rápida un grupo en la ventana de terminal.

groupadd y **groupdel**

Con el comando **groupadd**, se crean nuevos grupos. Cuando agrega un grupo al sistema, éste coloca el nombre del grupo en el archivo **/etc/group** y le asigna un número de ID de grupo. Si la seguridad de sombra está activa, los cambios se hacen al archivo **/etc/gshadow**. El comando **groupadd** sólo crea la categoría de grupo. Necesita agregar usuarios al grupo individualmente. En el siguiente ejemplo, el comando **groupadd** crea el grupo **motores**:

```
# groupadd motores
```

Se elimina un grupo con el comando **groupdel**. En el siguiente ejemplo, se elimina el grupo **motores**:

```
# groupdel motores
```

groupmod

Tiene la opción de cambiar el nombre de un grupo o su ID al utilizar el comando **groupmod**. Inserte **groupmod -g** con el nuevo número de ID y el nombre de grupo. Para cambiar el nombre de un grupo, se utiliza la opción **-n**. Inserte **groupmod -n** con el nuevo nombre de grupo, seguido por el nombre actual. En el siguiente ejemplo, se cambia el nombre del grupo **motores** a **trenes**:

```
# groupmod -n trenes motores
```

Control de acceso a directorios y archivos: **chmod**

Cada archivo y directorio de Linux contiene un conjunto de permisos que determinan quién accede a éstos y cómo. Se configuran estos permisos para limitar el acceso de una de tres formas: se restringe el acceso sólo a usted, se permite el acceso a usuarios en un grupo prediseñado o se permite que todos tengan acceso al sistema. También puede controlar la manera en que se accede a un archivo o directorio determinado.

NOTA Consulte el capítulo 17 para aprender a utilizar SELinux con el fin de configurar permisos para usuarios y archivos.

Permisos

Un archivo o directorio puede tener permisos de lectura, escritura y ejecución. Cuando un archivo se crea, automáticamente se le asignan permisos de lectura y escritura al propietario, permitiéndole desplegar y modificar el archivo. Puede cambiar estos permisos a cualquier combinación que quiera. Un archivo también tiene permisos de sólo lectura, que evita cualquier modificación.

SUGERENCIA En GNOME y KDE se cambian los permisos de manera sencilla al hacer clic con el botón derecho en un icono de archivo o directorio y seleccionar *Propiedades*. En el panel *Permisos* verá opciones para configurar los permisos *Propietario*, *Grupo* y *Otros*.

Categorías de permisos

Tres categorías diferentes de usuarios tienen acceso a un archivo o directorio: el propietario, el grupo y todos los demás que no pertenecen al grupo. El propietario es el usuario que creó el archivo. Cualquier archivo que cree usted, le pertenecerá. También se permite que un grupo tenga acceso al archivo. A menudo, los usuarios se colocan en grupos. Por ejemplo, el administrador de sistemas colocará a todos los usuarios de una clase o proyecto dado en un grupo. Un usuario puede otorgar acceso a un archivo a los miembros de un grupo designado. Por último, también se abre el acceso a un archivo a todos los usuarios del sistema. En este caso, cada usuario que no es parte del grupo del archivo tiene acceso a éste. En ese sentido, todos los usuarios del sistema entran en la categoría “otros”. Si quiere dar el mismo acceso a todos los usuarios de su sistema, configure los permisos para el grupo y otros. De esa forma, se incluye a los miembros del grupo (permiso de grupo) y a todos los usuarios que no son miembros (permiso de otros).

Permisos de lectura, escritura y ejecución

Cada categoría tiene su propio conjunto de permisos de lectura, escritura y ejecución. La primera configuración controla el propio acceso del usuario a sus archivos (el acceso de propietario). La segunda configuración controla el acceso del grupo a los archivos del usuario. La tercera controla el acceso para todos los usuarios a sus archivos. Las tres configuraciones de permisos de lectura, escritura y ejecución para las tres categorías (propietario, grupo y otros) hacen un total de nueve tipos de permisos.

El comando **ls** con la opción **-l** despliega información detallada acerca del archivo, incluidos sus permisos. En el siguiente ejemplo, el primer conjunto de caracteres a la izquierda es una lista de permisos configurados para el archivo **misdatos**:

```
$ ls -l misdatos
-rw-r--r-- 1 carlos clima 207 Feb 20 11:55 misdatos
```

Un permiso vacío se representa con un guión, **-**. El permiso de lectura se representa con **r**, el de escritura con **w** y el de ejecución con **x**. Observe que hay diez posiciones. El primer carácter indica el tipo de archivo. En un sentido general, un directorio se considera un tipo de archivo. Si el primer carácter es un guión, se muestra un archivo. Si el primer carácter es una **d**, se presenta la información de un directorio.

Los siguientes nueve caracteres se ordenan de acuerdo con diferentes categorías de usuario. El primer conjunto de tres caracteres es el de permisos de propietario para el archivo. El segundo es el conjunto de permisos de grupo para el archivo. Y el tercero es el conjunto de permisos de otros para el archivo.

Permisos en GNOME

En GNOME, se configura un directorio o archivo de permisos al utilizar el panel *Permisos* en su ventana *Propiedades*. Haga clic con el botón derecho en la entrada del archivo o directorio en la ventana del administrador de archivos y seleccione *Propiedades*. Después seleccione el panel *Permisos*. Aquí encontrará menús emergentes para *Lectura*, *Escritura* y *Ejecución*, junto con filas para *Propietario*, *Grupo* y *Otros*. Se configuran permisos de propietario como *Sólo lectura* o *Lectura y escritura*. Para grupo y otros, también se configura la opción *Ninguno*, que niega el acceso. El nombre de grupo se expande a un menú emergente que presenta diferentes grupos; seleccione uno para cambiar el grupo del archivo. Si quiere ejecutar esto como una aplicación (digamos, una

secuencia de comandos de shell) coloque una marca en la casilla Permitir ejecutar el archivo como un programa. Esto tiene el efecto de configurar el permiso de ejecución.

El panel Permisos para directorios opera casi de la misma forma, pero incluye dos entradas, Acceso a carpeta y Acceso a archivo. La entrada Acceso a carpeta controla el acceso a la carpeta con opciones para Sólo listar archivos, Acceder a archivos y Crear y borrar archivos. Éstas corresponden a los permisos de lectura, lectura y ejecución y lectura/escritura/ejecución dados a los directorios. La entrada Acceso a archivo le permite configurar permisos para esos archivos en el directorio. Son iguales que para los archivos: para propietario, Lectura o Lectura y escritura; para el grupo y otros, la entrada agrega una opción Ninguno para negar el acceso. Para configurar los permisos para todos los archivos en el directorio, como corresponde (no sólo a la carpeta), haga clic en el botón Aplicar permisos a los archivos contenidos.

Permisos en KDE

En KDE, puede establecer un permiso de directorio o archivo al utilizar el panel Permisos en la ventana Propiedades. Haga clic con el botón derecho en la entrada del archivo o directorio, en la ventana del administrador de archivos, y seleccione Propiedades. Después seleccione el panel Permisos. Aquí encontrará menús emergentes para Propietario, Grupo y Otros. Las opciones incluyen Lectura posible, Lectura y escritura posibles, y Prohibido. En el caso de un acceso más detallado, haga clic en el botón Permisos avanzados para desplegar una tabla con el fin de revisar el acceso de lectura, escritura y ejecución (r, w, x) para el propietario, el grupo y otros. También puede establecer los permisos sticky bit y usuario e ID de grupo. El botón Añadir entrada le permite configurar el acceso a ACL, que especifica si ciertos usuarios o grupos pueden o no tener acceso al archivo.

Los directorios tienen opciones un poco diferentes: Se puede visualizar el contenido y Se puede visualizar y modificar el contenido, que son los permisos de lectura y escritura. Tiene la opción de aplicar cambios a todos los subdirectorios y los archivos en éstos. Al hacer clic en el botón Permisos avanzados, se despliega la misma tabla de lectura, escritura y ejecución para propietario, grupo y otros. Haga clic en una entrada de la tabla para activar o desactivar un permiso. En la columna Efectivo, se muestran permisos seleccionados. Utilice el botón Añadir entrada, para agregar una entrada de ACL para controlar el acceso a usuarios específicos y grupos.

chmod

Se utiliza el comando **chmod** para cambiar diferentes configuraciones de permisos. **chmod** toma dos listas como argumentos: cambios de permisos y nombres de archivos. Se especifica una lista de permisos de dos diferentes formas. Una forma utiliza símbolos de permisos y se le conoce como *método simbólico*. La otra utiliza lo que se conoce como “enmascaramiento binario” y se hace referencia a éste ya sea como *método absoluto* o *relativo*. En la tabla 28-5 se muestra una lista de opciones para el comando **chmod**.

NOTA Cuando un programa pertenece al usuario root, el establecimiento de los permisos de ID de usuario le dará al usuario la habilidad de ejecutar el programa con permisos de root. Esto plantea un riesgo de seguridad serio para cualquier programa que realice cambios (como **rm**, que elimina archivos).

Propietario

Los archivos y directorios pertenecen a un propietario y un grupo. Un grupo suele constar de una colección de usuarios, que pertenecen al mismo grupo. En el siguiente ejemplo, el usuario **roberto** es el propietario del archivo **misdatos** y pertenece al grupo **clima**:

```
-rw-r--r-- 1 roberto clima 207 Feb 20 11:55 misdatos
```

Comando u Opción	Ejecución
<code>chmod</code>	Cambia los permisos de un directorio o archivo.
Opciones	
<code>+</code>	Agrega un permiso.
<code>-</code>	Elimina un permiso.
<code>=</code>	Asigna un conjunto completo de permisos.
<code>r</code>	Establece permisos de lectura para un archivo o directorio. Un archivo puede desplegarse o imprimirse. Un directorio tiene la lista de sus archivos desplegados.
<code>w</code>	Establece permisos de escritura para un archivo o directorio. Un archivo puede editarse o eliminarse. Un directorio puede eliminarse.
<code>x</code>	Establece un permiso de ejecución para un archivo o directorio. Si el archivo es una secuencia de comandos de shell, se ejecuta como un programa. Se puede cambiar y entrar a un directorio.
<code>u</code>	Establece permisos para el usuario que creó y es propietario del archivo o directorio.
<code>g</code>	Establece permisos para acceso de grupo a un archivo o directorio.
<code>o</code>	Establece permisos para acceder a un archivo o directorio por todos los usuarios en el sistema.
<code>a</code>	Establece permisos de acceso a propietario, grupo y todos los demás usuarios.
<code>s</code>	Configura los permisos de ID de usuario e ID de grupo; el programa pertenece al propietario y al grupo.
<code>t</code>	Configura permisos sticky bit; el programa permanece en la memoria.
Comandos	
<code>chgrp nombredegrupo nombresdearchivo</code>	Cambia el grupo para un archivo o archivos.
<code>chown nombre-usuario nombresdearchivo</code>	Cambia el propietario de un archivo o archivos.
<code>ls -l nombredearchivo</code>	Presenta un nombre de archivo con sus permisos desplegados.
<code>ls -ld directorio</code>	Presenta un nombre de directorio con sus permisos desplegados.
<code>ls -l</code>	Despliega todos los archivos de un directorio con sus permisos desplegados.

TABLA 28-5 Operaciones de permisos de archivo y directorio

Un grupo también consta de un usuario, por lo general el que crea el archivo. Cada usuario del sistema, incluido el root, es asignado a su propio grupo del que es el único miembro, asegurando el acceso sólo a ese usuario. En el siguiente ejemplo, el usuario **roberto** es propietario del archivo informe y pertenece al grupo de un solo usuario, **roberto**:

```
-rw-r--r-- 1 roberto roberto 305 Mar 17 12:01 informe
```

El usuario **root**, el administrador del sistema, es propietario de casi todos los archivos de sistema que también pertenecen al grupo **root**, cuyo único miembro es el usuario **root**. Casi todos los archivos de administración, como los de configuración en el directorio **/etc**, pertenecen al usuario **root** y también al grupo **root**. Sólo el usuario **root** tiene permiso para modificarlos, mientras que los usuarios normales pueden leerlos y, en el caso de los programas, también los ejecutan. En el siguiente ejemplo, el usuario **root** es propietario del archivo **fstab** del directorio **/etc**, que también pertenece al grupo de usuario **root**.

```
-rw-r--r-- 1 root root 621 Apr 22 11:03 fstab
```

Ciertos directorios y archivos que se ubican en el directorio del sistema pertenecen a un servicio y no al usuario **root**, porque los servicios necesitan cambiar directamente estos archivos. Esto resulta particularmente cierto en el caso de servicios que interactúan con usuarios remotos, como los de Internet. Casi todos estos archivos se ubican en el directorio **/var**. Aquí encontrará archivos y directorios administrados por servicios como el servidor proxy Squid y el servidor de nombres de dominio (DNS, Domain Name Server). En este ejemplo, el directorio del servidor proxy Squid es propiedad del usuario **squid** y pertenece al grupo **squid**:

```
drwxr-x--- 2 squid squid 4096 Jan 24 16:29 squid
```

Cambio del propietario o grupo de un archivo: **chown** y **chgrp**

Aunque otros usuarios tengan acceso a un archivo, sólo el propietario puede cambiar sus permisos. Sin embargo, si quiere dar control a otro usuario sobre uno de sus permisos del archivo, puede cambiar el propietario del archivo de usted a otro usuario. El comando **chown** transfiere el control de un archivo a otro usuario. Este comando toma como primer argumento el nombre del otro usuario. Después del nombre de usuario, se incluyen los archivos que está transfiriendo. En el siguiente ejemplo, el usuario da el control del archivo **misdatos** al usuario **roberto**:

```
$ chown roberto misdatos
$ ls -l misdatos
-rw-r--r-- 1 roberto clima 207 Feb 20 11:55 misdatos
```

Si así lo desea, también puede cambiar el grupo para un archivo, empleando el comando **chgrp**. Éste toma como primer argumento el nombre del nuevo grupo para uno o varios archivos. Después del nombre del nuevo grupo, se incluyen los archivos que quiere cambiar a ese grupo. En el siguiente ejemplo, el usuario cambia el nombre de grupo por **hoy** y **findesemana** al grupo **pronostico**. Luego el comando **ls -l** refleja el cambio de grupo.

```
$ chgrp pronostico hoy finesemana
$ ls -l
-rw-rw-r-- 1 carlos pronostico 568 Feb 14 10:30 hoy
-rw-rw-r-- 1 carlos pronostico 308 Feb 17 12:40 finesemana
```

Tiene la opción de combinar la operación **chgrp** con el comando **chown** al adjuntar un grupo al nuevo propietario con dos puntos.

```
$ chown jorge:pronostico mañana
-rw-rw-r-- 1 jorge pronostico 568 Feb 14 10:30 mañana
```

Configuración de permisos: símbolos de permisos

El método simbólico de establecimiento de permisos utiliza los caracteres **r**, **w** y **x** para lectura, escritura y ejecución, respectivamente. Es posible agregar o eliminar cualquiera de estos permisos. El símbolo para agregar un permiso es el signo más, **+**. El símbolo para eliminar un permiso es el signo menos, **-**. En el siguiente ejemplo, el comando **chmod** agrega el permiso de ejecución y elimina el permiso de escritura para el archivo **misdatos** de todas las categorías. El permiso de lectura no se cambia.

```
$ chmod +x-w misdatos
```

Los símbolos de permiso también especifican cada categoría de usuario. Las categorías propietario, grupo y otros se representan con los caracteres **u**, **g** y **o**, respectivamente. Observe que la categoría de propietario se representa con una **u** y se considera como la del usuario. El símbolo para una categoría se coloca antes del signo más o menos que se pone antes de los permisos de lectura, escritura y ejecución. Si no se utiliza un símbolo de categoría, se suponen todas las categorías, y los permisos que se especifican se establecen para usuario, grupo y otros. En el siguiente ejemplo, el primer comando **chmod** establece los permisos de lectura y escritura para el grupo. El segundo comando establece permisos de lectura para los usuarios. Observe que no hay espacios entre las especificaciones de permisos y la categoría. La lista de permisos es sólo una frase larga, sin espacios.

```
$ chmod g+rw misdatos
$ chmod o+r misdatos
```

Un usuario puede eliminar permisos y también agregarlos. En el siguiente ejemplo, el permiso de lectura se establece para los otros usuarios, pero se eliminan los permisos de escritura y ejecución:

```
$ chmod o+r-wx misdatos
```

Existe otro carácter de permiso, **a**, que representa todas las categorías. El carácter **a** es el predeterminado. En el siguiente ejemplo, los dos comandos son equivalentes. El permiso de lectura se configura explícitamente con el carácter **a**, que denota todos los tipos de usuarios: otros, grupo y usuario.

```
$ chmod a+r misdatos
$ chmod +r misdatos
```

Una de las operaciones de permiso más comunes consiste en configurar el permiso de ejecución del archivo. Esto suele hacerse en el caso de archivos de programa de shell. Los permisos de ejecución indican que un archivo contiene instrucciones, ejecutadas directamente por el sistema. En el siguiente ejemplo, se le asigna un permiso de ejecución al archivo **lsc** y después se ejecuta:

```
$ chmod u+x lsc
$ lsc
main.c lib.c
$
```

Permisos absolutos: enmascaramiento binario

En lugar de los símbolos de permiso en la tabla 28-5, muchos usuarios encuentran más conveniente utilizar el método absoluto. El *método absoluto* cambia todos los permisos a la vez, en lugar de especificar uno u otro. Utiliza un *enmascaramiento binario* que hace referencia a todos los permisos de cada categoría.

Las tres categorías, cada una con tres permisos, cumplen con un formato binario octal. Los números octales están basados en una estructura basada en 8. Cuando se traducen a un número binario, cada dígito octal se convierte en tres dígitos binarios. Un número binario es un conjunto de dígitos 1 y 0. Tres dígitos octales en un número se traducen en tres conjuntos de tres dígitos binarios, que en total son nueve (y el número exacto de permisos para ese archivo).

Puede utilizar los dígitos octales como una máscara para establecer los diferentes permisos de archivos. Cada dígito octal se aplica a una de las categorías de usuario. Puede considerar que los dígitos coinciden con las categorías de permisos de izquierda a derecha, empezando con la categoría propietario. El primer dígito octal se aplica a la categoría propietario, el segundo al grupo y el tercero a otros. El dígito octal real que seleccione determinará los permisos de lectura, escritura y ejecución para cada categoría. En este punto, necesita saber cómo se traducen los dígitos octales en sus equivalentes binarios.

Cálculo de números octales

Una forma sencilla de calcular el número octal aprovecha el hecho de que cualquier número utilizado para permisos será una combinación derivada de la suma, en términos decimales, los números 4, 2 y 1. Se utiliza 4 para permisos de lectura, 2 para escritura y 1 para ejecución. Los permisos de lectura, escritura y ejecución son simplemente la suma de 4 + 2 + 1 para obtener 7. Los permisos de lectura y ejecución agregan 4 y 1 para obtener 5. Se utiliza este método para calcular el número octal para cada categoría. Para obtener 755, se suma 4 + 2 + 1 para los permisos de lectura, escritura y ejecución del propietario, 4 + 1 para los permisos de lectura y ejecución del grupo y 4 + 1 de nuevo para los permisos de escritura y ejecución de otros.

Máscaras binarias

Cuando se trabaja con una máscara binaria, necesita especificar tres dígitos para las tres categorías, además de sus permisos. Esto hace que la máscara binaria sea menos versátil que los símbolos de permisos. Para configurar el permiso de ejecución del propietario y quitar el permiso de escritura para el archivo **misdatos** y retener el mismo permiso de lectura, necesita utilizar el dígito octal 5 (101). Al mismo tiempo, necesita especificar los dígitos para acceso de grupo y otros usuarios. Si estas categorías retienen el acceso de lectura, necesita el número octal 4 para cada uno (100). Esto le da tres dígitos octales, 544, que se traduce en los dígitos binarios 101 100 100.

```
$ chmod 544 misdatos
```

Permisos de ejecución

Uno de los usos más comunes de la máscara binaria es configurar los permisos de ejecución. Se crean archivos que contienen comandos de Linux, llamados *secuencias de comandos de shell*. Para ejecutar los comandos en una secuencia de comandos de shell, primero debe indicar que el archivo es ejecutable (que contiene comandos que el sistema ejecuta). Se hace esto de varias formas, una de las cuales consiste en establecer el permiso de ejecución en el archivo de secuencia de comandos de shell. Suponga que acaba de completar un archivo de secuencia de comandos de shell y necesita otorgarle permiso de ejecución. También quiere retener permisos de lectura y escritura pero negar cualquier acceso al grupo u otros usuarios. El dígito octal 7 (111) configurará los tres permisos, incluido el de ejecución (también se suma 4-lectura, 2-escritura y 1-ejecución para obtener 7). Al utilizar 0 para el grupo y otros usuarios se les niega el acceso. Esto le da los dígitos 700, que son equivalentes a los dígitos binarios 111 000 000. En este ejemplo, se establece el permiso de propietario para el archivo **miprog** para incluir el permiso de ejecución:

```
$ chmod 700 miprog
```

Si quiere que otros sean capaces de ejecutar y leer el archivo pero no de cambiarlo, puede establecer permisos de lectura y ejecución y desactivar el permiso de escritura con el dígito 5 (101). En este caso, se utilizan los dígitos octales 755, que tiene el equivalente binario de 111 101 101.

```
$ chmod 755 myprog
```

Permisos de directorio

También se pueden establecer permisos en directorios. El permiso de lectura que se establece en un directorio permite que se despliegue una lista de archivos en un directorio. El permiso de ejecución le permite al usuario cambiar a ese directorio. El permiso de escritura permite al usuario crear y eliminar sus archivos en el directorio. Si permite que otros usuarios tengan permisos de escritura en un directorio, éstos pueden agregar sus propios archivos al directorio. Cuando crea un directorio, automáticamente se le dan permisos de lectura, escritura y ejecución para el propietario. Puede desplegar una lista de los archivos en tal directorio, cambiarlos y crear archivos en éste.

Al igual que los archivos, los directorios tienen conjuntos de permisos para propietario, grupo y todos los demás usuarios. A menudo, tal vez quiera permitir que otros usuarios cambien y desplieguen listas de los archivos en su directorio, sin dejar que agreguen sus propios archivos. En este caso, se establecen permisos de lectura y ejecución en ese directorio, pero no se establece un permiso de escritura. Esto permite que otros usuarios cambien el directorio y desplieguen listas de los archivos, sin crear nuevos archivos o copiar ninguno de sus archivos en éste. En el siguiente ejemplo se establece un permiso de lectura y ejecución para el grupo en el directorio **gracias** pero elimina el permiso de escritura. Los miembros del grupo pueden entrar en el directorio **gracias** y desplegar una lista de los archivos, pero no pueden crear nuevos archivos.

```
$ chmod g+rx-w cartas/gracias
```

Al igual que con archivos, también puede utilizar dígitos octales para configurar el permiso de un directorio. Para configurar el mismo permiso como en el ejemplo anterior, se utilizan los dígitos octales 750, que tienen los equivalentes binarios 111 101 000.

```
$ chmod 750 cartas/gracias
```

Despliegue de los permisos de directorio

El comando **ls** con la opción **-l** muestra una lista de todos los archivos de un directorio. Para mostrar sólo la información del propio directorio, agregue el modificador **d**. En el siguiente ejemplo, **ls -ld** despliega información del directorio **gracias**. Observe que el primer carácter de la lista de permisos es **d**, que indica que es un directorio:

```
$ ls -ld gracias
drwxr-x--- 2 carlos 512 Feb 10 04:30 gracias
```

Permisos del directorio principal

Si tiene un archivo al que quiere que otros usuarios tengan acceso, no sólo necesita establecer permisos para ese archivo, sino también debe asegurarse de que establecen los permisos para el directorio en el que está ubicado el archivo. Para acceder a su archivo, un usuario debe acceder primero al directorio en que se encuentra el archivo. Lo mismo aplica a los directorios principales. Aunque un directorio puede dar permiso de acceso a otros, si su directorio principal niega el acceso, no se puede acceder a él. Por tanto, debe prestar mucha atención a su árbol de directorios. Para proporcionar acceso a un directorio, todos los directorios que están arriba de éste en el árbol de directorios también deben ser accesibles para otros usuarios.

Permisos de propietario

Además de los permisos de lectura, escritura y ejecución, también se configuran permisos de propietario para programas ejecutables. Por lo general, el usuario que ejecuta un programa es el propietario mientras se ejecuta, aunque el propietario del propio archivo de programa sea otro. El permiso Set User ID establece que el propietario original del programa será siempre su propietario, aunque otro usuario esté ejecutando el programa. Por ejemplo, el usuario root es el propietario de casi todo el software del sistema, pero los usuarios ordinarios pueden ejecutarlo. En este caso, el usuario ordinario necesita ejecutar ese programa, mientras el root retiene la propiedad para que el programa tenga los permisos de cambiar esos archivos que son propiedad del root. Los permisos Group ID funcionan de la misma manera, excepto en el caso de grupos. Los programas que pertenecen a cierto grupo retienen sus permisos de propietario, aunque los ejecuten usuarios en otro grupo. Entonces el programa cambia los archivos del grupo propietario. Existe un riesgo de seguridad implícito porque, en esencia, se le da a un usuario acceso limitado de nivel root.

Permisos de propietario mediante símbolos

Para agregar los permisos User ID y Group ID a un archivo, se utiliza la opción **s**. En el siguiente ejemplo se agrega el permiso User ID al programa **pppd**, que pertenece al usuario root. Cuando un usuario ordinario ejecuta **pppd**, el usuario root se mantiene como propietario, permitiendo que el programa **pppd** cambie los archivos que pertenecen al root.

```
# chmod +s /usr/sbin/pppd
```

Los permisos Set User ID y Set Group ID se muestran como **s** en la posición de ejecución de los segmentos de propietario y grupo. Set User ID y Set Group ID son, en esencia, variaciones de permisos de ejecución, **x**. Los permisos de lectura, escritura y User ID son **rws** en lugar de **rwX**.

```
# ls -l /usr/sbin/pppd
-rwsr-sr-x 1 root root 184412 Jan 24 22:48 /usr/sbin/pppd
```

Permisos de propietario con el método binario

En el caso de los permisos de propietario, se agrega otro número octal al principio de los dígitos octales. El dígito octal para el permiso User ID es 4 (100) y para el Group ID es 2 (010) (utilice 6 para configurar ambos, 110). En el siguiente ejemplo se configuran permisos de User ID para el programa **pppd**, junto con permisos de lectura y ejecución para propietario, grupo y otros:

```
# chmod 4555 /usr/sbin/pppd
```

Permisos sticky bit

Otro permiso especial proporciona mayor seguridad en directorios, *sticky bit*. Originalmente, sticky bit se utilizó para mantener un programa en memoria después de que terminó su ejecución para incrementar la eficiencia. En la actualidad, los sistemas Linux ignoran esta característica. En cambio, se utiliza en directorios para proteger archivos dentro de éstos. En el caso de los archivos de un directorio con sticky bit establecido, sólo el usuario root o el propietario del directorio puede eliminarlos o cambiarles el nombre.

Permiso sticky bit con símbolos

El símbolo del permiso sticky bit es **t**. Sticky bit se muestra como una **t** en la posición de ejecución de los permisos de otros. Un programa con permisos de lectura y ejecución con sticky bit tiene sus permisos desplegados como **r-t**.

```
# chmod +t /home/daniel/misinformes
# ls -l /home/daniel/misinformes
-rwxr-xr-t 1 root root 4096 /home/daniel/misinformes
```

El permiso sticky bit al utilizar el método binario

Al igual con los de propietario, con los permisos sticky bit se agrega otro número al principio de los dígitos octales. El dígito octal de sticky bit es 1 (001). En el siguiente ejemplo se configura sticky bit para el directorio **misinformes**:

```
# chmod 1755 /home/daniel/misreportes
```

En el siguiente ejemplo se establecen los permisos sticky bit y User ID en el directorio **nuevosprog**. El permiso 5755 tiene el equivalente binario 101 111 101 101:

```
# chmod 5755 /usr/bin/nuevosprog
# ls -l /usr/bin/nuevosprog
drwsr-xr-t 1 root root 4096 /usr/bin/nuevosprog
```

Permisos predeterminados: umask

Siempre que crea un archivo o directorio, se otorgan permisos predeterminados. Puede desplegar los permisos predeterminados actuales o cambiarlos con el comando **umask**. Los permisos se despliegan en formato binario o simbólico, como se describe en las siguientes secciones. Entre los permisos predeterminados se incluye cualquier permiso de ejecución que se aplica a un directorio. Los permisos de ejecución de un archivo se desactivan como opción predeterminada cuando lo crea, porque los archivos de datos estándar no utilizan los permisos de ejecución (para que un archivo, como una secuencia de comandos, sea ejecutable, tiene que configurar manualmente su permiso de ejecución). Para desplegar los permisos predeterminados actuales, se utiliza el comando **umask** sin argumentos. La opción **-S** utiliza el formato simbólico.

```
$ umask -S
u=rwx,g=rx,o=rx
```

Este comando umask predeterminado proporciona permisos **rw-r--r--** para archivos estándar y agrega permisos de ejecución para directorios, **rw-r-xr-x**.

Puede establecer un nuevo valor predeterminado al especificar permisos en formato simbólico o binario. Para especificar los nuevos permisos, utilice la opción **-S**. En el siguiente ejemplo se niega a otros el permiso de lectura, mientras que se permite el acceso de lectura a usuario y grupo, que da como resultado permisos **rw-r-x---**:

```
$ umask -S u=rwx,g=rx,o=
```

Cuando se utiliza el formato binario, la máscara es lo inverso de los permisos que quiere. Para establecer permisos de lectura y ejecución y desactivar el permiso de escritura, se utiliza el número octal 2, un binario 010. Para activar todos los permisos, se utiliza un octal 0, binario 000. En el

siguiente ejemplo se muestra la máscara de las opciones predeterminadas de permisos `rwX`, `rx` y `rx` (`rw`, `r` y `r` para archivos):

```
$ umask
0022
```

Para establecer como opción predeterminada que sólo se nieguen todos los permisos para otros, debe usar `0027`, utilizando la máscara binaria `0111` para los otros permisos.

```
$ umask 0027
```

Cuotas de disco

Puede utilizar cuotas de disco para controlar cuánto espacio en disco utiliza un usuario particular en su sistema. En su sistema Linux, el espacio no utilizado en disco se almacena en un recurso común al que cada usuario accede cuando lo necesita. Mientras los usuarios crean más archivos, toman el espacio que necesitan de un almacén de espacio disponible en disco. En este sentido, todos los usuarios comparten un solo recurso de espacio en disco no utilizado. Sin embargo, si un usuario fuera a utilizar todo el espacio sobrante, ninguno de los demás usuarios podría crear archivos o incluso ejecutar programas. Para contrarrestar este problema, se crean cuotas de disco en usuarios particulares, que limitan la cantidad de disco disponible que pueden utilizar.

Herramientas de cuotas

Las revisiones de cuota se implementan en el sistema de archivos de una partición de disco duro montada en su sistema. Las cuotas se habilitan al utilizar los programas **quotacheck** y **quotaon**. Se ejecutan en la secuencia de comandos `/etc/rc.d/rc.sysinit`, que se lanza siempre que quiere iniciar su sistema. Cada partición necesita montarse con las opciones de cuota, **usrquota** o **grpquota**. La primera permite controles de cuota para usuarios y la segunda funciona con grupos. Estas opciones suelen colocarse en la entrada mount del archivo `/etc/fstab` para una partición particular. Por ejemplo, para montar la partición de disco duro `/dev/hda6` que está montada en el directorio `/home` con soporte para cuotas de grupo y usuarios, necesita una entrada de montaje como la siguiente:

```
/dev/hda6 /home ext2 defaults, usrquota,grpquota 1 1
```

También necesita crear archivos **quota.user** y **quota.group** en cada partición para la que se habilitan cuotas. Son las bases de datos de cuotas que almacena la información de cuota para cada usuario y grupo. Se crean estos archivos al ejecutar el comando **quotacheck** con la opción **-a** o el nombre de dispositivo del sistema de archivo donde quiere habilitar cuotas. En el siguiente ejemplo se crea una base de datos de cuota en la partición de disco duro `hda1`:

```
quotacheck -a /dev/hda1
```

edquota

Puede establecer cuotas de disco al utilizar el comando **edquota**. Con éste, se accede al registro de cuotas de un usuario y grupo particular, que se mantiene en la base de datos de cuota de disco. También puede establecer cuotas predeterminadas que se aplicarán a cualquier usuario o grupo en el sistema de archivos para el que no se han establecido cuotas. **edquota** abrirá el registro en su editor predeterminado y puede utilizar éste para hacer cualquier cambio. Para abrir el registro para un usuario en particular, utilice la opción **-u** y el nombre de usuario como argumento para

Opciones de edquota	Descripción
-u	Edita la cuota de usuario. Es la opción predeterminada.
-g	Edita la cuota de grupo.
-p	Duplica las cuotas del usuario típico especificado. Es el mecanismo normal que se utiliza para inicializar cuotas para grupos de usuarios.
-t	Edita los límites de tiempo flexible para cada sistema de archivos.

TABLA 28-6 Opciones para **edquota**

edquota (consulte la tabla 28-6). En el siguiente ejemplo se abre el registro de cuota de disco para el usuario **larisa**:

```
edquota -u larisa
```

El límite que establezca para una cuota puede ser estricto o flexible. Un límite estricto le negará al usuario la capacidad de exceder esta cuota, mientras que un límite flexible sólo enviará una advertencia. Para el límite flexible, se designa un periodo de gracia durante el cual el usuario tiene la oportunidad de reducir su espacio en disco bajo el límite. Si el espacio en disco aún excede el límite después de que expira el periodo de gracia, se le niega el acceso a su cuenta al usuario. Por ejemplo, un límite flexible suele ser de 75MB, mientras que un límite estricto puede ser de 100MB. Los usuarios que exceden este límite flexible tienen un periodo de gracia de 48 horas.

El registro de cuota comienza con el nombre del dispositivo de disco duro y los bloques de memoria e inodos en uso. Los segmentos de límite tienen parámetros para límites flexibles y estrictos. Si estas entradas son 0, no hay límites. Puede establecer límites flexibles y estrictos, empleando el límite estricto como una restricción firme. Los bloques en Linux son en la actualidad de unos 1000 bytes. Los archivos usan los inodos para almacenar información acerca de los bloques de memoria que integran un archivo. Para configurar el límite de tiempo para un límite flexible, se utiliza el comando **edquota** con la opción **-t**. En el siguiente ejemplo se despliega el registro de cuotas para **larisa**:

```
Quotas for user larisa:
/dev/hda3: blocks in use: 9000, limits (soft = 40000, hard = 60000)
inodes in use: 321, limits (soft = 0, hard = 0)
```

quotacheck, quotaon y quotaoff

Los registros de cuota se mantienen en la base de datos de cuota de la partición. Cada partición que tiene cuotas habilitadas cuenta con su propia base de datos de cuota. Se revisa la validez de su base de datos de cuota con el comando **quotacheck**. Se activan o desactivan las cuotas al utilizar los comandos **quotaon** y **quotaoff**. Cuando inicia su sistema, **quotacheck** se ejecuta para revisar las bases de datos de cuotas, y después **quotaon** se ejecuta para activar las cuotas.

repquota y quota

Como administrador del sistema, puede utilizar el comando **repquota** para generar un resumen del uso de disco para un sistema de archivos especificado, que revisa cuáles usuarios se están

Opciones de cuota	Descripción
-g	Imprime las cuotas para el grupo al que pertenece el usuario.
-u	Imprime la cuota del usuario
-v	Despliega las cuotas del sistemas de archivo donde no hay nada almacenado.
-q	Imprime información en el sistema de archivos donde el uso está arriba de la cuota.

TABLA 28-7 Opciones de cuota

acercando al límite de cuota, o excediéndolo. **repquota** toma como argumento el sistema de archivos que se revisa; la opción **-a** revisa todos los sistemas de archivos.

```
repquota /dev/hda1
```

Los usuarios individuales utilizan el comando **quota** para revisar su uso de memoria y cuánto espacio en disco les queda en su cuota (consulte la tabla 28-7).

Protocolo ligero de acceso directo

El protocolo ligero de acceso directo (LDAP, Lightweight Directory Access Protocol) está diseñado para implementar directorios de información que es accesible en red. En este contexto, el término directorio está definido como una base de datos de información que es principalmente de sólo lectura, simple, pequeña, de amplio acceso y rápida distribución. No está diseñado para transacciones o actualizaciones. Se usa principalmente para proporcionar información acerca de los usuarios en una red, que provee información acerca de éstos, como su dirección de correo electrónico o número de teléfono. Estos directorios también se utilizan para fines de autenticación, identificando que cierto usuario pertenece a una red específica. Encontrará más información acerca de LDAP en ldapman.org. Considere que un directorio LDAP para usuarios es una libreta telefónica que se accede en Internet, donde cualquiera puede buscar direcciones de correo electrónico u otra información. En realidad, tal vez sea más acertado referirse a tales directorios como bases de datos. Son bases de datos de información de usuario, que se acceden en redes como Internet. Por lo general, los usuarios de una red local se extienden a través de varios sistemas diferentes, y para obtener información acerca de un usuario, tiene que saber en qué sistema está un usuario y después consultar ese sistema. Con LDAP, la información de todos los usuarios de una red se almacena en el servidor LDAP. Sólo tiene que consultar al servidor LDAP de la red para obtener información acerca de un usuario. Por ejemplo, Sendmail utiliza LDAP para buscar direcciones de usuario. También se utiliza Firefox o Netscape para consultar LDAP.

NOTA LDAP es un protocolo de acceso de directorio a un servicio de directorio X.500, el servicio de directorio OSI.

Clientes y servidores de LDAP

Los directorios LDAP se implementan como clientes y servidores, donde se utiliza un cliente LDAP para acceder a un servicio que administra la base de datos LDAP. Casi todas las distribuciones de Linux utilizan OpenLDAP, una versión de fuente abierta de LDAP (aprenderá más acerca de

OpenLDAP en openldap.org). Este paquete incluye un servidor LDAP (**slapd**), un servidor de réplica LDAP (**slurpd**), un cliente LDAP y herramientas. **slurpd** se utiliza para actualizar otros servidores LDAP en su red, si es que tiene más de uno.

Archivos de configuración de LDAP

Todos los archivos de configuración de LDAP se almacenan en el directorio **/etc/openldap**. Entre éstos se incluyen **slapd.conf**, el archivo de configuración de servidor LDAP, y **ldap.conf**, el archivo de configuración de herramientas y clientes de LDAP. Para habilitar el servidor LDAP, tiene que editar manualmente el archivo **slapd.conf** y cambiar el valor de dominio (dc) para el sufijo y las entradas **rootdn** para su propia dirección de dominio de red. Se trata de la red a la que dará servicio el servidor LDAP.

Para habilitar clientes LDAP y sus herramientas, tiene que especificar la dirección de dominio propia en el archivo **ldap.conf** en la opción **BASE**, junto con la dirección del servidor en la opción **HOST** (nombre de dominio o dirección IP). En el caso de clientes, puede editar directamente el archivo de configuración **ldap.conf** o utilizar la herramienta Configuración de la autenticación, al hacer clic en el botón Configurar LDAP, en el panel Información de usuario o Autenticación. Aquí se inserta su nombre de dominio y la dirección del servidor LDAP. Consulte la entrada de **Man ldap.conf** para conocer descripciones detalladas de las opciones de LDAP.

SUGERENCIA Tenga en cuenta que los archivos **/etc/ldap.conf** y **/etc/openldap/ldap.conf** no son iguales: **/etc/ldap.conf** se utiliza para configurar LDAP para soporte de Nameservice Switch y PAM, mientras que **/etc/openldap/ldap.conf** se utiliza para todos los clientes LDAP.

Configuración del servidor LDAP: **/etc/slapd.conf**

Se configura el servidor LDAP con el archivo **/etc/slapd.conf**. Aquí encontrará entradas para cargar esquemas y para especificar controles de acceso, la base de datos del directorio y contraseñas. El archivo se comenta en detalle, con diversas configuraciones para casi todas las opciones, aunque tendrá que insertar configuraciones para varias. En primer lugar, necesita especificar su sufijo de dominio y el administrador de dominio root. Aquí se muestran las configuraciones predeterminadas:

```
suffix          "dc=my-domain,dc=com"
rootdn          "cn=Manager,dc=my-domain,dc=com"
```

En este ejemplo, **suffix** se cambia a **mipista**, para **mipista.com**. **rootdn** permanece igual.

```
suffix          "dc=mipista,dc=com"
rootdn          "cn=Manager,dc=mipista,dc=com"
```

Después tendrá que especificar una contraseña con **rootpw**. Existen entradas para versiones de texto simple y cifradas. Ambas se comentan. Elimine el comentario de una. En el siguiente ejemplo se utiliza la opción de contraseña simple, "secreta":

```
rootpw          secreta
# rootpw        {crypt}ijFYNcSNctBYg
```

En el caso de una contraseña cifrada, primero puede crear la versión cifrada con **slappasswd**. Esto generará una cadena de texto de cifrado para la contraseña. Después se copia la cadena cifrada generada a la entrada **rootpw**. En GNOME, simplemente puede copiar y pegar de una ventana terminal al archivo **/etc/slapd.conf** en Editor de textos (Accesorios). También puede redirigir la

cadena cifrada a un archivo y leerlo después. El cifrado SSHA se utilizará como opción predeterminada.

```
# slappasswd
New password:
Re-enter new password:
{SSHA}0a+szaAwElK57Y8AoD5uMULSvLfCUfg5
```

Entonces la entrada de contraseña root **rootpw** se verá así:

```
rootpw {SSHA}0a+szaAwElK57Y8AoD5uMULSvLfCUfg5
```

Utilice la contraseña que insertó en la petición **slappasswd** para acceder a su directorio LDAP.

El archivo de configuración también presenta los esquemas que se utilizarán. Los esquemas se incluyen con la directiva **include**.

```
include /etc/openldap/schema/core.schema
include /etc/openldap/schema/cosine.schema
include /etc/openldap/schema/inetorgperson.schema
include /etc/openldap/schema/nis.schema
```

NOTA LDAP da soporte a capa simple de autenticación y seguridad (SASL, Simple Authentication and Security Layer) para una autenticación segura con métodos como MD5 y Kerberos.

Base de datos de directorio LDAP: Idif

Un registro (también conocido como entrada) de una base de datos LDAP comienza con un nombre, conocido como *nombre distintivo*, seguido por un conjunto de atributos y valores. El nombre distintivo identifica el registro de manera única. Por ejemplo, un nombre puede ser un nombre de usuario y el atributo podría ser la dirección de correo del usuario, mientras que la dirección es el valor del atributo. Los atributos permitidos se determinan mediante esquemas que se definen en el directorio **/etc/openldap/schema**. Este directorio almacenará varios archivos de definición de esquema, cada uno con una extensión **schema**. Algunos dependerán de otros, mejorando los atributos y las clases soportados. El núcleo básico de atributos se define en el archivo **core.schema**. Aquí encontrará definiciones de atributos como el nombre del país y la dirección. Otros esquemas, como **inetorgperson.schema**, especifican **core.schema** como un esquema dependiente, haciendo que sus atributos queden disponibles para las clases. El esquema **inetOrgPerson** también definirá sus propios atributos como **jpegPhoto** para la fotografía de una persona.

Atributos y clases de esquemas

Los atributos y las clases se definen oficialmente por especificaciones RFC que se presentan con cada entrada de atributo y clase en los archivos de esquema. Éstas son definiciones estandarizadas y no deben cambiarse. Los atributos contienen una definición **attributetype**. A cada una se le da un número de identificación único seguido por un nombre con el cual se hace referencia a éstas. Entre los campos se incluyen la descripción de atributo (DESC), características de búsqueda como EQUALITY y SUBSTR y los identificadores de objeto (SYNTAX). Consulte la guía administrativa OpenLDAP para conocer una descripción detallada.

```
attributetype ( 2.5.4.9 NAME ( 'calle' 'dirección' )
    DESC 'RFC2256: dirección de este objeto'
    EQUALITY caseIgnoreMatch
    SUBSTR caseIgnoreSubstringsMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{128} )
```

Una clase define el tipo de base de datos (directorio) que se crea. Ésta especificará los tipos de atributos que se incluyen en sus registros. Las clases pueden ser dependientes, de modo que una clase se vuelve una extensión de otra. La clase que se utiliza más a menudo en bases de datos LDAP es `inetOrgPerson`, que se define en el archivo `inetOrgPerson.schema`. El término `inetOrgPerson` viene de Internet Organization Person, porque muchos directorios LDAP realizan tareas de Internet. La clase se deriva de la clase `organizationalPerson` que se define en `core.schema`, que incluye los atributos originales de uso común, como direcciones y nombre.

```
# inetOrgPerson
# The inetOrgPerson represents people who are associated with an
# organization in some way. It is a structural class and is derived
# from the organizationalPerson which is defined in X.521 [X521].
objectclass ( 2.16.840.1.113730.3.2.2
    NAME 'inetOrgPerson'
    DESC 'RFC2798: Internet Organizational Person'
    SUP organizationalPerson
    STRUCTURAL
    MAY (
        audio $ businessCategory $ carLicense $ departmentNumber $
        displayName $ employeeNumber $ employeeType $ givenName $
        homePhone $ homePostalAddress $ initials $ jpegPhoto $
        labeledURI $ mail $ manager $ mobile $ o $ pager $
        photo $ roomNumber $ secretary $ uid $ userCertificate $
        x500uniqueIdentifier $ preferredLanguagei $
        userSMIMECertificate $ userPKCS12 )
    )
```

Puede crear sus propias clases, a partir de las estándares que ya están definidas. También puede crear sus propios atributos, pero cada atributo requerirá un identificador de objeto único (OID).

Nombres distintivos

Los datos en el directorio LDAP se organizan jerárquicamente, desde categorías generales hasta datos específicos. Luego, un directorio LDAP se organiza empezando con los países, reduciéndose a estados, después a organizaciones y sus subunidades y, por último, a individuos. Por lo general, los directorios LDAP se organizan junto con las líneas de dominios de Internet. En este formato, la categoría más alta es la extensión de nombre de dominio, como `.com` o `.ca`. Luego, se desglosa en la red (organización), las unidades y, al final, los usuarios.

Esta organización ayuda a definir los nombres distintivos que definirán registros de LDAP. En una organización basada en red, la organización de nivel más alto se define con un componente de dominio especificado por la clase `dcObject`, que incluye el atributo `domainComponent` (`dc`). Por lo general, se definen la red y la extensión como componentes de dominio para crear la organización de nivel superior que se vuelve el nombre distintivo para la propia base de datos.

```
dc=mipista, dc=com
```

Bajo el nombre de la organización se encuentra una unidad organizacional, como los usuarios. Éstos se definen como `organizationalUnitName` (`ou`), que es parte de la clase `organizationalUnit`. El nombre distintivo para la unidad organizacional del usuario sería:

```
ou=users, dc=mipista, dc=com
```

Bajo la unidad organizacional puede tener después usuarios individuales. Aquí el nombre de usuario está definido con el atributo `commonName (cn)`, que se utiliza en varias clases, incluida `Person`, que es parte de `organizationalPerson`, que a su vez es parte de `inetOrgPerson`. El nombre distintivo para el usuario **daniel** es entonces:

```
cn=daniel,ou=users,dc=mipista,dc=com
```

Entradas LDIF

Las entradas de base de datos se colocan en un archivo de formato de intercambio de LDAP (LDIF, LDAP Interchange Format). Este formato proporciona un estándar global que permite el acceso a una base de datos por parte de cualquier cliente compatible con LDAP. Un LDIF es un archivo de texto simple con una extensión **.ldif** colocado en el directorio **/etc/openldap**. Las entradas de un registro LDIF constan de un nombre distintivo o un atributo seguido por dos puntos y su lista de valores. Cada registro comienza con un nombre distintivo que sólo identifica el registro. Los atributos van después. Puede considerar que el nombre es un registro y los atributos son campos en ese registro. Se termina el registro con una línea vacía.

Al principio, se crea un archivo LDIF con cualquier editor de texto, y luego se insertan los registros. En el siguiente ejemplo, el archivo **mipista.ldif** LDIF contiene registros de usuarios en la red.

En primer lugar, se crean registros que definen su organización y unidades de organización. Estos nombres distintivos se utilizarán en registros en el nivel de usuario. También tiene que especificar un administrador para la base de datos, en este caso `Manager`. Asegúrese de incluir las clases de objeto apropiadas. La organización utiliza `dcObject` (objeto de componente de dominio) y objetos de organización. El `Manager` utiliza `organizationalRole` y `users` utiliza `organizationalUnit`. Dentro de cada registro se tienen definiciones de atributos, como el de organización, **o**, en el primer registro, en `Mipista`.

```
dn: dc=mipista,dc=com
objectclass: dcobject
objectclass: organization
dc: mipista
o: Mipista

dn: cn=Manager,dc=mipista,dc=com
cn: Manager
objectclass: organizationalRole

dn: ou=users,dc=mipista,dc=com
objectclass: organizationalUnit
ou: users
```

Luego siguen los registros individuales, como en el siguiente ejemplo para **daniel**. Aquí las clases de objeto son `organizationalPerson` e `inetOrgPerson`. Después van los atributos, como nombre común (**cn**), ID de usuario (**uid**), organización (**o**), apellido (**sn**) y `street`.

```
dn: cn=daniel,ou=users,dc=mipista,dc=com
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn:daniel
uid:daniel
o: Mipista
sn: prado
street: ave saturno 77777
```


Aquí se muestra un ejemplo de un archivo LDIF. La organización es mipista.com. Aquí existen dos registros, uno para **daniel** y otro para **carlos**:

mytrek.ldif

```
dn: dc=mytrek,dc=com
objectclass: dcobject
objectclass: organization
dc: mytrek
o: MyTrek

dn: cn=Manager,dc=mytrek,dc=com
cn: Manager
objectclass: organizationalRole

dn: ou=users,dc=mytrek,dc=com
objectclass: organizationalUnit
ou: users

dn: cn=dylan,ou=users,dc=mytrek,dc=com
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: dylan
uid: dylan
o: MyTrek
sn: shark
street: 77777 saturn ave

dn: cn=chris,ou=users,dc=mytrek,dc=com
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: chris
uid: chris
o: MyTrek
sn: dolphin
street: 99999 neptune way
```

Adición de registros

Una vez que ha creado su archivo LDIF, puede utilizar el comando **ldapadd** para agregar registros a su directorio LDAP. Utilice la opción **-D** para especificar el directorio al que agregará los registros, y la opción **-f** para especificar el archivo LDIF de donde se leerán. Se puede utilizar **ldapadd** para insertar campos directamente. La opción **-x** indica que se utilice acceso de contraseña simple, **-W** pedirá una contraseña y la opción **-D** especifica el administrador del directorio.

```
# ldapadd -x -D "cn=Manager,dc=mipista,dc=com" -W -f mipista.ldif
```

Enter LDAP Password:

```
adding new entry "dc=mipista,dc=com"
```

```
adding new entry "cn=Manager,dc=mipista,dc=com"
```

```
adding new entry "ou=users,dc=mipista,dc=com"
```

```
adding new entry "cn=daniel,ou=users,dc=mipista,dc=com"
```

```
adding new entry "cn=carlos,ou=users,dc=mipista,dc=com"
```

Asegúrese de reiniciar el servidor LDAP para que surtan efecto sus cambios.

Búsqueda de LDAP

Una vez que ha agregado sus registros, se utiliza el comando **ldapsearch** para buscar su directorio LDAP. Las opciones **-x** y **-W** proporcionan un acceso de contraseña simple y la opción **-b** especifica la base de datos LDAP que se usará. Después de las opciones están los atributos que se buscarán, en este caso el atributo **street**.

```
# ldapsearch -x -W -D 'cn=Manager,dc=mipista,dc=com' -b 'dc=mipista,dc=com' street
Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=mipista,dc=com> with scope sub
# filter: (objectclass=*)
# requesting: street

# daniel, users, mipista.com
cn=daniel,ou=users,dc=mipista,dc=com
street: ave saturno 77777

# carlos, users, mipista.com
cn=carlos,ou=users,dc=mipista,dc=com
street: calle neptuno 99999

# search result
search: 2
Result: 0 Success

# numResponses: 6
# numEntries: 5
```

Si quiere ver todos los registros de la base de datos, se utiliza el mismo comando de búsqueda sin ningún atributo.

Herramientas LDAP

Para crear o cambiar entradas en la base de datos LDAP, se utilizan las utilerías **ldapadd** y **ldapmodify**. Con **ldapdelete**, se eliminan entradas. Una vez que ha creado una base de datos LDAP, puede consultarse, a través del servidor LDAP, con **ldapsearch**. En el caso del servidor LDAP, puede crear un archivo de texto de entradas LDAP al utilizar el formato de intercambio de datos de LDAP (LDIF, Data Interchange Format LDAP). Luego se leen todos estos archivos de texto al mismo tiempo en la base de datos LDAP al utilizar la herramienta **slapadd**. La herramienta **slapcat** extrae entradas de la base de datos LDAP y las guarda en un archivo LDIF. Para volver a indizar adiciones y cambios, se utiliza la utilería **slapindex**. Consulte el HOWTO de LDAP en el proyecto de documentación de Linux para conocer más detalles sobre el uso y la configuración de bases de datos LDAP como libros de direcciones (tldp.org).

LDAP y PAM

Con LDAP, también puede controlar con más cuidado el tipo de información que se da y a quién. Al utilizar el módulo PAM (**pam_ldap**), LDAP realiza tareas de autenticación de usuario, que proporciona autenticación centralizada para usuarios. Las operaciones de inicio de sesión que realizan los usuarios para diferentes servicios como inicios de sesión de servidor de correo POP, de sistema y de Samba se llevan a cabo a través de LDAP al utilizar un solo ID de usuario y una sola contraseña asegurados por PAM. Para utilizar LDAP en autenticación, necesita configurar PAM para utilizarlo, además de migrar archivos de autenticación al formato LDAP. El directorio **/usr/share/openldap/migration** almacena secuencias de comandos que se utilizan para traducir los archivos antiguos en versiones de LDAP.

LDAP y el servicio Name Service Switch

Con el módulo **libnss_ldap**, LDAP también se utiliza en el servicio Nameservice Switch (NSS) junto con archivos NIS y de sistema para servicios de base de datos de sistema como contraseñas y grupos. Los clientes habilitan de manera sencilla LDAP para NSS, al utilizar la herramienta Configuración de la autenticación y seleccionar Habilitar el soporte LDAP, en el panel Información de usuario. También necesita asegurarse de que se especifica el servidor LDAP. También se agrega manualmente **ldap** para entradas en el archivo **/etc/nsswitch.conf**.

SUGERENCIA Para asegurar mejor el acceso al servidor LDAP, debe cifrar su contraseña del administrador de LDAP. Éste se especifica en la entrada **rootdn**, y la contraseña en la entrada **rootpw**. Para crear una contraseña cifrada, se utiliza el comando **slappasswd**, que le pide una contraseña y despliega la versión cifrada. Copie esa versión cifrada en la entrada **rootpw**.

Habilitación de LDAP en Thunderbird

En Thunderbird, abra la libreta de direcciones y seleccione Archivo | Nuevo, y elija Directorio LDAPD. Aquí se inserta el servidor LDAP. Esto despliega un panel donde se inserta el nombre de la libreta de direcciones, el nombre de host del servidor LDAP, la base DN en que se buscará y el número de puerto, como 389.

Pluggable Authentication Modules

Pluggable Authentication Modules (PAM) es un servicio de autenticación que permite a un sistema determinar el método de autenticación que utilizarán los usuarios. Lo tradicional es que, en un sistema Linux, la autenticación se realice al buscar contraseñas. Cuando un usuario inicia sesión, este proceso busca su contraseña en el archivo de contraseñas. Con PAM, las solicitudes de autenticación por parte del usuario se dirigen a PAM, que a cambio utiliza un método especificado para autenticar al usuario. Podría tratarse de una búsqueda de contraseña simple o una solicitud a un servidor LDAP, pero es PAM el que proporciona autenticación, no una búsqueda de contraseña directa por el usuario o aplicación. En este aspecto, la autenticación se vuelve más centralizada y se controla con un servicio específico, PAM. El administrador del sistema configura dinámicamente los procedimientos de actualización reales. La autenticación se realiza mediante módulos que varían de acuerdo con el tipo de autenticación necesario. Un administrador agrega o reemplaza módulos al cambiar simplemente los archivos de configuración de PAM. Consulte el sitio Web de PAM en kernel.org/pub/linux/libs/pam para conocer más información y una lista de módulos de PAM. Los módulos de PAM se ubican en el directorio **/lib/security**.

Archivos de configuración de PAM

PAM utiliza diferentes archivos de configuración para diferentes servicios que requieren autenticación. Estos archivos de configuración se almacenan en el directorio `/etc/pam.d`. Por ejemplo, tiene un archivo de configuración para iniciar sesión en su sistema (`/etc/pam.d/login`), uno para inicio de sesión gráfico (`/etc/pam.d/gdm`) y uno para acceder a su servidor de Samba (`/etc/pam.d/samba`). Un archivo de configuración de PAM predeterminado, llamado `/etc/pam.d/other`, se invoca si no hay un archivo de servicios presente. El archivo `system-auth` contiene módulos de autenticación estándar para servicios de sistema.

Módulos de PAM

Un archivo de configuración de PAM contiene una lista de módulos que habrán de utilizarse para autenticación. Tienen el siguiente formato:

```
tipo-modulo marca-control ruta-modulo argumentos-modulo
```

Ruta-modulo es el módulo que se ejecuta y *argumentos-modulo* son los parámetros que quiere pasar a ese módulo. Aunque existen algunos argumentos genéricos, casi todos los módulos tienen los propios. *Tipo-modulo* se refiere a grupos diferentes para administración de autenticación: cuenta, autenticación, sesión y contraseña. La administración de cuentas realiza una verificación de cuentas, que revisa aspectos como si el usuario tiene acceso, o si la contraseña ha caducado. La autenticación (**auth**) verifica quién es el usuario, por lo general mediante una confirmación de contraseña. La administración de contraseñas realiza actualización de éstas como cambios de contraseña. La administración de contraseñas alude a las tareas realizadas antes de que se acceda a un servicio y antes de que se apague. Estas tareas incluyen iniciar un registro de actividad del usuario o montar o desmontar directorios de inicio.

SUGERENCIA Como una opción al directorio `letclpam.d`, se crea un archivo de configuración llamado `letclpam.conf`. Las entradas en este archivo tienen un campo de servicio, que hace referencia a la aplicación para la que el módulo se utiliza. Si existe el directorio `letclpam.d`, `letclpam.conf` se ignora automáticamente.

El campo *marca-control* indica cómo responde PAM si el módulo falla. El control es una directiva simple o una respuesta más complicada que especifica códigos de devolución como `open_err` con las acciones que habrán de tomarse. Las directivas simples son **requisite**, **required**, **sufficient** y **optional**. La directiva **requisite** termina el proceso de autenticación de inmediato si falla éste en el módulo. La directiva **required** sólo termina la autenticación después de que los módulos restantes se ejecutan. La directiva **sufficient** indica que el éxito de este módulo es suficiente para proporcionar autenticación a menos que un módulo obligatorio haya fallado antes. La directiva **optional** indica que no es obligatorio el éxito del módulo, salvo que sea el único módulo de autenticación para este servicio. Si especifica códigos de devolución, se mejoran las condiciones para el éxito o la falla de la autenticación. A los códigos de devolución se les dan valores como **die** u **ok**. Al código de regreso `open_err` se le puede dar la acción **die**, que detiene toda la autenticación y regresa una falla. Aquí se muestra el archivo de configuración `/etc/pam.d/vsftpd` para el servidor FTP:

```
#%PAM-1.0
auth required pam_listfile.so item=user sense=deny
                        file=/etc/vsftpd.ftpusers onerr=succeed
```

```
auth    required pam_stack.so service=system-auth
auth    required pam_shells.so
account required pam_stack.so service=system-auth
session required pam_stack.so service=system-auth
```

NOTA Los usuarios proporcionan mejor control de sus archivos y directorios al utilizar listas de control de acceso (ACL, Access Control Lists). Los sistemas de archivo necesitan montarse con la opción **acl**. Las herramientas de ACL (paquete **acl**) incluyen comandos **setfacl** y **getfacl** para establecer permisos. El comando **getfacl** mostrará una lista de permisos de usuario, propietario y grupo. **setfacl** controla el acceso por parte de usuarios específicos, al establecer permisos de lectura, escritura y ejecución. Utilice la opción **--set** para agregar nuevos permisos y la opción **-m** para cambiar los actuales (**-d** crea una configuración predeterminada). Se hace referencia a los usuarios y permisos con una lista separada por dos puntos con permisos específicos al utilizar las opciones **r**, **w** y **x**, y **u**, **g** y **o** para hacer referencia a las categorías usuario, grupo y otros. El argumento **u:carlos:rw** permitirá que el usuario carlos tenga permisos de lectura y escritura para el archivo especificado (véanse las páginas Man de **setfacl** y **getacl** para conocer más información).

Sistemas de archivo

Los archivos residen en dispositivos de almacenamiento físicos como discos duros, CD-ROM o discos flexibles. Los archivos en cada dispositivo de almacenamiento se organizan en sistemas de archivos. Los dispositivos de almacenamiento en sistemas Linux están presentes como una colección de sistemas de archivos que puede administrar. Cuando quiera agregar un dispositivo de almacenamiento nuevo, necesita formatearlo como un sistema de archivos y después adjuntarlo a su estructura de archivos de Linux. Los discos duros se dividen en dispositivos de almacenamiento separados denominados *particiones*, y cada uno tiene su propio sistema de archivos. Puede realizar tareas administrativas en sus sistemas de archivos, como crear copias de seguridad, adjuntarlos o separarlos de su estructura de archivos, formatear nuevos dispositivos o eliminar antiguos y revisar un sistema de archivos para ver si tiene problemas.

Para acceder a los archivos de un dispositivo, se adjunta su sistema de archivos a un directorio específico. A esto se le denomina *montar* el sistema de archivos. Por ejemplo, para acceder a archivos en un disco flexible, primero se monta su sistema de archivos a un directorio particular. Con Linux, puede montar varios tipos de archivos de sistema. Incluso puede acceder a particiones de discos duros de Windows o unidades de cinta, además de sistemas de archivo en un servidor remoto.

Los sistemas de archivos desarrollados recientemente ahora dan soporte a *registro por diario*, que le permite a su sistema recuperarse de manera sencilla de interrupciones o bloqueos. Los sistemas de archivos **ext3**, ReiserFS, XFS y JFS (IBM) mantienen un registro de cambios en archivos y directorios, llamado *diario*, que se utiliza para recuperar archivos y directorios en uso cuando el sistema se bloquea debido a eventos no previstos como interrupciones de energía. Casi todas las distribuciones utilizan actualmente el sistema de archivos **ext3** de manera predeterminada, aunque también tiene la opción de utilizar ReiserFS o JFS, como un sistema de registro por diario desarrollado independientemente.

Su sistema Linux es capaz de manejar cualquier variedad de dispositivos de almacenamiento que puedan conectarse a éste. Puede configurar su sistema para acceder a varios discos duros, particiones en un disco duro, discos CD-ROM, DVD, discos flexibles e incluso cintas. Tiene la opción de adjuntar estos componentes de almacenamiento manualmente o hacer que se monten de manera automática cuando arranca su sistema. Los montajes automáticos se manejan con el archivo de configuración **/etc/fstab**. Por ejemplo, las particiones principales que almacenan sus programas del sistema Linux se montan automáticamente siempre que arranca, mientras que un disco flexible se monta de manera manual cuando inserta uno en su unidad de discos flexibles, aunque éstos pueden montarse de manera automática. Los dispositivos de almacenamiento extraíbles como CD-ROM,

además de dispositivos como cámaras e impresoras USB, ahora se manejan con udev y Hardware Abstract Layer (HAL), como se describe en el capítulo 32 y se analiza parcialmente en éste.

Sistemas de archivos

Aunque todos los archivos de su sistema Linux se conectan en un árbol de directorios general, las partes de este árbol pueden residir en dispositivos de almacenamiento diferentes como discos duros o CD-ROM. Los archivos de un dispositivo de almacenamiento particular se organizan en lo que se conoce como *sistema de archivos*. Un sistema de archivos está formado por un dispositivo, con su propio árbol de directorios y archivos. Su árbol de directorio de Linux puede abarcar varios sistemas de archivos, cada uno en dispositivos de almacenamiento diferentes. En un disco duro con varias particiones, tiene un sistema de archivos para cada partición. Los archivos por sí solos se organizan en un árbol uniforme de directorios, que comienza desde el directorio raíz. Por ejemplo, si inserta un CD-ROM en su sistema, un nombre de ruta llevará directamente del directorio raíz de su sistema de archivos de la partición del disco duro a los archivos en el sistema de archivos del CD-ROM.

SUGERENCIA Con Linux se montan sistemas de archivos de diferentes tipos, incluidos los que se crean por otros sistemas operativos, como Windows, IBM OS, Unix y SGI. Dentro de Linux se da soporte a diversos sistemas de archivos, incluidos varios sistemas de registro por diario como ReiserFS y ext3.

Un sistema de archivos tiene sus archivos organizados en su árbol de directorios propio. Se le considera como un *subárbol* que debe adjuntarse al árbol de directorios principal. El árbol permanece separado de su árbol de directorios del sistema hasta que se conecte específicamente a éste. Por ejemplo, un disco flexible con archivos de Linux tiene su propio árbol de directorios. Necesita adjuntar su subárbol al árbol principal en la partición de su disco duro. Hasta que estén unidos, no podrá acceder a los archivos de su disco flexible.

Estándar de jerarquía de sistema de archivos (FHS)

Linux organiza sus archivos y directorios en un árbol general interconectado, que comienza en el directorio raíz y se extiende a los directorios de sistema y usuario. La organización y el diseño de sus directorios de sistema se determinan con el estándar de jerarquía de sistema de archivos (FHS, File System Hierarchy Standard). FHS proporciona un diseño estandarizado que todas las distribuciones de Linux deben seguir al configurar sus directorios de sistema. Por ejemplo, debe haber un directorio **/etc** para almacenar todos los archivos de configuración y un directorio **/dev** para archivos de dispositivos. Encontrará más acerca de FHS, incluida la documentación oficial, en pathname.com/fhs. Las distribuciones, los desarrolladores y los administradores de Linux siguen la FHS para proporcionar una organización constante al sistema de archivos de Linux.

Linux utiliza varios directorios con nombres específicos para tareas administrativas especializadas. Todos estos directorios están en el nivel más alto de su sistema de archivos principal de Linux, el directorio raíz del sistema de archivos, que se representa con una sola diagonal, **/**. Por ejemplo, el directorio **/dev** almacena archivos de dispositivos, y el directorio **/home** almacena los directorios de inicio de usuario y todos sus archivos de usuario. Tiene acceso a estos directorios y archivos sólo como el administrador del sistema (aunque los usuarios suelen tener acceso de sólo lectura). Necesita iniciar sesión como usuario **root**, colocándose en un directorio administrativo de usuario **root** especial llamado **/root**. De aquí, se accede a cualquier directorio en el sistema de archivos de Linux, ya sea administrativo o de usuario.

Directorio	Función
/	Comienza con la estructura de sistema de archivos (llamado raíz).
/boot	Almacena los archivos de imagen de kernel e información y archivos de arranque asociados.
/home	Contiene directorios de inicio de los usuarios.
/sbin	Almacena comandos en el nivel administrativo y cualquier comando que se utiliza por el usuario root.
/dev	Contiene interfaces de archivos generados dinámicamente para dispositivos como la terminal y la impresora (consulte “udev: archivos de dispositivos” en el capítulo 31).
/etc	Almacena archivos de configuración y cualquier otro archivo de sistema.
/etc/opt	Contiene archivos de configuración para aplicaciones en /opt.
/etc/X11	Almacena archivos de configuración para X Window System y sus aplicaciones.
/bin	Contiene comandos de usuario esenciales y programas de utilerías.
/lib	Almacena bibliotecas compartidas esenciales y módulos kernel.
/lib/modules	Contiene módulos kernel.
/media	Almacena directorios para montar sistemas de archivos extraíbles basados en medios, como CD-ROM, discos flexibles, lectores de tarjetas USB y cámaras digitales.
/mnt	Contiene directorios para sistemas de archivos adicionales como discos duros.
/opt	Almacena aplicaciones de software agregadas (por ejemplo, KDE en algunas distribuciones).
/proc	Directorio de procesos, directorio residente en memoria que contiene archivos utilizados para proporcionar información acerca del sistema.
/sys	Almacena el sistema de archivos sysfs para objetos de kernel, que despliega dispositivos y módulos a los que da soporte el kernel.
/tmp	Contiene archivos temporales.
/usr	Almacena los archivos y comandos que utiliza el sistema; este directorio se desglosa en varios subdirectorios.
/var	Contiene archivos que varían, como los de bandeja de entrada, Web y FTP.

TABLA 29-1 Directorios de sistema de archivos de Linux

El directorio raíz: /

Los subdirectorios que se almacenan en el directorio raíz, /, se muestran en la tabla 29-1, junto con otros subdirectorios útiles. Los directorios a los que suele acceder como administrador son /etc, que almacena archivos de configuración; /dev, que almacena archivos de dispositivos generados dinámicamente; y /var, que almacena archivos de datos de servidor para servidores DNS, Web, de correo y FTP, junto con registros de sistema y tareas programadas. Para administrar versiones diferentes del kernel, tal vez necesite acceder a los directorios /boot y /lib/modules y /usr/src/linux. El directorio /boot almacena archivos de imagen de kernel para cualquier kernel nuevo que instale, y el directorio /lib/modules almacena módulos para sus diferentes kernels.

Directorio	Descripción
<code>/bin</code>	Almacena programas relacionados con el sistema.
<code>/sbin</code>	Contiene programas de sistema para tareas especializadas.
<code>/lib</code>	Almacena bibliotecas de sistema.
<code>/etc</code>	Contiene archivos de configuración para servicios y aplicaciones de sistema y red.
<code>/home</code>	Almacena directorios de inicio de usuario y de datos de servidor, como archivos de sitios Web y sitios FTP.
<code>/media</code>	Donde se montan sistemas de archivo de medios extraíbles como CD-ROM, unidades USB y discos flexibles.
<code>/var</code>	Almacena directorios de sistema cuyos archivos cambian continuamente, como registros, archivos spool de impresora y archivos de bloqueo.
<code>/usr</code>	Contiene programas y archivos relacionados con el usuario. Incluye varios subdirectorios clave, como <code>/usr/bin</code> , <code>/usr/X11</code> y <code>/usr/share/doc</code> .
<code>/usr/bin</code>	Almacena programas para usuarios.
<code>/dev</code>	Contiene archivos de dispositivo.
<code>/sys</code>	Almacena el sistema de archivos sysfs con información de dispositivo para dispositivos que soportan kernel en su sistema.
<code>/usr/X11</code>	Contiene archivos de configuración de X Window System.
<code>/usr/share</code>	Almacena archivos compartidos.
<code>/usr/share/doc</code>	Contiene documentación para aplicaciones.
<code>/usr/share/hal</code>	Almacena configuración para dispositivos extraíbles HAL.
<code>/etc/udev</code>	Contiene configuración para archivos de dispositivos.
<code>/tmp</code>	Almacena archivos temporales de sistema.

TABLA 29-2 Directorio de sistema

Directorios del sistema

Su árbol de directorio de Linux contiene ciertos directorios cuyos archivos se utilizan para diferentes funciones de sistema. Para administración básica del sistema, debe estar familiarizado con los directorios del programa de sistema, donde se almacenan las aplicaciones; el directorio de configuración de sistema (`/etc`), donde se colocan casi todos los archivos de configuración y el directorio de registro de sistema (`/var/log`), que almacena los registros del sistema, registrando la actividad de su sistema. Ambos se cubren con detalle en este capítulo. En la tabla 29-2 se muestra una lista de directorios del sistema.

Directorios de programa

Los directorios con `bin` en el nombre se utilizan para almacenar programas. El directorio `/bin` almacena programas de usuario básicos, como inicio de sesión, shell (BASH, TCSH y zsh) y comandos de archivos (`cp`, `mv`, `rm`, `ln`, etc.). El directorio `/sbin` almacena programas de sistema especializados para tareas tales como administración de sistema de archivos (`fsck`, `fdisk`, `mkfs`)

y operaciones como apagado e inicio (**init**). El directorio **/usr/bin** almacena archivos de programa diseñados para tareas de usuario. El directorio **/usr/sbin** almacena operaciones de sistema relacionadas con usuarios, como **useradd** para agregar nuevos usuarios. El directorio **/lib** almacena todas las bibliotecas que usa su sistema, incluida la biblioteca principal de Linux, **libc** y los subdirectorios como **modules**, que almacena todos los módulos de kernel actuales.

Configuración de directorios y archivos

Cuando se configuran elementos diferentes en su sistema, como cuentas de usuario, aplicaciones, servidores o conexiones de red, se hace uso de archivos de configuración que se almacenan en ciertos directorios del sistema. Los archivos de configuración se colocan en el directorio **/etc**.

El directorio **/usr**

El directorio **/usr** contiene multitud de subdirectorios importantes utilizados para dar soporte a usuarios, y proporcionar aplicaciones, bibliotecas y documentación. El directorio **/usr/bin** contiene varias aplicaciones y utilerías accesibles para el usuario; **/usr/sbin** almacena utilerías administrativas, accesibles también para el usuario. El directorio **/usr/share** almacena datos de arquitectura independiente que incluye un extenso número de subdirectorios, incluidos los de la documentación como archivos **man**, **info** y **doc**. En la tabla 29-3 se muestran los subdirectorios para el directorio **/usr**.

El directorio **/media**

El directorio **/media** se utiliza para puntos de montaje de medios extraíbles como CD-ROM, DVD, discos flexibles o unidades ZIP, además de otros sistemas de archivos basados en medios como lectores de tarjeta USB, cámaras y reproductores MP3. Existen sistemas de archivos que puede cambiar con frecuencia, a diferencia de particiones en discos fijos. Casi todos los sistemas Linux utilizan la capa de abstracción de hardware (HAL, Hardware Abstraction Layer) para administrar de forma dinámica la creación, el montaje y la asignación de dispositivo. Como lo indica HAL, esta herramienta creará subdirectorios de discos flexibles, CD-ROM, tarjeta de almacenamiento, cámara y reproductor MP3 en **/media**, de acuerdo con lo que se necesite. El subdirectorio predeterminado para montaje es **/media/disk**. Las unidades adicionales tienen un número adjunto a su nombre.

Directorio	Descripción
/usr/bin	Contiene la mayor parte de los comandos de usuario y programas de utilería.
/usr/sbin	Almacena aplicaciones administrativas.
/usr/lib	Contiene bibliotecas para aplicaciones, lenguajes de programación, escritorios, etc.
/usr/games	Almacena nombres y programas educativos.
/usr/include	Contiene archivos de encabezado de lenguajes de programación C (.h).
/usr/doc	Almacena documentación de Linux.
/usr/local	Contiene software instalado de manera local.
/usr/share	Almacena datos de arquitectura independiente como documentación.
/usr/src	Contiene código fuente, incluido el del kernel.
/usr/X11R6	Almacena aplicaciones y bibliotecas basadas en X Windows System.

TABLA 29-3 Subdirectorios de **/usr**

El directorio **/mnt**

El directorio **/mnt** suele utilizarse para puntos de montaje de otros sistemas de archivos montados como las particiones de Windows. Puede crear directorios para cualquier partición que quiera montar, como **/mnt/windows** para una partición de Windows.

El directorio **/home**

El directorio **/home** contiene directorios de inicio de usuario. Cuando se define una cuenta de usuario, también se configura un directorio de inicio aquí para esa cuenta, por lo general con el mismo nombre que el usuario. Como administrador de sistema, puede acceder a cualquier directorio de inicio del usuario, así que tiene control sobre los archivos de los usuarios.

El directorio **/var**

El directorio **/var** almacena subdirectorios para tareas cuyos archivos cambian con frecuencia, como archivos de bloqueo, de registro, de servidor Web o spool de impresora. Por ejemplo, el directorio **/var** almacena directorios de datos de servidor, como **/var/www** para los archivos de sitio Web del servidor Web Apache o **/var/ftp** para sus archivos de sitio FTP, además de **/var/named** para el servidor DNS. El directorio **/tmp** es simplemente un directorio para almacenar archivos temporales que probablemente los programas necesitarán para realizar una tarea particular.

Directorio	Descripción
/var/account	Registros de procesos de contabilidad.
/var/cache	Almacena datos de caché de aplicación para páginas Man, datos de proxy Web, fuentes o datos de aplicación específica.
/var/crash	Contiene volcados tras fallas del sistema.
/var/games	Contiene datos de diversos juegos.
/var/lib	Almacena información de estado para aplicaciones particulares.
/var/local	Contiene datos que cambian de programas instalados en /usr/local .
/var/lock	Almacena archivos de bloqueo que indican cuando un programa o archivo particular está en uso.
/var/log	Contiene archivos de registro como /var/log/messages que contienen todos los mensajes de kernel y programas de sistema.
/var/mail	Almacena archivos de bandeja de entrada de usuario.
/var/opt	Contiene datos variables para aplicaciones instaladas en /opt .
/var/run	Almacena información acerca de procesos en ejecución del sistema.
/var/spool	Contiene datos de spool de aplicaciones como las de correo, noticias y colas de impresión, además de trabajos cron y at .
/var/tmp	Almacena archivos temporales que deben preservarse entre reinicios de sistema.
/var/yp	Contiene archivos de datos del servicio de información de red (NIS, Network Information Service).
/var/www	Almacena archivos de sitio Web de servidores Web.
/var/ftp	Contiene archivos FTP de servidores FTP
/var/named	Almacena archivos de configuración de dominio de servidor DNS.

TABLA 29-4 Subdirectorios de **/var**

Los directorios **/var** están diseñados para almacenar datos que cambian con la operación normal del sistema Linux. Por ejemplo, aquí se almacenan los archivos spool para documentos que está imprimiendo. Un archivo spool se crea como un archivo de impresión temporal y se elimina después de la impresión. Otros archivos, como los de registro del sistema, se cambian constantemente. En la tabla 29-4 se muestra una lista de subdirectorios del directorio **/var**.

El sistema de archivos **/proc**

/proc es un sistema de archivos especial que se genera en la memoria del sistema. No existe en ningún disco. Contiene archivos que proporcionan información importante acerca del estado de su sistema. Por ejemplo, **/proc/cpuinfo** almacena información acerca de la CPU de su computadora, **/proc/devices** incluye los dispositivos configurados para ejecutarse con su kernel, **/proc/filesystems** despliega los sistemas de archivo y los archivos **/proc** son realmente interfaces al kernel, que obtienen información del kernel acerca de su sistema. En la tabla 29-5 se muestran los subdirectorios y archivos de **/proc**.

Como cualquier sistema de archivos, es necesario montar **/proc**. El archivo **/etc/fstab** tendrá una entrada especial para **/proc** con un tipo sistema de archivos de proc y sin ningún dispositivo especificado.

```
none    /proc    proc    defaults    0    0
```

SUGERENCIA Se utiliza *sysctl*, la herramienta Kernel Tuning, para establecer valores de archivo proc que tiene permitido cambiar, como el número máximo de archivos o activar el reenvío IP.

Archivo	Descripción
/proc/num	Existe un directorio para cada proceso etiquetado por su número. /proc/1 es el directorio del proceso 1.
/proc/cpuinfo	Contiene información acerca de la CPU, como tipo, marca, modelo y rendimiento.
/proc/devices	Incluye los controladores de dispositivos configurados para el kernel de ejecución actual.
/proc/dma	Despliega los canales DMA en uso.
/proc/filesystems	Muestra los sistemas de archivo configurado en el kernel.
/proc/interrupts	Despliega las interrupciones en uso.
/proc/ioports	Muestra los puertos de entrada y salida en uso.
/proc/kcore	Almacena una imagen de la memoria física del sistema.
/proc/kmsg	Contiene mensajes generados por el kernel.
/proc/loadavg	Muestra el porcentaje de carga de sistema.
/proc/meminfo	Despliega la memoria en uso.
/proc/modules	Muestra los módulos kernel cargados actualmente.
/proc/net	Despliega la información de estado acerca de los protocolos de red.
/proc/stat	Contiene estadísticas de sistema operativo, como situaciones de falla de página.
/proc/uptime	Despliega el tiempo que ha estado activo el sistema.
/proc/version	Muestra la versión de kernel.

TABLA 29-5 Subdirectorios y archivos de **/proc**

El sistema de archivos sysfs: /sys

El sistema de archivos **sysfs** es un sistema de archivos virtual que proporciona un mapa jerárquico de sus dispositivos soportados por el kernel como dispositivos PCI, buses y dispositivos de bloqueo, además de módulos de soporte de kernel. El subdirectorio **classes** mostrará todos sus dispositivos soportados por categoría, como los de red y sonido. Con **sysfs**, su sistema determina de manera sencilla el archivo de dispositivo al que está asociado un dispositivo determinado. Esto es de mucha ayuda para administrar dispositivos extraíbles, además de configurar y administrar dispositivos como lo hacen HAL y udev. Este último utiliza el archivo **sysfs** para generar de forma dinámica archivos de dispositivos necesarios en el directorio **/dev**; HAL lo utiliza para administrar los archivos y el soporte de dispositivos extraíbles, a medida que se necesitan (técnicamente, HAL proporciona información acerca de dispositivos, aunque utiliza herramientas para cambiar las configuraciones de manera dinámica, de acuerdo con lo necesario). El tipo de sistema de archivos **/sys** es **sysfs**. El subdirectorio **/sys** organiza sus dispositivos en diferentes categorías. **systool** utiliza el sistema de archivos para desplegar una lista de sus dispositivos instalados. En el siguiente ejemplo se muestran todos sus dispositivos de sistema.

```
Systool
```

Como **/proc**, el directorio **/sys** reside sólo en la memoria, pero todavía necesita montarlo en el archivo **/etc/fstab**.

```
none    /sys    sysfs   defaults 0      0
```

Archivos de dispositivo: /dev, udev y HAL

Para montar un sistema de archivos, tiene que especificar su nombre de dispositivo. Las interfaces a los dispositivos que pueden conectarse a su sistema son proporcionadas por archivos especiales conocidos como *archivos de dispositivo*. Los nombres de estos archivos son los de los dispositivos. Los archivos de dispositivo se ubican en los directorios **/dev** y suelen tener nombres abreviados que terminan con el número de dispositivo. Por ejemplo, tal vez **fd0** haga referencia a la primera unidad de disco flexible conectada a su sistema. El prefijo **sd** hace referencia a los discos duros SCSI, así que **sda2** haría referencia a la segunda partición del primer disco duro SCSI. En casi todos los casos, se utiliza el comando **man** con un prefijo para obtener más información detallada acerca de este tipo de dispositivo. Por ejemplo, **man sd** despliega las páginas Man para dispositivos SCSI. Una lista completa de todos los nombres de dispositivos se encuentra en el archivo **devices** que se ubica en el directorio **linux/doc/device-list** en el sitio Web **kernel.org** y en el archivo **devices.txt** en el directorio **/etc/usr/linux-2.4/Documentación** de su sistema. En la tabla 29-6 se muestran varios de los nombres de dispositivo de uso común.

udev y HAL

Los archivos de dispositivo ya no se manejan de forma estática; ahora se generan de forma dinámica, de acuerdo con las necesidades. Antes, se creaba un archivo para cada dispositivo posible, lo que daba lugar a un gran número de archivos de dispositivos en el directorio **/etc/dev**. Ahora su sistema sólo detectará los dispositivos que utiliza y sólo creará archivos de dispositivo para éstos, lo que produce una lista mucho más pequeña de archivos de dispositivo. La herramienta que se utiliza para detectar y generar archivos de dispositivos es udev (abreviatura de user device, dispositivos de usuario). Cada vez que su sistema se inicia, udev detecta automáticamente sus dispositivos y genera archivos para éstos en el directorio **/etc/dev**. Esto significa que el directorio **/etc/dev** y sus archivos se vuelven a crear cada vez que arranca. Se trata de un directorio dinámico, ya no estático. Para administrar estos archivos de dispositivo, necesita utilizar archivos de configuración ubicados en el

Nombre de dispositivo	Descripción
hd	Discos duros IDE; de 1 a 4 son las particiones principales; de 5 en adelante son particiones lógicas.
sd	Discos duros SCSI.
scd	Unidades de CD-ROM SCSI.
fd	Discos flexibles.
st	Unidades de cinta SCSI.
nst	Unidades de cinta SCSI, sin regresar.
ht	Unidades de cinta IDE.
tty	Terminales.
lp	Puertos de impresora.
pty	Seudoterminales (se utilizan para inicios de sesión remotos).
js	Joysticks análogos.
midi	Puertos midi.
ttyS	Puertos seriales.
md	Dispositivos RAID.
rd/cndn	El directorio que almacena dispositivos RAID es rd ; cn es el controlador RAID y dn es el disco RAID para tal controlador.
cdrom	Vincula con su archivo de dispositivo de CD-ROM, se configura en /etc/udev/rules.d
cdrecorder	Vincula con su archivo de dispositivo CD-R o CD-RW; se configura en /etc/udev/rules.d
modem	Vincula con su archivo de dispositivo de módem, se configura en /etc/udev/rules.d
floppy	Vincula con su archivo de dispositivo de disco flexible; se configura en /etc/udev/rules.d
tape	Vincula a su archivo de dispositivo de cinta: se configura en /etc/udev/rules.d
scanner	Vincula a su archivo de dispositivo de escáner; se configura en /etc/udev/rules.d

TABLA 29-6 Prefijos de nombre de dispositivo

directorio **/etc/udev**. Esto significa que udev también puede administrar de forma dinámica todos los dispositivos extraíbles; udev generará y configurará archivos de dispositivos para dispositivos extraíbles cuando se conectan, y después elimina estos archivos cuando los dispositivos se extraen. En este sentido, todos los dispositivos ahora se consideran de inserción activa, y los dispositivos fijos sólo son de inserción activa que nunca se extraen.

Debido a que ahora **/etc/dev** es dinámico, cualquier cambio que haga manualmente al directorio **/etc/dev** se perderá cuando reinicie. Esto incluye la creación de cualquier vínculo simbólico como **/dev/cdrom** que utilizan muchas aplicaciones de software. En cambio, estos vínculos simbólicos tienen que configurarse al utilizar reglas udev que se muestran en archivos de configuración ubicados en el directorio **/etc/udev/rules.d**. Ya hay reglas predeterminadas para vínculos simbólicos, pero puede crear reglas propias. Consulte el capítulo 32 para conocer más detalles.

Además de udev, la información acerca de dispositivos extraíbles como CD-ROM y discos flexibles, junto con cámaras e impresoras USB, utilizados por aplicaciones como el escritorio para interactuar de forma dinámica con éstos, se administra con una utilidad llamada capa abstracta de

hardware (HAL, Hardware Abstract Layer). HAL permite que se reconozcan dispositivos extraíbles como impresoras USB, sin importar las conexiones particulares que pueda utilizar. Por ejemplo, puede conectar una impresora USB en un puerto USB en un momento y después cambiar a otro. El archivo **fstab** se edita al utilizar la herramienta **fstab-sync**, que se invoca mediante reglas HAL en archivos de configuración en el directorio **/usr/share/hal/fdi**. Consulte el capítulo 32 para conocer más detalles.

HAL tiene un impacto clave en el archivo **/etc/fstab** que se utiliza para administrar sistemas de archivo. Ya no se mantienen entradas en el archivo **/etc/fstab** para dispositivos extraíbles como sus CD-ROM. HAL administra directamente estos dispositivos al utilizar su conjunto de llamadas de almacenamiento, como **hal-system-storage-mount** para montar un dispositivo o **hal-system-storage-eject** para eliminar uno. En efecto, ahora tiene que utilizar los archivos de información de dispositivo de HAL para administrar sus sistemas de archivo extraíbles. Si quiere evitar HAL y configurar manualmente un dispositivo CD-ROM, sólo coloque una entrada para éste en el archivo **/etc/fstab**.

Dispositivos de discos flexible y duro

El nombre de dispositivo para su unidad de disco flexible es **fd0**; se ubica en el directorio **/dev**. **/dev/fd0** hace referencia a su unidad de disco flexible. Observe el número **0** después de **fd**. Si tiene más de una de estas unidades, las adicionales se representan por **fd1**, **fd2**, etcétera.

Los discos duros IDE utilizan el prefijo **hd**, mientras que los discos duros SCSI utilizan el prefijo **sd**. Los dispositivos RAID, por otra parte, utilizan el prefijo **md**. El prefijo para un disco duro es seguido por una letra que etiqueta a éste y un número para la partición. Por ejemplo, **hda2** alude a la segunda partición en el primer disco duro IDE, mientras que se hace referencia al primer disco duro con la letra **a**, como en **hda**. El dispositivo **sdb3** hace referencia a la tercera partición en el segundo disco duro SCSI (**sdb**). Sin embargo, los dispositivos RAID se numeran desde 0, como las unidades de disco flexible. Los dispositivos **md0** hacen referencia al primer dispositivo RAID, y **md1** al segundo. En un dispositivo de disco duro IDE, Linux da soporte a cuatro particiones de disco duro IDE principales, que se numeran de 1 a 4. Se permite cualquier número de particiones lógicas. Para encontrar el nombre de dispositivo, se utiliza **df** para desplegar sus particiones de disco duro o examinar el archivo **/etc/fstab**.

NOTA GNOME ahora administra todos los medios extraíbles directamente con HAL, en vez de utilizar entradas **fstab**.

Dispositivos de CD-ROM

El nombre del dispositivo para su unidad CD-ROM varía dependiendo del tipo de CD-ROM que tiene. El nombre de dispositivo para un CD-ROM IDE tiene el mismo prefijo que una partición de disco duro IDE, **hd**, y se identifica con una letra después que lo distingue de otros dispositivos IDE. Por ejemplo, un CD-ROM IDE conectado a un puerto IDE secundario puede tener el nombre **hdc**. Un CD-ROM IDE conectado como esclavo al puerto secundario puede tener el nombre **hdd**. El nombre real se determina cuando se instala el CD-ROM, como pasa cuando instala su sistema Linux. Las unidades CD-ROM SCSI utilizan una nomenclatura diferente para sus nombres de dispositivo. Comienzan con **scd** para una unidad SCSI y siguen por un número distintivo. Por ejemplo, el nombre de un CD-ROM SCSI podría ser **scd0** o **scd1**. El nombre de su CD-ROM fue determinado cuando instaló su sistema.

Como ya se observó, ahora los dispositivos CD-ROM se configuran con HAL. HAL hace esto en un archivo de información de dispositivo en su directorio de política de configuración. Para configurar un dispositivo de CD-ROM, como para agregar capacidades de montaje de usuario, necesita configurar su entrada en el archivo de configuración **storage-methods.fdi** (consulte el

capítulo 31 para conocer más detalles). El Administrador de volumen de GNOME utiliza HAL y udev para acceder a medios extraíbles directamente y a Samba para proporcionar soporte a redes de Windows. Los medios se montan con **gnome-mount**, una envoltura para acceder a HAL, y udev, que realiza el montaje (*/etc/fstab* ya no se utiliza).

Montaje de sistemas de archivos

A la acción de adjuntar un sistema de archivos en un dispositivo de almacenamiento de su árbol de directorios principal se le denomina *montar* el dispositivo. Su sistema de archivos se monta en un directorio vacío del árbol de directorio principal. Luego se cambia a ese directorio y se accede a tales archivos. Si el directorio aún no existe, tiene que crearlo. El directorio de la estructura de archivos al que se adjunta el nuevo sistema de archivos se conoce como *punto de montaje*. Por ejemplo, para acceder a los archivos de su CD-ROM, primero tiene que montar el CD-ROM.

Por lo general, sólo el usuario root puede montar un sistema de archivos. Se trata de una tarea de administración del sistema y, en esencia, no debe realizarla un usuario regular. Sin embargo, como usuario root, puede hacer que un dispositivo particular, como un CD-ROM, sea montable por el usuario. De esta forma, cualquier usuario puede montar un CD-ROM. Se puede hacer lo mismo en el caso de una unidad de disco flexible.

SUGERENCIA En GNOME, se utiliza la herramienta Administrador del disco, en la ventana Herramientas del sistema, para montar y desmontar sistemas de archivos, incluidos discos flexibles y CD-ROM. En KDE, se utiliza la utilidad KDiskFree (en el menú Más aplicaciones), que también muestra sus sistemas de archivos montables, además de su uso de disco.

Incluso deben montarse explícitamente los sistemas de archivos en su partición de disco duro. Sin embargo, cuando instala su sistema de Linux y se crea la partición de Linux en su disco duro, su sistema se configura automáticamente para montar su sistema de archivos principal siempre que inicia. Cuando su sistema se apaga, se desmontan automáticamente. Tiene la opción de desmontar cualquier sistema de archivos, eliminarlo del árbol de directorios y posiblemente remplazarlo con otro, como cuando reemplaza un CD-ROM.

Una vez que se ha montado en realidad el sistema de archivos, se hace una entrada en el sistema operativo, en el archivo */etc/mstab*. Aquí encontrará una lista de todos los sistemas de archivos montados actualmente.

Información de sistema de archivo

Los sistemas de archivos de cada dispositivo de almacenamiento se formatean para ocupar una cantidad máxima específica de espacio. Por ejemplo, tal vez haya formateado su partición de disco duro para tomar 3 GB. Los archivos instalados o creados en tal sistema de archivos ocupan parte de este espacio, mientras que el resto está disponible para nuevos archivos y directorios. Para conocer más acerca de cuánto espacio libre tiene en un sistema de archivos, se utiliza el comando **df** o, en el escritorio, el Monitor del sistema de GNOME o la utilidad KDiskFree de KDE. En el caso del Monitor del sistema, haga clic en la ficha Recursos para desplegar una lista de espacio libre en sus sistemas de archivo. KDiskFree despliega una lista de dispositivos, que muestra cuánto espacio está libre en cada partición, y el porcentaje utilizado.

df

El comando **df** reporta el uso de espacio en disco del sistema de archivos. Despliega todos sus sistemas de archivos por sus nombres, cuánto espacio toman y el porcentaje de espacio utilizado en disco y el lugar en que se montan. Con la opción **-h**, se despliega información en un formato más

legible, como medir el espacio en disco en megabytes en lugar de hacerlo en bloques de memoria. El comando **df** también es seguro para obtener una lista de todas sus particiones, en lugar de utilizar **fdisk** (porque con **fdisk** se eliminan particiones). Sin embargo, **df** sólo muestra particiones montadas, mientras que **fdisk** muestra todas las particiones.

```
$ df -h
Filesystem Size Used Avail Use% Mounted on
/dev/hda3 9.7G 2.8G 6.4G 31% /
/dev/hda2 99M 6.3M 88M 7% /boot
/dev/hda2 22G 36M 21G 1% /home
/dev/hdc 525M 525M 0 100% /media/disk
```

También puede utilizar **df** para que le indique qué sistema de archivos pertenece a un directorio dado. Inserte **df** con el nombre de directorio o **df .** para el directorio actual.

```
$ df .
Filesystem 1024-blocks Used Available Capacity Mounted on
/dev/hda3 297635 169499 112764 60% /
```

e2fsck y fsck

Para revisar la constancia del sistema de archivos y para repararlo, si está dañado, se utilizan las herramientas de revisión del sistema de archivos. **fsck** revisa y repara un sistema de Linux. **e2fsck** está diseñado para exportar archivos de sistema **ext2** y **ext3**, mientras que el **fsck** más genérico también trabaja en cualquier sistema de archivos. Los sistemas de archivos **ext2** y **ext3** son los que suelen utilizarse con particiones de discos duro y flexible de Linux. Por lo general, los sistemas de archivos de Linux son **ext3**, que utilizan **e2fsck** para revisiones. **fsck** y **e2fsck** toman como argumento el nombre de dispositivo de la partición de disco duro que utiliza el sistema de archivos.

```
fsck nombre-dispositivo
```

Antes de que revise un sistema de archivos, asegúrese de que el sistema de archivos no está montado. **e2fsck** no debe utilizarse en un sistema de archivos montado. Para utilizar **e2fsck**, inserte **e2fsck** y el nombre de dispositivo que hace referencia a ese sistema de archivos. La opción **-p** repara de forma automática un sistema de archivos sin tener que pedir primero la aprobación del usuario para cada tarea de reparación. En los siguientes ejemplos se revisa el disco en las unidades de discos flexible y duro principal:

```
# e2fsck /dev/fd0
# e2fsck /dev/hda1
```

Con **fsck**, la opción **-t** le permite especificar el tipo de sistema de archivos que se revisa, y la opción **-a** repara automáticamente los sistemas, mientras que la opción **-r** primero pide la confirmación. La opción **-A** revisa todos los sistemas en el archivo **/etc/fstab**.

Registro por diario

Los sistemas de archivos **ext3** y ReiserFS introdujeron las capacidades de registro por diario en los sistemas Linux. Registro por diario proporciona una recuperación rápida y efectiva en caso de bloqueos de disco, en lugar de utilizar **e2fsck** o **fsck**. Con registro por diario se mantiene un registro de todas las acciones del sistema de archivos, que se coloca en un archivo de diario. En caso de bloqueo, Linux sólo necesita leer el archivo de diario y reproducirlo para restaurar el sistema a

su estado previo (estable). Los archivos que estaban en proceso de escritura en el disco se restauran a su estado original. Registro por diario también evita revisiones **fsck** largas en reinicios que ocurren cuando se suspende de pronto el suministro de energía eléctrica, o se congela y tiene que reiniciarse físicamente. En lugar de usar **fsck** para verificar manualmente cada archivo y directorio, su sistema sólo lee sus archivos de diario para restaurar el sistema de archivos.

El mantenimiento de un diario conlleva más trabajo para el sistema de archivos que un método sin diario. Aunque todos los sistemas de registro por diario mantienen una estructura de directorio del sistema de archivos (a lo que se le conoce como *metadatos*), ofrecen varios niveles de recuperación de datos. El mantenimiento de información para recuperación de datos de archivo es muy tardado, porque alienta el tiempo de respuesta del sistema de archivos. Al mismo tiempo, los sistemas de registro por diario hacen un uso más eficiente del sistema de archivos, proporcionando un tiempo de respuesta más rápido que el **ext2** sin diario.

Existen otros tipos de sistemas de archivos de registro por diario que se utilizan en Linux. Éstos incluyen ReiserFS, JFS y XFS. ReiserFS proporciona una estructura de sistema de archivos totalmente rediseñada, basada en registro por diario (**namesys.com**). Casi todas las distribuciones también proporcionan soporte para sistemas de archivos ReiserFS. JFS es la versión IBM del sistema de archivos de registro por diario, diseñada para usarse en servidores que proporcionan alto rendimiento, como los de empresas de negocios electrónicos (<http://jfs.sourceforge.net>). Se distribuye de manera gratuita bajo la licencia pública GNU. XFS es otro sistema registro por diario de alto rendimiento desarrollado por Silicon Graphics (oss.sgi.com/projects/xfs). XFS es compatible con sistemas de archivos RAID y NFS.

Registro por diario de ext3

El soporte a registro por diario se da en el kernel de Linux con **ext3**. El sistema de archivos **ext3** también es totalmente compatible con la versión anterior **ext2** que remplazó. Para crear un sistema de archivos **ext3**, se utiliza el comando **mkfs.ext3**. Incluso puede actualizar automáticamente sistemas de archivos **ext2** a versiones **ext3**, sin pérdida de datos o cambio de particiones. Esta actualización sólo agrega un archivo de diario a un sistema de archivos **ext2** y permite registro por diario en éste, al utilizar el comando **tune2fs**. Asegúrese de cambiar el tipo de archivo **ext2** a **ext3** en cualquier entrada **/etc/fstab** correspondiente. En el siguiente ejemplo se convierte el sistema de archivos **ext2** en **/dev/hda3** al sistema de archivos **ext3**, al agregar un archivo de diario (-j).

```
tune2fs -j /dev/hda3
```

El sistema de archivos **ext3** mantiene soporte completo a recuperación de metadatos (recuperación del árbol de directorios), pero ofrece varios niveles de recuperación de datos de archivo. En efecto, está compensando menos datos de archivo con más velocidad. El sistema de archivos **ext3** da soporte a tres opciones: **writeback**, **ordered** y **journal**. La opción predeterminada es **writeback**. La opción **writeback** sólo proporciona recuperación de metadatos, sin recuperación de datos de archivo. La opción **ordered** da soporte a una recuperación limitada de datos de archivos, y la opción **journal** proporciona recuperación completa de datos de archivos. Se recuperará cualquier archivo en proceso de ser cambiado durante una falla del sistema. Para especificar una opción de **ext3**, utilice la opción **data** en el comando **mount**.

```
data=ordered
```

ReiserFS

Aunque a menudo se utiliza el registro por diario para recuperarse de fallas de disco, un sistema de archivos basado en diario hace mucho más. Los sistemas de archivos **ext3**, JFS y XFS sólo

proporcionan operaciones de registro que se utilizan para la recuperación, mientras que ReiserFS utiliza técnicas de registro por diario para volver a trabajar completamente operaciones de sistema de archivos. En ReiserFS, el registro por diario se utiliza para leer y escribir datos; abandona la estructura de bloque utilizada tradicionalmente en sistemas Unix y Linux. Esto da la capacidad de acceder a un gran número de archivos pequeños de manera muy rápida, además de utilizar sólo la cantidad de espacio en disco necesaria. Sin embargo, la eficiencia no es mucho mejor con archivos grandes.

Montaje automático de sistemas de archivos: /etc/fstab

Los sistemas de archivos se montan al utilizar el comando **mount**, que se describe en la siguiente sección. Aunque puede montar directamente un sistema de archivos sólo con el comando **mount**, se simplifica el proceso si se coloca la información de montaje en el archivo de configuración **/etc/fstab**. Al utilizar las entradas de este archivo, puede hacer que ciertos sistemas de archivos se monten automáticamente cada vez que su sistema se inicie. Para otros, puede especificar información de configuración, como puntos de montaje y permisos de acceso, que se utilizan de manera automática siempre que monta un sistema de archivos. No es necesario que inserte esta información como argumentos en un comando **mount**, como sucedería de otra forma. Esta característica es lo que permite montar utilerías en GNOME o KDE para permitirle montar un sistema de archivos con sólo hacer clic en un icono de ventana. Toda la información de montaje ya está en el archivo **/etc/fstab**. Por ejemplo, cuando agrega una nueva partición de disco duro a su sistema Linux, lo más probable es que quiera que se monte automáticamente en el inicio y después se desmonte cuando se apague. De otra forma, tendría que montar y desmontar la partición explícitamente cada vez que inicie o apague el sistema.

HAL y fstab

Para que Linux monte automáticamente el sistema de archivos en su nueva partición de disco duro, sólo necesita agregar su nombre al archivo **fstab**, excepto en el caso de dispositivos extraíbles como CD-ROM e impresoras USB. Los dispositivos extraíbles se administran con HAL, al utilizar archivos de directiva de almacenamiento ubicados en **/usr/share/hal/fdi** y **/etc/hal/fdi**. El servicio **haldaemon** detecta automáticamente los dispositivos, y HAL los administra directamente al utilizar su conjunto de llamadas de almacenamiento, como **hal-system-storage-mount**, para montar un dispositivo o **hal-system-storage-eject** para eliminar uno. En efecto ahora tiene que utilizar los archivos de información de dispositivo de HAL para administrar sus sistemas de archivos extraíbles. Si quiere configurar opciones diferentes para el dispositivo, debe crear su propio archivo **storage-methods.fdi** en el directorio **30user**. La configuración se implementa al utilizar el lenguaje XML. Revise el archivo de almacenamiento predeterminado en **10osvenders/20-storage-methods.fdi**, además de ejemplos en el directorio **/usr/share/doc/hal/versión/conf**. Consulte el capítulo 31 para conocer ejemplos sobre la manera de utilizar HAL para configurar opciones de dispositivos.

Campos de fstab

Una entrada en el archivo **fstab** contiene varios campos, cada uno separado del siguiente por un espacio o tabulador. Éstos se describen como los campos dispositivo, punto de montaje, tipo de sistema de archivo, opciones, volcado y **fsck**, ordenados en la secuencia que se muestra aquí:

```
<dispositivo> <puntodemontaje> <tipodesistemadearchivo> <opciones> <volcado> <fsck>
```

El primer campo es el nombre del sistema que se monta. Esta entrada es un nombre de dispositivo o una etiqueta de sistema de archivo **ext2** o **ext3**. Un nombre de dispositivo suele

Tipo	Descripción
auto	Intenta detectar automáticamente el tipo de sistema de archivos.
minux	Los sistemas de archivos Minux (los nombres de archivo se limitan a 30 caracteres).
ext	Versión anterior de sistema de archivos de Linux; ya no se usa.
ext3	Sistema de archivos estándar de Linux que da soporte a nombres de archivos largos y tamaños de archivos más grandes; incluye registro por diario.
ext2	Sistema anterior de archivos estándar de Linux que da soporte a nombres de archivos largos y tamaños de archivos más grandes; no tiene registro por diario.
ntfs-3g	Sistemas de archivos Windows NT, XP Vista y 2000 con capacidades de escritura, el proyecto NTFS-3g.
msdos	Sistema de archivos para particiones MS-DOS (16-bits).
vfat	Sistema de archivos para particiones Windows 95, 98 y Millennium (32-bits).
reiserfs	Sistema de archivos de registro por diario de ReiserFS.
xfs	Sistema de archivos de Silicon Graphics (SGI).
ntfs	Sistema de archivos de Windows NT, XP y 2000, sólo lectura.
smbfs	Sistemas de archivos remotos de Samba, como NFS.
hpfs	Sistema de archivos para particiones de alto rendimiento OS/2.
nfs	Sistema de archivos NFS para montar particiones de sistemas remotos.
nfs4	Sistema de archivos NFSv4 para montar particiones de sistemas remotos.
umsdos	Sistema de archivos UMS-DOS.
swap	Partición de intercambio o archivo de intercambio de Linux.
sysv	Sistemas de archivos System V de Unix.
iso9660	Sistema de archivos para montar CD-ROM.
proc	Lo utiliza el sistema operativo para procesos (sistema de archivos que da soporte a kernel).
sysfs	Lo utiliza el sistema operativo para dispositivos (sistema de archivos que da soporte a kernel).
usbfs	Lo utiliza el sistema operativo para dispositivos USB (sistema de archivos que da soporte a kernel).
devpts	Seudoterminales de Unix 98 (ttys, sistema de archivos que interactúa con el kernel).
shmfs y tmpfs	Memoria virtual de Linux, acceso de mantenimiento de memoria compartida POSIX (sistema de archivos que interactúa con el kernel).
adfs	Sistemas de archivos DOS de Apple.
affs	Sistemas de archivos rápidos de Amiga.
ramfs	Sistemas de archivos basados en RAM.
udf	Formato de disco universal que se utiliza en CD/DVD-ROM.
ufs	Sistema de archivos de Unix; se encuentra en sistemas Unix (formato antiguo).

TABLA 29-7 Tipos de sistema de archivos

comenzar con **/dev**, como **/dev/hda3** para la tercera partición del disco duro. Una etiqueta se especifica al asignar su nombre a la etiqueta **LABEL**, como en **LABEL=/** para una partición raíz **ext2**. El siguiente campo es el directorio de su estructura de archivos adonde quiere que se conecte el sistema de archivos en este dispositivo. El tercer campo es el tipo de sistema de archivos que se monta. En la tabla 29-7 se proporciona una lista de todos los tipos que se montan. El tipo de partición de disco duro de Linux estándar es **ext3**. En el siguiente ejemplo se muestra una entrada para la partición principal del disco duro de Linux. Esta entrada se monta en el directorio root, **/**, y tiene el tipo de archivos de **ext3**:

```
/dev/hda3    /      ext3    defaults    0    1
```

En el siguiente ejemplo se muestra una entrada **LABEL** para la partición de disco duro, donde el nombre de etiqueta es **/**:

```
LABEL=/      /      ext3    defaults    0    1
```

Montajes automáticos

El tipo de sistema de archivos para un disco flexible puede diferir dependiendo del disco que esté tratando de montar. Por ejemplo, tal vez quiera leer un disco flexible con formato para Windows en un momento y uno con formato para Linux en otro. Por esto, el tipo de sistema de archivos especificado para el dispositivo de disco flexible es **auto**. Con esta opción, se detecta automáticamente el tipo de sistema de archivos formateado en el disco flexible y se utiliza el tipo de sistema de archivos apropiado.

```
/dev/fd0    /media/floppy    auto    defaults,noauto    0    0
```

Opciones de montaje

El campo que se encuentra después del tipo de sistema de archivos presenta diferentes opciones para montar el sistema de archivos. El conjunto predeterminado de opciones se especifica con **defaults**, y las opciones específicas se incluyen una tras otra, separadas por comas (sin espacios). La opción **defaults** especifica que un dispositivo es de lectura/escritura (**rw**), asíncrono (**async**), un dispositivo de bloqueo (**dev**), que no permite el montaje por parte de usuarios ordinarios (**nouser**) y que los programas se ejecutan en éste (**exec**).

Ahora HAL administra los dispositivos extraíbles, como discos CD-ROM y flexibles. HAL utiliza sus propios archivos para configurar las opciones de estos dispositivos. Para evitar HAL, coloque sus propias entradas en el archivo **/etc/fstab** para CD-ROM. Sin embargo, esto no permitirá la detección automática de sus CD-ROM y DVD-ROM.

En una configuración de HAL, un CD-ROM tiene opciones **ro** y **noauto**. **ro** especifica que el dispositivo es de sólo lectura; **noauto**, que no se monta automáticamente. La opción **noauto** se utiliza con unidades CD-ROM y flexibles, para que no se monten automáticamente, porque no sabe si tiene algo en éstos cuando inicia. Al mismo tiempo, las entradas HAL para las unidades CD-ROM y flexibles especifican dónde se montarán cuando decida hacerlo. La opción **users** le permite a cualquier usuario montar el sistema, lo que es útil para dispositivos extraíbles. La opción **group** sólo permite montarlo a los usuarios que pertenecen al grupo del dispositivo. SELinux utiliza la opción **fscontext**. En la tabla 29-8 se presentan las opciones para montar un sistema de archivos. Un ejemplo de una entrada de disco duro sería:

```
/dev/VolGroup00/LogVol100    /      ext3    defaults    1    1
```

Opción	Descripción
async	Indica que todas las entradas y salidas del sistema de archivos se deben hacer de forma asíncrona.
auto	Indica que el sistema de archivos se monta con la opción -a . Un comando mount -a ejecutado al inicio del sistema, monta automáticamente el sistema de archivos.
defaults	Utiliza opciones predeterminadas: rw , suid , dev , exec , auto , nouser y async .
dev	Interpreta un carácter o bloquea dispositivos especiales en el sistema de archivos.
group	Los usuarios que pertenecen al grupo del dispositivo tienen la opción de montar.
noauto	Indica que el sistema de archivos sólo se monta explícitamente. La opción -a no causa que el sistema de archivos se monte.
exec	Permite la ejecución de binarios.
owner	Permite montar el sistema de archivos al usuario que es el propietario del dispositivo.
nouser	Evita que un usuario ordinario (es decir, cualquiera que no sea root) monte el sistema de archivos.
fscontext	Proporciona contexto de seguridad de SELinux a cualquier sistema de archivos que no tiene uno.
remount	Intenta volver a montar un sistema de archivos ya montado. Esto suele utilizarse para cambiar las marcas de montaje de un sistema de archivos, sobre todo para que un sistema de archivos de sólo lectura permita la escritura.
ro	Monta el sistema de archivos como sólo lectura.
rw	Monta el sistema de archivos como lectura/escritura.
suid	Permite que tengan efecto los bits set-user-identifier o set-group-identifier.
sync	Indica que todas las entradas y salidas del sistema de archivos se deben hacer de forma sincronizada.
user	Permite que un usuario ordinario monte el sistema de archivos. Los usuarios ordinarios siempre tienen las siguientes opciones activadas: noexec , nosuid y nodev .
nodev	No interpreta dispositivos de carácter o bloqueo en el sistema de archivos.
noexec	No permite la ejecución de binarios en los sistemas de archivos montados.
nosuid	No permite que tomen efecto los bits set-user-identifier o set-group-identifier.

TABLA 29-8 Opciones de montaje para sistemas de archivos

Boot y revisión de disco

Los dos últimos campos de una entrada **fstab** constan de valores enteros. El primero lo utiliza el comando **dump** para determinar si un sistema de archivos necesita volcarse, respaldando el sistema de archivos. El segundo valor es utilizado por **fsck** para ver si un sistema de archivos debe revisarse al reiniciar, y el orden que ocuparía en relación con otros sistemas de archivos. Si el campo tiene un valor 1, indica una partición de arranque (boot), y 2 indica otras particiones. El valor 0 significa que **fsck** no necesita revisar el sistema de archivos.

Ejemplo de fstab

Aquí se muestra una copia de un archivo `/etc/fstab`. Observe que la primera línea es un comentario, traducida en este caso para mejor referencia. Todas las líneas de comentario comienzan con un `#`. Las entradas para los sistemas de archivos `/proc` y `/sys` son especializadas; su sistema operativo Linux las utiliza para administrar sus procesos y dispositivos; no son dispositivos reales. Para crear una entrada en el archivo `/etc/fstab`, puede editar directamente el archivo mencionado. Puede utilizar el archivo `/etc/fstab` de ejemplo que se muestra aquí como una guía para mostrar el aspecto que deben tener las entradas. Las entradas de las particiones `/proc` y `swap` son particularmente críticas.

<code>/etc/fstab</code>					
# <dispositivo>	<punto-montaje>	<tipo-sistema-archivo>	<opciones>	<volcado>	<fsck>
<code>/dev/hda3</code>	<code>/</code>	<code>ext3</code>	<code>defaults</code>	<code>0</code>	<code>1</code>
<code>none</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>none</code>	<code>/sys</code>	<code>sysfs</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>none</code>	<code>/dev/pts</code>	<code>devpts</code>	<code>gid=5,mode=620</code>	<code>0</code>	<code>0</code>
<code>none</code>	<code>/dev/shm</code>	<code>tmpfs</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>/dev/hda2</code>	<code>swap</code>	<code>swap</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>/dev/hda1</code>	<code>/mnt/windows</code>	<code>vfat</code>	<code>defaults</code>	<code>0</code>	<code>0</code>

Etiquetas de partición: e2label

Linux utiliza etiquetas de sistema de archivos para `ext2` y `ext3` en las particiones del disco duro. Por tanto, en el archivo `/etc/fstab` que se mostró antes, la primera entrada utiliza una etiqueta para su nombre de dispositivo, como se muestra aquí. En este caso, la etiqueta es la diagonal, `/`, que indica la partición raíz. Se cambia esta etiqueta del dispositivo con `e2label`, pero asegúrese de cambiar también la entrada de éste en `/etc/fstab`.

```
LABEL=/          /          ext3    defaults    0    1
```

En el caso de particiones `ext2` y `ext3`, se cambia o se agrega una etiqueta con la herramienta `e2label` o `tune2fs` con la opción `-L`. Especifique el dispositivo y el nombre de etiqueta. Si cambia una etiqueta, asegúrese de cambiar las entradas correspondientes en el archivo `/etc/fstab`. Sólo use `e2label` con el nombre del dispositivo para conocer cuál es la etiqueta actual. En el siguiente ejemplo, el usuario cambia la etiqueta del dispositivo `/dev/hda3` por `TORTUGA`:

```
e2label /dev/hda3  TORTUGA
```

Particiones de Windows

Puede montar particiones MS-DOS, Windows 95/98/Me o Windows XP, NT y 2000 para utilizar su sistema operativo Windows en su estructura de archivos de Linux, de la misma forma en que lo haría al montar cualquier sistema de archivos de Linux. Tiene que especificar el tipo de archivos `vfat` para Windows 95/98/Me y `msdos` para MS-DOS. Windows XP, NT y 2000 utiliza el tipo de archivos `ntfs`. Tal vez le resulte conveniente hacer que sus particiones de Windows se monten automáticamente cuando inicia su sistema Linux. Para esto, necesita poner una entrada para sus particiones de Windows en su archivo `/etc/fstab` y darle la opción `defaults` o asegurarse de incluir una opción `auto`. Cree una entrada para cada partición de Windows que quiera montar y después especifique el nombre de dispositivo para esa partición, seguido por el directorio en que quiere montarlo. El directorio `/mnt/windows` es una opción lógica (asegúrese de que ya se ha creado el directorio `windows` en `/mnt`). En el siguiente ejemplo se muestra una entrada de partición de Windows estándar para un archivo `/etc/fstab`. Observe que la última entrada en el archivo `/etc/fstab` de ejemplo es una entrada para montar una partición de Windows.


```
/dev/hda1 /mnt/windows vfat defaults 0 0
```

En el caso de Windows XP, NT y 2000, se especifica el tipo **ntfs** o **ntfs-3g**. Asegúrese de tener instalado soporte a NTFS, como el módulo NTFS del proyecto Linux-NTFS (linux-ntfs.sourceforge.net), o el controlador de lectura/escritura del proyecto NTFS-3G (www.ntfs-3g.org). El controlador NTFS-3G proporciona un soporte para lectura y escritura estable. Se incluye con casi todas las distribuciones. La herramienta de configuración **ntfs-config** le permite configurar sus particiones de manera sencilla en GNOME o KDE al utilizar NTFS-3G. El módulo kernel del proyecto Linux-NTFS es una solución más antigua que sólo proporciona capacidad de lectura.

```
/dev/hda2 /mnt/windows ntfs-3g defaults 0 0
```

NOTA El controlador NTFS-3G usa el sistema de archivos en el espacio del usuario (FUSE, *Filesystem in Userspace*). FUSE implementa un sistema de archivos virtual en el espacio del usuario, que actúa como una conexión con las operaciones de administración del sistema de archivos del kernel. Con NTFS-3G, los usuarios configuran un sistema de archivos virtual para una partición NTFS, cuyas acciones son controladas por el kernel. FUSE se ha implementado en otros sistemas operativos como Mac OSX y Windows XP para diferentes tareas. Es notable el sistema de archivos GmailFS que trata el almacenamiento de Gmail como si fuera un sistema de archivos. Consulte fuse.sourceforge.net para conocer más detalles.

Interfaces del kernel de Linux

Su archivo **/etc/fstab** tal vez también tenga entradas para dos interfaces de sistemas de archivos del kernel: **devpts** y **tmpfs**. Ambos proporcionan interfaces de kernel que no tienen soporte en dispositivos estándar. La entrada **/dev/pts** monta un sistema de archivos **devpts** para pseudoterminales. La entrada **/dev/shm** monta el sistema de archivos **tmpfs** (también conocido como **shmfs**) para implementar la memoria virtual de Linux, mantenimiento de acceso de memoria compartida POSIX. Esto está diseñado para sobreponerse a la limitación de memoria de 4 GB en los sistemas actuales, extendiendo la memoria utilizable a 64 GB.

Si alguna vez se corrompe su archivo **/etc/fstab** (por ejemplo, si una línea se elimina por accidente o se cambia), su sistema iniciará en modo de mantenimiento, dándole acceso de sólo lectura a sus particiones. Para obtener acceso de lectura/escritura para corregir el archivo **/etc/fstab**, tiene que volver a montar su partición principal. El siguiente comando realiza tal operación:

```
# mount -n -o remount,rw /
```

noauto

Los sistemas de archivos que se incluyen en el archivo **/etc/fstab** se montan automáticamente cuando inicia, a menos que esta característica se desactive de manera explícita con la opción **noauto**. Observe que los discos CD-ROM y flexible en el archivo **fstab** del ejemplo anterior de este capítulo tienen una opción **noauto**. Además, si utiliza un comando **mount -a**, se montan todos los sistemas de archivo sin una opción **noauto**. Si quiere que el usuario pueda montar el CD-ROM, agregue la opción **user**.

```
/dev/hdc /media/cdrom iso9660 ro,noauto,user 0 0
```


Opción de montaje	Descripción
-f	Finge el montaje de un sistema de archivos. Se utiliza para revisar si un sistema de archivos se puede montar.
-v	Modo textual extenso. mount despliega descripciones de las acciones que está tomando. Se utiliza con -f para revisar si hay problemas al montar un sistema de archivos, -fv .
-w	Monta el sistema de archivos con permiso de lectura/escritura.
-r	Monta el sistema de archivos con permiso de sólo lectura.
-n	Monta el sistema de archivos sin colocar una entrada para éste en el archivo mstab .
-t tipo	Especifica el tipo de sistema de archivo que se monta. Consulte la tabla 29-7 para ver tipos de sistemas de archivos válidos.
-a	Monta todos los sistemas de archivos que se incluyen en /etc/fstab .
-o lista-opciones	Monta el sistema de archivos al utilizar una lista de opciones. Ésta es una lista separada por comas de opciones después de -o . Véase en la Tabla 29-8 una lista de opciones.

TABLA 29-9 El comando **mount**

***SUGERENCIA** El montaje “automático” de los sistemas de archivos de **letclfstab** realmente se implementa al ejecutar un comando **mount -a** en la secuencia de comandos de inicio (como **/etc/rc.d/rc.sysinit**) que se ejecuta siempre que inicia. El comando **mount -a** monta cualquier sistema de archivos incluido en su archivo **/etc/fstab** que no tiene una opción **noauto**. El comando **umount -a** (que se ejecuta cuando apaga su sistema), desmonta el sistema de archivos en **letclfstab**.*

Montaje manual del sistemas de archivos: mount y umount

También puede montar o desmontar cualquier sistema de archivos utilizando directamente los comandos **mount** y **umount**. Las operaciones de montaje que se analizaron en las secciones anteriores utilizan el comando **mount** para montar un sistema de archivos. Por lo general, sólo el usuario root puede montar los sistemas de archivos en particiones de disco duro, mientras que cualquier usuario puede montar los CD-ROM y discos flexibles. En la tabla 29-9 se muestra una lista de diferentes opciones para el comando **mount**.

El comando mount

El comando **mount** toma dos argumentos: el dispositivo de almacenamiento que permite que Linux acceda al sistema de archivos, y el directorio en la estructura de archivos al que se conecta el nuevo sistema de archivos. El *punto de montaje* es el directorio en su árbol de directorios principal donde quiere que se conecten los archivos en el dispositivo de almacenamiento. El *dispositivo* es un archivo de dispositivo especial que conecta su sistema al dispositivo de hardware. La sintaxis para el comando **mount** es la siguiente:

```
# mount device mountpoint
```

Como ya se observó, los archivos de dispositivo se ubican en los directorios **/dev** y suelen tener nombres abreviados que terminan con el número de dispositivo. Por ejemplo, **fd0** se refiere a la primera unidad de disco flexible conectada a su sistema. En el siguiente ejemplo se monta un disco duro en el primero (**hdc2**) al directorio **/mismedios**. Es necesario que el directorio de punto de montaje esté vacío. Si ya tiene un sistema de archivos montado ahí, recibirá un mensaje de que otro sistema de archivos ya está montado y de que el directorio está ocupado. Si monta un sistema de archivos a un directorio que ya tiene archivos y subdirectorios, éstos se evitarán, dándole acceso sólo a los archivos en el sistema de archivos montado. Al desmontar un sistema de archivos, por supuesto, se restaura el acceso a los archivos del directorio original.

```
# mount /dev/hdc2 /mismedios
```

En el caso de cualquier partición con una entrada en el archivo **/etc/fstab**, puede montar la partición empleando sólo el directorio de montaje especificado en su entrada **fstab**; no es necesario que inserte el nombre de archivo del dispositivo. El comando **mount** busca la entrada de la partición en el archivo **fstab**, empleando el directorio para identificar la entrada y, de esa forma, encontrar el nombre del dispositivo. Por ejemplo, para montar la partición de Windows **/dev/hda1** en el ejemplo anterior, el comando **mount** sólo necesita conocer el directorio en que está montado (en este caso, **/mnt/windows**).

```
# mount /mnt/windows
```

Si no está seguro acerca del tipo de sistema de archivos que contiene un disco, puede montarlo al especificar el tipo de sistema de archivos **auto** con la opción **-t**. Una vez que se ha dado el tipo de sistema de archivos **auto**, **mount** intenta detectar automáticamente el tipo de sistema de archivos en el disco. Esto resulta útil si está montando de forma manual un disco flexible de cuyo tipo de sistema de archivos no está seguro (HAL también detecta automáticamente el tipo de sistema de archivos de cualquier medio extraíble, incluidos los discos flexibles).

```
# mount -t auto /dev/fd0 /media/floppy
```

El comando **umount**

Si quiere reemplazar un sistema de archivos montado con otro, primero debe desmontar explícitamente el que ya está montado. Digamos que ha montado un disco flexible y ahora quiere sacarlo e insertar uno nuevo. Primero debe desmontar el disco flexible antes de insertar y montar uno nuevo. Se desmonta un sistema de archivos con el comando **umount**. El comando **umount** toma como argumento un nombre de dispositivo o el directorio donde fue montado. Aquí se muestra la sintaxis:

```
# umount dispositivo-o-puntodemontaje
```

En el siguiente ejemplo se desmonta el disco flexible, sin importar dónde esté montado:

```
# umount /dev/fd0
```

Empleando el ejemplo donde el dispositivo está montado en el directorio **/midir**, puede usar ese directorio para desmontar el sistema de archivos:

```
# umount /midir
```

Una restricción importante aplica al comando **umount**: nunca podrá desmontar un sistema de archivos en que esté trabajando. Si se cambia a un directorio dentro del sistema de archivos y

después intenta desmontarlo, se recibe un mensaje de error que indica que el sistema de archivos está ocupado. Por ejemplo, suponga que un CD-ROM está montado en el directorio **/media/disk** y después se cambia al directorio **/media/disk**. Si decide cambiar el CD-ROM, primero tendrá que desmontar el disco actual con el comando **umount**. Esto fallará porque se encuentra actualmente en el directorio en que está montado. Tiene que salir del directorio antes de desmontar el CD-ROM.

```
# mount /dev/hdc /media/disk
# cd /media/disk
# umount /media/disk
umount: /dev/hdc: device is busy
# cd /root
# umount /media/disk
```

SUGERENCIA Si otros usuarios están utilizando un sistema de archivos que está tratando de desmontar, puede utilizar el comando **lsdf** o **fuser** para saber quiénes son.

Montaje de discos flexibles

Como ya se observó, para acceder a un archivo en un disco flexible, primero tiene que montarse el disco en su sistema Linux. El nombre de dispositivo para su unidad de disco flexible es **fd0**, y se ubica en el directorio **/dev**. La inserción de **/dev/fd0** hace referencia a su unidad de disco flexible. Observe el número **0** después de **fd**. Si tiene más de una unidad de disco flexible, las unidades adicionales se representan con **fd1**, **fd2**, etc. Se monta en cualquier directorio que quiera. Algunas distribuciones crean un directorio conveniente para uso con discos flexibles, **/media/floppy**. En el siguiente ejemplo se monta el disco flexible en su unidad de disco flexible en el directorio **/media/floppy**:

```
# mount /dev/fd0 /media/floppy
```

SUGERENCIA En GNOME, se monta una unidad de disco flexible al hacer clic con el botón derecho en el fondo del escritorio para desplegar el menú de escritorio y después seleccionar *Disquete*, en la entrada *Discos*. Para desmontar, haga clic con el botón derecho en el icono de disco flexible, y seleccione *Expulsar* del menú desplegable.

Recuerde que está montando un disco flexible particular, no la unidad de disco flexible. No puede simplemente quitar un disco flexible y poner otro. El comando **mount** ha conectado los archivos a su árbol de directorio principal, y su sistema espera encontrar esos archivos en un disco flexible en su unidad. Si saca el disco y pone otro, recibirá un mensaje de error cuando intente acceder a éste.

Para cambiar los discos, primero debe desmontar el disco flexible que ya está en su unidad de disco. Entonces, después de poner el nuevo disco, debe montar explícitamente el nuevo disco. Para esto, utilice el comando **umount**.

```
# umount /dev/fd0
```

En el caso de las operaciones **umount** o **mount**, se especifica el directorio donde está montado o el dispositivo **/dev/fd0**.

```
# umount /media/floppy
```

Ahora se puede quitar el disco flexible, poner uno nuevo y después montarlo:

```
# mount /media/floppy
```

Cuando apaga su sistema, cualquier disco que haya montado se desmontará de manera automática. No tiene que desmontarlo explícitamente.

Montaje de CD-ROM

Recuerde que cuando se monta un disco CD-ROM o flexible, no puede luego simplemente quitarlo y poner otro en la unidad. Primero tiene que desmontarlo, desconectando el sistema de archivos del árbol de directorios general. En realidad, la unidad CD-ROM permanece bloqueada hasta que la desmonte. Una vez que desmonte un CD-ROM, puede sacar uno y poner otro, mismo que tiene que montar después para acceder a él. Cuando cambia varios discos CD-ROM o flexibles, estará montándolos y desmontándolos continuamente. En el caso de un CD-ROM, en vez de utilizar el comando **umount**, puede utilizar el comando **eject** con el nombre del dispositivo o el punto de montaje, con lo que desmontará y expulsará el CD-ROM de la unidad.

Para montar un CD-ROM, todo lo que tiene que hacer es insertarlo en la unidad. HAL lo detectará y montará automáticamente en el directorio **/media/disk**.

Si, en cambio, quiere montar la unidad automáticamente desde la línea de comandos con el comando **mount**, tendrá que decidir primero en qué directorio lo montará (se creará, si no existe). El directorio **/media/disk** se crea de forma dinámica cuando se inserta un disco y se elimina cuando el disco se quita. Para montar un disco de forma manual, utilice el comando **mount**, el nombre de dispositivo, como **/dev/cdrom**, y el directorio donde se monta.

```
# mount /dev/cdrom /media/cdrom1
```

Si quiere desmontar una unidad de forma manual, digamos desde la línea de comandos, se utiliza el comando **umount** y el nombre del directorio donde está montado:

```
# umount /media/cdrom1
```

O si se monta con HAL, puede utilizar

```
# umount /media/disk
```

SUGERENCIA En GNOME, los CD-ROM se montan automáticamente, aunque puede montarlos de manera manual al hacer clic con el botón derecho en el fondo del escritorio para desplegar el menú de escritorio y seleccionar CD-ROM en la entrada Discos. Para desmontarlo, haga clic con el botón derecho en el icono del CD-ROM y seleccione Expulsar del menú desplegable.

Cuando graba un CD, tal vez necesite crear un archivo de imagen de CD. Se accede a tal archivo de imagen desde su disco duro, que se monta como si fuera otro sistema de archivos (incluso las imágenes extraídas se pueden montar de esta forma). Para esto, se utiliza la opción **loop**, que especifica un dispositivo de loop abierto como **/dev/loop0**. Si no se indica un dispositivo loop, **mount** intentará hallar uno abierto. El tipo de sistema de archivos es **iso9660**, un tipo de archivo de imagen ISO de CD-ROM.

```
# mount -t iso9660 -o loop=/dev/loop0 archivo-imagen directorio-montaje
```

Para montar el archivo de imagen **mimusic.cdimage** al directorio **/mnt/miscosas** y hacerlo de sólo lectura, utilizaría:

```
# mount -t iso9660 -o ro,loop=/dev/loop0 mimusica.cdimage /mnt/miscosas
```

Una vez que está montado, se accede a los archivos en el CD-ROM como si se tratara de cualquier otro directorio.

SUGERENCIA Puede utilizar **mkisofs** para crear una imagen de CD-ROM hecha con sus archivos u otro CD-ROM.

Montaje de particiones de disco duro: Linux y Windows

Puede montar particiones de disco duro de Linux o Windows con el comando **mount**. Sin embargo, es mucho más práctico hacer que se monten automáticamente al utilizar el archivo **/etc/fstab**, como se describió antes. Las particiones de disco duro de Linux que creó durante la instalación ya se montan automáticamente. Como ya se indicó, para montar una partición de disco duro de Linux, debe ingresar el comando **mount** con el nombre del dispositivo de la partición y el directorio en que quiere montarlo. Los discos duros IDE utilizan el prefijo **hd**, y los SCSI utilizan el prefijo **sd**. En el siguiente ejemplo se monta la partición de disco duro de Linux en **/dev/hda4** en el directorio **/mnt/misdatos**.

```
# mount -t ext3 /dev/hda4 /mnt/misdatos
```

También puede montar particiones de Windows y acceder directamente a sus archivos. Al igual que con una partición de Linux, se utiliza el comando **mount**, pero también tiene que especificar el tipo de sistema de archivos, como Windows. Para eso, se utiliza la opción **-t** y después se escribe **vfat** para Windows 95/98/Me (**msdos** para MS-DOS). En el caso de Windows XP, 2000 y NT, se utiliza **ntfs** (acceso de sólo lectura completo con acceso de escritura limitado). En el siguiente ejemplo, el usuario monta la partición de disco duro de Windows **/dev/hda1** en la estructura de archivos de Linux en el directorio **/mnt/windows**. El directorio **/mnt/windows** es una designación común para sistemas de archivos de Windows, aunque se monta en cualquier directorio (como **/mnt/dos** para MS-DOS). Si tiene varias particiones de Windows, puede crear un directorio de Windows y después un subdirectorio para cada unidad, mediante la etiqueta del disco o la letra, como **/mnt/windows/a** o **/mnt/windows/miscosas**. Asegúrese de que ya ha creado el directorio antes de montar el sistema de archivos.

```
# mount -t vfat /dev/hda1 /mnt/windows
```

NOTA El soporte de entrada y salida mejorado mediante **SG_IO** elimina la necesidad de hacer que unidades IDE CD-R y DVD-R emulen dispositivos SCSI. El soporte se implementa directamente con el kernel. Para comprobar que se reconozcan sus unidades IDE CD/DVD, ejecute **cdrecord** con la opción **-scanbus**.

Creación de sistemas de archivos: **mkfs**, **mke2fs**, **mkswap**, **parted** y **fdisk**

Linux proporciona varias herramientas para crear y administrar sistemas de archivos, permitiéndoles agregar nuevas particiones de disco duro, crear imágenes de CD y dar formato a discos flexibles. Para utilizar un nuevo disco duro, tendrá primero que particionarlo y después crear un sistema de archivos en él. Puede utilizar **parted** o **fdisk** para particionar su disco duro. Si desea crear el sistema de archivos en las particiones, se utiliza el comando **mkfs**, que es un portal para varios generadores de sistemas de archivos. En el caso de particiones de intercambio, se utiliza una herramienta especial, **mkswap**, y para crear sistemas de archivos en un CD-ROM, se utiliza la herramienta **mkisofs**. Las herramientas de partición y de sistemas de archivos de Linux se muestran en la tabla 29-10.

Herramienta	Descripción
fdisk	Programa orientado a menús que crea y elimina particiones.
cfdisk	Interfaz basada en pantalla para fdisk .
parted	Administra particiones GNU, utiliza GParted o QTParted para interfaz GUI.
mkfs	Crea un sistema de archivos en una partición o disco flexible al utilizar el tipo de sistema de archivos especificado. Portal para utilerías de formato.
mke2fs	Crea un sistema de archivos ext2 en una partición de Linux; se utiliza la opción -j para crear el sistema de archivos ext3 .
mkfs.ext3	Crea un sistema de archivos ext3 en una partición de Linux.
mkfs.ext2	Crea un sistema de archivos ext2 en una partición de Linux.
mkfs.reiserfs	Crea un sistema de archivos de registro por diario de ReiserFS en una partición de Linux (vincula con mkreiserfs).
mkfs.jfs	Crea un sistema de archivos de registro por diario de JFS en una partición de Linux.
mkfs.xfs	Crea un sistema de archivos de registro por diario de XFS en una partición de Linux.
mkfs.dos	Crea un sistema de archivos DOS en una partición dada.
mkfs.vfat	Crea un sistema de archivos de 16 bits de Windows en una partición dada (Windows 95/98/Me).
mkfs.cramfs	Crea un sistema de archivos de memoria flash comprimido de CramFS, de sólo lectura (se utiliza para dispositivos incrustados).
mkswap	Configura un área de intercambio de Linux en un dispositivo o en un archivo.
mkdosfs	Crea un sistema de archivos MS-DOS bajo Linux.
mkisofs	Crea una imagen de disco CD-ROM ISO.
dumpe2fs	Despliega información de bloque de bajo nivel para un sistema de archivos.
gfloppy	Herramienta de GNOME que da formato a un disco flexible (la entrada Formateador de disquetes en el menú Herramientas del sistema).
resize2fs	Extiende el tamaño de una partición, que utiliza el espacio no utilizado disponible en un disco.
hdparm	“Afinador” de discos duros IDE; configura las características de discos duros IDE.
tune2fs	Ajusta un sistema de archivos, que configura características como etiqueta, registro por diario y bloque de espacio reservado.

TABLA 29-10 Herramientas de creación de particiones y sistemas de archivos en Linux

fdisk

Para iniciar **fdisk**, inserte **fdisk** en la línea de comandos con el nombre de dispositivo del disco duro que está particionando. Esto inicia un programa interactivo que se utiliza para crear sus particiones de Linux. Tenga cuidado al utilizar **fdisk** de Linux. De lo contrario, puede eliminar, literalmente, particiones completas de disco duro y todos los datos incluidos en esas particiones. El siguiente comando invoca **fdisk** para crear particiones en el disco duro **hdb**.

```
fdisk /dev/hdb
```

Comando	Acción
a	Activa o desactiva una marca para arranque.
l	Muestra los tipos de partición conocidos.
m	Despliega comandos.
n	Agrega una nueva partición.
p	Imprime la tabla de partición.
q	Sale sin guardar los cambios.
t	Cambia el ID de sistema de una partición.
w	Escribe la tabla en el disco y sale.

TABLA 29-11 Comandos de **fdisk** utilizados comúnmente

Las particiones tienen diferentes tipos, mismos que necesita especificar. **fdisk** de Linux es un programa orientado a línea. Tiene un conjunto de comandos de un carácter que sólo tiene que presionar. Luego tal vez se le pida que escriba información y que oprima **ENTER**. Si se encuentran problemas durante el procedimiento de **fdisk**, oprima **q** en cualquier momento y se le regresará a la pantalla anterior sin aplicar ningún cambio. En realidad no se hace ningún cambio a su disco duro hasta que oprime **w**. Éste debe ser su último comando; aplica los cambios reales a su disco duro y después sale de **fdisk**, para regresarlo a su programa de instalación. En la tabla 29-11 se muestra una lista de los comandos de uso común de **fdisk**. Siga los pasos indicados en este párrafo para crear una partición de Linux. Oprima **n** para definir una nueva partición; se le preguntará si es la partición primaria. Oprima **p** para indicar que es la partición primaria. Linux soporta hasta cuatro particiones primarias. Inserte el número de la partición que quiera crear y el cilindro inicial de la partición (es el primer número entre paréntesis, al final del indicador). Entonces se le pide que inserte el número del último cilindro. Puede insertar el último cilindro que quiera para esta partición o insertar el tamaño. Por ejemplo, se inserta el tamaño como **+1000M** para 1 GB, con un signo **+** antes de la cantidad. Tenga en cuenta que el tamaño no excede el espacio libre. Luego se especifica el tipo de partición. El tipo predeterminado para una partición es 83. Si crea un tipo de partición diferente, como una de intercambio, oprima **t** para indicar que es el tipo que quiere. Inserte el número de partición, como 82 para una de intercambio. Cuando termine, oprima **w** para escribir los cambios en el disco duro, y después oprima **ENTER** para continuar.

parted

Como una opción a **fdisk**, se utiliza **parted** (gnu.org/software/parted), que le permite administrar particiones de disco duro, crear nuevas y eliminar antiguas. A diferencia de **fdisk**, también le permite cambiar el tamaño de las particiones. Para utilizar **parted** en particiones de un disco duro determinado, no se utiliza ninguna de estas particiones en esa unidad. Esto significa que si quiere utilizar **parted** en particiones localizadas en el mismo disco duro que su kernel, tiene que iniciar su sistema en modo de rescate y seleccionar que no se monten sus archivos de sistema. En el caso de cualquier otro disco duro, sólo necesita desmontar sus particiones y desactivar su espacio de intercambio con el comando **swapoff**. Luego se inicia **parted** con el comando **parted** y el nombre de dispositivo de su disco duro en el que quiere trabajar. Como opción, se utiliza **GParted** en **GNOME** o **QTParted** en **KDE**. En el siguiente ejemplo se inicia **parted** para el disco duro **/dev/hda**.

`parted /dev/hda`

Utilice el comando **print** para desplegar una lista de todas sus particiones. El número de cada partición aparecerá en la primera columna bajo el encabezado Minor. Las columnas Start y End muestran las posiciones de inicio y fin que la partición utiliza en el disco duro. Los números están en megabytes, y van del primer megabyte hasta el total disponible. Para crear una nueva partición, utilice el comando **mkpart** con **primary** o **extended**, el tipo de sistema de archivos y las posiciones de inicio y fin. Se crean las tres particiones primarias y una partición extendida (o cuatro particiones primarias, si no hay particiones extendidas). La partición extendida, a su vez, tiene varias particiones lógicas. Una vez que haya creado la partición, puede utilizar **mkfs** para dar formato al sistema de archivos. Para eliminar una partición, se utiliza el comando **rm** y el número de partición. Para cambiar el tamaño de una partición, se utilizan el comando **resize** con el número de partición y las posiciones de inicio y fin. Incluso puede mover una partición al utilizar el comando **move**. El comando **help** presenta una lista de todos los comandos.

mkfs

Una vez que cree su partición, tiene que crear un sistema de archivos en ésta. Para esto, utilice el comando **mkfs**, que genera el sistema de archivos de Linux y pasa el nombre de la partición del disco duro como parámetro. Debe especificar su nombre de ruta completo con el comando **mkfs**. En la tabla 29-12 se presenta una lista de las opciones para el comando **mkfs**. Por ejemplo, la segunda partición en la primera unidad de disco duro tiene el nombre de dispositivo **/dev/hdb1**. Ahora puede montar su nueva partición de disco duro, conectándola a la estructura de archivos. En el siguiente ejemplo se da formato a esa partición:

```
# mkfs -t ext3 /dev/hdb1
```

El comando **mkfs** sólo es un portal para varios generadores de sistema de archivos. Un *generador de sistema de archivos* realiza la tarea real de crear un sistema de archivos. Linux da soporte a varios generadores, incluidos varios sistemas de archivos de registro por diario y de Windows. El nombre de un generador de sistema de archivos tiene el prefijo **mkfs** y el sufijo del nombre del tipo de

Opción	Descripción
<i>Bloques</i>	Especifica el número de bloques para el sistema. Existen 1 440 bloques en un disco flexible 1.44 MB.
-t tipo-sistema-archivos	Especifica el tipo de sistema de archivos que se formateará. La opción predeterminada es el tipo de sistema de archivos estándar de Linux, ext3 .
<i>opciones-sistema-archivos</i>	Opciones para el tipo de sistema de archivos especificado. Se muestra antes del nombre de dispositivo, pero después del tipo de sistema de archivos.
-v	Modo extenso de texto. Despliega una descripción para cada acción que toma mkfs .
-v	Instruye al programa de generador de sistema de archivos que invoca mkfs para mostrar las acciones que toma.
-c	Revisa una partición en busca de bloques defectuosos antes de formatearla (tal vez ocupe mucho tiempo).
-l nombredearchivo	Lee una lista de bloques defectuosos.

TABLA 29-12 Opciones de **mkfs**

sistema de archivos. Por ejemplo, el generador del sistema de archivos **ext3** es **mkfs.ext3**. En el caso de sistemas de archivos ReiserFS, es **mkfs.reiserfs** (vincula a **mkreiserfs**, que es parte del paquete **reiser-utils**). En el caso de sistemas de archivos de 16 bits de Windows (95/98/Me), es **mkfs.vfat**. Algunos de estos generadores de archivos son sólo otros nombres para las herramientas de creación de sistema de archivos tradicionales. Por ejemplo, el generador de sistema de archivos **mkfs.ext2** es sólo otro nombre para la herramienta de creación de sistema de archivos **mke2fs ext2**, y **mkfs.msdos** es el comando **mkdosfs**. Como **ext3** es una extensión de **ext2**, **mkfs.ext3** simplemente invoca a **mke2fs**, la herramienta para crear sistemas de archivos **ext2** y **ext3**, y lo instruye para crear un sistema de archivos **ext3** (al utilizar la opción **-j**). Cualquiera de estos generadores se utiliza directamente para crear un sistema de archivos de ese tipo. Las opciones se muestran antes del nombre de dispositivo. El siguiente ejemplo es equivalente al anterior, que crea un sistema de archivos **ext3** en el dispositivo **hdb1**:

```
mkfs.ext3 /dev/hdb1
```

La sintaxis del comando **mkfs** es la siguiente. Se agregan opciones para un sistema de archivos particular después del tipo y antes del dispositivo. El tamaño de bloque se utiliza para generadores que no detectan el tamaño del disco.

```
mkfs opciones [-t tipo] opciones-sis-de-archivos dispositivo tamaño
```

SUGERENCIA Una vez que haya dado formato a su disco, se etiqueta con el comando **e2label**, como se describió en páginas anteriores de este capítulo.

El mismo procedimiento funciona para discos flexibles. En este caso, el comando **mkfs** toma como argumento el nombre de dispositivo. Utiliza el sistema de archivos **ext2** (el sistema predeterminado para **mkfs**), porque un disco flexible es muy pequeño para soportar un sistema de archivos de registro por diario.

```
# mkfs /dev/fd0
```

SUGERENCIA **GParted** proporciona una interfaz GUI para **parted** en **GNOME**. En **KDE** se utiliza **QTParted**. Ambos proporcionan una interfaz muy fácil de usar para administrar particiones.

mkswap

Si quiere crear una partición de intercambio, primero utilice **fdisk** o **parted** para crear la partición (si aún no existe) y después utilice el comando **mkswap** para darle formato como partición de intercambio. **mkswap** da formato a toda la partición, a menos que se indique de otra forma. Toma como argumento el nombre de dispositivo para la partición de intercambio.

```
mkswap /dev/hdb2
```

Entonces necesita crear una entrada para ésta en el archivo **/etc/fstab**, para que se monte automáticamente cuando inicia su sistema.

Grabación de CD-ROM y DVD-ROM

Ahora la grabación de datos en discos DVD y CD-ROM de Linux se maneja directamente en los escritorios de **GNOME** y **KDE**. Las operaciones simples de arrastrar y soltar en un disco DVD y CD en blanco le permiten grabar un disco. También se utilizan las aplicaciones de grabación de CD de

GNOME y KDE como KOnCD y Toaster de GNOME para crear sus DVD y CD de forma sencilla. Todos son portales de las herramientas **mkisofs** y **cdrecord**. Para grabar un DVD en quemadores de DVD, se utiliza **cdrecord** para unidades DVD-R/RW y las herramientas de DVD+RW para unidades DVD+RW/R. Si quiere grabar un CD-ROM en un quemador de DVD, basta con utilizar la aplicación **cdrecord** con muchas opciones.

En la actualidad, la aplicación **cdrecord** sólo trabaja con unidades DVD-R/RW; es parte del paquete **dvd+rw-tools**. Si quiere utilizar unidades DVD+RW/R, se utilizan las herramientas de DVD+RW como **growisofs** y **dvd+rw-format**. Algunas de estas herramientas se incluyen en el paquete **dvd+rw-tools**. Revise el sitio Web de las herramientas DVD+RW para conocer más información, <http://fy.chalmers.se/~appro/linux/DVD+RW>.

Con el comando **mkisofs**, se crea un archivo de imagen de CD, que después se graba en un dispositivo de grabación de CD-R/RW. Una vez que haya creado su archivo de imagen de CD, se graba en un dispositivo de grabación de CD con la aplicación **cdrecord** o **cdwrite**. La primera es la que tiene más opciones.

mkisofs

Para crear una imagen de DVD y CD, primero se seleccionan los archivos que quiere incluir. Después se utiliza **mkisofs** para crear una imagen DVD/CD ISO de éstos.

Opciones de mkisofs

Tal vez necesite incluir varias opciones importantes con **mkisofs** para crear un CD de datos apropiado. La opción **-o** especifica el nombre de un archivo de imagen de CD. Puede ser cualquier nombre que quiera. La opción **-R** especifica los protocolos de CD RockRidge, y la opción **-J** da soporte a nombres largos de Windows 95/98/Me o XP. La opción **-r**, además de los protocolos RockRidge (**-R**), configura permisos globales estándar para sus archivos, como acceso de lectura para todos los usuarios y sin acceso de escritura porque un CD-ROM es de sólo lectura. La opción **-T** crea tablas de traducción para nombres de archivos, que habrán de usarse en sistemas que no son compatibles con RockRidge. La opción **-U** da soporte a nombres de archivos relajados que no son compatibles con el ISO estándar, como nombres de archivos largos, con más de un punto en el nombre, que comienzan con un punto (como los archivos de configuración de shell) y los que utilizan caracteres en minúsculas (también existen opciones separadas para cada una de estas características, si sólo quiere usar algunas de ellas). Casi todos los nombres de paquetes de código fuente y RPM entran en esta categoría. La opción **-iso-level** le permite eliminar restricciones de ISO como el tamaño de un nombre de archivo. La opción **-v** configura la etiqueta del volumen (nombre) para el CD. Por último, la opción **-v** despliega el proceso de la creación de imagen.

Creación de imagen de disco

El último argumento es el directorio que contiene los archivos para los cuales quiere elaborar la imagen de CD. Para esto, se especifica un directorio. Por ejemplo, si está creando un CD-ROM que contenga los archivos de datos en el directorio **misdocs**, se especifica ese directorio. Este directorio superior no se incluirá, sólo los archivos y subdirectorios de éste. También se cambia a tal directorio y después se utiliza **.** para indicar el directorio actual.

Si está creando un CD simple para usarlo en Linux, primero se utiliza **mkisofs** para crear la imagen de CD. Aquí la opción de texto extenso mostrará el proceso de creación y la opción **-v** le permite especificar la etiqueta de CD. Una imagen de CD llamada **canciones.iso** se crea al utilizar el archivo ubicado en el directorio **nuevascanciones**:

```
mkisofs -v -V "Buenascanciones" -o cancioneseextra.iso nuevascanciones
```

Si también quiere utilizar el CD en un sistema Windows, se agregan las opciones **-r** (RockRidge con acceso de archivo global estándar) y **-J** (Joliet):

```
mkisofs -v -r -J -V "Buenascanciones" -o cancioneseextra.iso newsongs
```

Necesita incluir ciertas opciones, si está utilizando nombres de archivos que no son compatibles con ISO, como los que tienen más de 31 caracteres o los que usan caracteres en minúscula. La opción **-U** le permite utilizar nombres de archivos sin restricciones, mientras que ciertas opciones como **-L** para tamaño sin restricciones sólo liberarán restricciones específicas. En el siguiente ejemplo se crea una imagen de CD llamada **midoc.iso** al utilizar los archivos y subdirectorios ubicados en el directorio **mdoc**, y se etiqueta la imagen de CD con el nombre "Documentosgrandiosos":

```
mkisofs -v -r -T -J -U -V "Documentosgrandiosos" -o misdocumentos.iso mdocs
```

Montaje de imágenes de disco

Una vez que ha creado su imagen de CD, puede revisarla para ver si es correcta al montarlo como un sistema de archivos en Linux. En efecto, para probar la imagen de CD, se monta en un directorio y después se accede a éste como si fuera simplemente otro sistema de archivos. Montar una imagen de CD requiere el uso de un dispositivo de bucle. Especifique este dispositivo con la opción **loop**, como se muestra en el siguiente ejemplo. Aquí **midoc.iso** se monta en el directorio **/media/cdrom** como si fuera un sistema de archivos de tipo **iso9660**. Asegúrese de desmontarlo cuando termine.

```
mount -t iso 9660 -o ro,loop=/dev/loop0 misdocumentos.iso /media/cdrom
```

CD-ROM de arranque

Si está creando un CD-ROM de arranque, necesita indicar el archivo de imagen de arranque y el catálogo de arranque que habrán de utilizarse. Con la opción **-c**, se especifica el catálogo de arranque. Con la opción **-b**, se especifica la imagen de arranque. La *imagen de arranque* es una imagen del disco de arranque, como el que se utiliza para iniciar un procedimiento de instalación. Por ejemplo, en el CD-ROM de Fedora, la imagen de arranque es **isolinux/isolinux.bin**, y el catálogo de arranque es **isolinux/boot.cat** (también se utiliza **images/boot.img** y **boot.cat**). Copie esos archivos a su disco duro. En el siguiente ejemplo se crea una imagen de CD-ROM de arranque al utilizar los archivos de las distribuciones de Red Hat Linux y Fedora ubicados en la unidad de CD-ROM.

```
mkisofs -o fed7-0iso -b isolinux/isolinux.bin -c isolinux/boot.cat \
-no-emul-boot -boot-load-size 4 -boot-info-table \
-v -r -R -T -J -V "Fed7" /media/cdrom
```

cdrecord

Una vez que **mkisofs** ha creado la imagen de CD, se utiliza Nautilus (el administrador de archivos de GNOME) para grabar directamente una imagen ISO en un disco CD/DVD. En la interfaz de línea de comandos puede, en cambio, utilizar el comando **cdrecord** para grabar un disco CD/DVD. Existe un comando llamado **dvdrecord**, pero sólo es una secuencia de comandos que llama a **cdrecord**, que ahora graba en DVD y CD. Si tiene más de un dispositivo de grabación de CD, debe especificar la unidad de DVD/CD-R/RW que se utilizará al indicar el nombre de dispositivo. En este ejemplo, el dispositivo es un CD-R IDE ubicado en **/dev/hdc**. La opción **dev=** se utiliza para indicar la unidad. El argumento final para **cdrecord** es el nombre del archivo de imagen de CD.

```
cdrecord dev=/dev/hdc misdocumentos.iso
```

En este ejemplo, se utiliza un dispositivo regrabable CD-RW SCSI con el dispositivo `/dev/scd0`.

```
cdrecord dev=/dev/scd0 misdocumentos.iso
```

Si está creando un CD de audio, se utiliza la opción `-audio`, como se muestra aquí. Esta opción utiliza el formato de audio CD-DA:

```
cdrecord dev=/dev/hdc -audio mascanciones.iso
```

SUGERENCIA La opción *dummy* para `cdrecord` le permite probar la operación de escritura de CD para una imagen dada.

Herramientas de DVD+RW

La principal herramienta de DVD+RW es `growisofs`, con la que se crean discos DVD+RW/R. También se incluyen otras dos herramientas con soporte menor, una para formatear, `dvd+rw-format`, y una herramienta de compatibilidad, `dvd+rw-booktype`. Consulte la página de las herramientas `dvd+rw` en `/usr/share/doc` para conocer instrucciones detalladas.

La herramienta `growisofs` funciona como la herramienta `mkisofs`, excepto que graba directamente en el disco DVD+RW/R, en vez de hacerlo en una imagen. Tiene las mismas opciones que `mkisofs`, con unas cuantas excepciones, y realmente es un portal del comando `mkisofs`. Por supuesto, no existe la opción `-o` para especificar una imagen de disco. En cambio, se especifica el dispositivo DVD. Por ejemplo, para grabar el contenido del directorio `nuevascanciones` en un disco DVD+RW, usaría directamente `growisofs`.

```
growisofs -v -V "Buenascanciones" -Z /dev/hdc nuevascanciones
```

El dispositivo se especifica con su nombre, por lo general `/dev/scd0` para el primer dispositivo SCSI o `/dev/hdc` para la primera unidad IDE secundaria. Recuerde que los quemadores de DVD IDE se configuran como dispositivos SCSI cuando su sistema inicia. `growisofs` proporciona una opción especial `-z` para grabar una sesión inicial. En el caso de varias sesiones (DVD-RW), se utiliza la opción `mkisofs -M`. Si quiere volver a utilizar un disco DVD-RW, sólo reescríbalo. No tiene que darle formato.

Para grabar un archivo de imagen ISO en un disco, se utiliza la opción `-z` y se asigna la imagen ISO al dispositivo.

```
growisofs -v -V "Buenascanciones" -Z /dev/hdc=mascanciones.iso
```

Aunque `growisofs` dará formato automáticamente a un nuevo disco DVD+RW, entre las herramientas de DVD+RW también se incluye `dvd+rw-format` para dar formato explícitamente a nuevos discos DVD+RW (lectura/escritura), lo que los prepara para la grabación. Esto se hace sólo una vez, y sólo en discos DVD+RW que nunca se han utilizado. Los discos DVD+R no necesitan formato.

La herramienta `dvd+rw-booktype` configura las opciones de compatibilidad para lectores DVD-ROM antiguos que no puedan leer los discos DVD+RW/R.

Mono y soporte a .NET

Con Mono, Linux ahora proporciona soporte a .NET, junto con aplicaciones .NET como la herramienta de búsqueda de escritorio Beagle y la herramienta de administración de fotografías F-Spot. Mono proporciona un entorno de desarrollo de fuente abierta para aplicaciones .NET. Mono

es un proyecto de fuente abierta al que da soporte Novell y que implementa .NET Framework en sistemas Unix, Linux y OS X. Actualmente se incluyen Mono 1.2 y 2.0. Mono 1.2 tiene correspondencia, por lo general, a las características de .NET 1.1 y Mono 2.0 a las de .NET 2.0. Consulte **mono-project.com** para conocer información detallada.

Mono se implementa en Linux mediante el uso de varios componentes. Éstos abarcan la aplicación básica .NET de Mono, que incluye herramientas de Mono como el administrador de certificados de Mono (certmgr); la herramienta Global Assemblies Cache Manager (gacutil) para que estén disponibles los ensamblados en tiempo de ejecución; y mcs, el compilador C# de Mono. Existen varias herramientas adicionales para diferentes características, como el soporte a Visual Basic, consultas de base de datos SQL y soporte Web a .NET. También se incluye la herramienta de prueba de lenguaje Mono, NUnit.

La configuración se encuentra en el archivo **/etc/mono/config**, que es un archivo tipo XML que correlaciona referencias DLL con bibliotecas de Linux. El archivo **/etc/mono** también contiene archivos de configuración para Mono 1.0 y 2.0. Mono se instala en **/usr/lib/mono**. En los directorios **1.0** y **2.0** correspondientes encontrará los ensamblados de soporte .NET DLL y EXE para diferentes aplicaciones de Mono. Otros directorios almacenarán DLL y configuración de .NET para diferentes aplicaciones y servicios, incluidos evolution, dbus y gtk.

La información de configuración local y aplicaciones en tiempo de ejecución se colocan en el directorio **.config** del usuario.

RAID y LVM

Con la llegada de los discos duros baratos, eficientes y muy grandes, incluso sistemas caseros simples pueden emplear varios discos duros. El uso de varios discos duros abre oportunidades para asegurar la confiabilidad del almacenamiento, además de un acceso organizado más sencillo a sus discos duros. Linux proporciona dos métodos para una mejor administración de sus discos: administrador de volumen lógico (LVM, Logical Volume Management) y los conjuntos redundantes de discos independientes (RAID, Redundant Arrays of Independent Disks). LVM es un método para organizar todos sus discos duros en volúmenes lógicos, permitiéndole conjuntar las capacidades de almacenamiento de varios discos duros en un volumen lógico. Así, su sistema ve un solo dispositivo grande y no tiene que microadministrar cada disco duro básico y sus particiones. LVM es, quizás, la manera más eficiente de agregar discos duros a su sistema, creando un conjunto grande y accesible de almacenamiento. RAID es una manera de almacenar los mismos datos en diferentes lugares de varios discos duros. Todos estos discos duros se tratan como uno solo. Incluyen información de recuperación que le permite restaurar sus archivos si uno de sus discos falla. Ambos sistemas pueden mezclarse, al implementar volúmenes LVM en un conjunto RAID. LVM proporciona flexibilidad y RAID protección de datos.

Con LVM ya no tiene que dar seguimiento a varios discos con sus particiones, al tratar de recordar dónde están almacenados sus archivos y en qué particiones del disco. Las particiones y sus discos se combinan en un sistema de archivos lógico que se adjunta al árbol de directorio de su sistema. Puede tener varios sistemas de archivos lógicos, cada uno con sus propios discos, particiones, o ambos

En un sistema con varios discos duros con LVM y RAID puede combinar los discos duros en un sistema de archivos lógico que accede al almacenamiento como un conjunto grande. Los archivos se almacenan en una sola estructura de directorio, no en directorios de una determinada partición. En lugar de montar los sistemas de archivos para cada disco duro individual, sólo existe un sistema de archivos para montar todos los discos duros. LVM tiene la ventaja agregada de permitirle implementar varios sistemas de archivos lógicos en diferentes particiones a través de varios discos duros.

RAID se ajusta mejor para escritorios y servidores que tienen varios discos duros y requieren recuperación de datos. La forma de RAID preferida, RAID 5, requiere un mínimo de tres discos duros. RAID, con la excepción de RAID 0, proporciona mejor protección contra fallas de disco duro y se considera una necesidad para tareas de almacenamiento intenso como operaciones de empresas, bases de datos y servidores de Internet. También proporciona tranquilidad para

pequeñas operaciones, al ofrecer recuperación de una falla de disco duro. Tenga en cuenta que existen varias formas de RAID, cada una con ventajas y desventajas. RAID 0 no proporciona capacidad de recuperación. Después de configurar un conjunto RAID, puede implementar volúmenes LVM en el conjunto.

En comparación con LVM, RAID proporciona un acceso más rápido para aplicaciones que trabajan con archivos muy grandes, como multimedia, bases de datos o aplicaciones gráficas. Pero para operaciones normales, LVM es tan eficiente como RAID. Sin embargo, LVM requiere que se ejecute su sistema Linux y se configure para su sistema operativo Linux. RAID, que ahora tiene soporte en el nivel de hardware en casi todos los equipos de cómputo, es mucho más fácil de configurar, sobre todo en el caso de una operación simple RAID 0 que combina los discos duros en una unidad.

Administrador de volumen lógico (LVM)

Para una administración más sencilla de almacenamiento de disco duro, se configura su sistema para utilizar el administrador de volumen lógico (LVM), al crear particiones LVM que se organizan en volúmenes lógicos a los que se asigna automáticamente el espacio libre. Los volúmenes lógicos proporcionan una forma mucho más flexible y poderosa de tratar con el almacenamiento en disco, al organizar las particiones físicas en volúmenes lógicos en los que se administra de manera sencilla el espacio en disco. El almacenamiento en disco para volúmenes lógicos se trata como un almacén de memoria, aunque el volumen puede, en realidad, contener varias particiones de disco duro en diferentes discos. Al agregar una nueva partición LVM simplemente se incrementa el almacén al que accede todo el sistema. El paquete LVM original fue desarrollado para el kernel 2.4. El paquete LVM2 actualmente se utiliza para el kernel 2.6. Revise el HOWTO de LVM en tldp.org para conocer ejemplos detallados.

NOTA Muchas distribuciones incluyen la herramienta **system-config-lvm** de Red Hat. Con esta herramienta se manejan de manera sencilla sus dispositivos LVM con una interfaz GUI simple.

La estructura LVM

En una estructura LVM, las particiones físicas de LVM, también conocidas como *extensiones*, se organizan en grupos lógicos, las que, a su vez, son utilizadas por volúmenes lógicos. En efecto, está tratando con tres diferentes niveles de organización. En los niveles más bajos, se tienen volúmenes físicos. Se trata de particiones físicas de disco duro que se crean con herramientas de creación de particiones como **parted** o **fdisk**. El tipo de partición será una LVM de Linux, código **8e**. Estos volúmenes físicos se organizan en grupos lógicos, conocidos como grupos de volumen, que operan casi de la misma forma que los discos duros lógicos. Se asignan colecciones de volúmenes físicos a diferentes grupos lógicos.

Una vez que tenga sus grupos lógicos, puede crear volúmenes lógicos. Los volúmenes lógicos funcionan en gran medida como particiones de disco duro en una configuración estándar. Por ejemplo, en el volumen de grupo **tortuga**, puede crear un volumen lógico **/var**, y en el grupo lógico **conejo**, puede crear los volúmenes lógicos **/home** y **/proyectos**. Se tienen varios volúmenes lógicos en un grupo lógico, así como se tienen varias particiones en un disco duro.

Los volúmenes lógicos se tratan como cualquier partición de disco duro ordinario. Se crea un sistema de archivos en uno con el comando **mkfs**, y después se monta el sistema de archivos que habrá de utilizarse con el comando **mount**. El tipo de sistema de archivos de Linux puede ser **ext3**.

Los volúmenes lógicos se administran con lo que se conoce como extensiones. Un grupo lógico define un tamaño estándar para una extensión, digamos 4 MB, y después divide cada volumen

físico en su grupo en extensiones de ese tamaño. Los volúmenes lógicos, a su vez, se dividen en extensiones del mismo tamaño que después se correlacionan con ellos en los volúmenes físicos.

Los volúmenes lógicos son lineales, en franjas o de espejo. La opción de espejo creará una copia de espejo de un volumen lógico, que proporciona capacidades de restauración. La opción de franjas le permite distribuir automáticamente su volumen lógico entre varias particiones, como lo haría un dispositivo RAID. Esto agrega eficiencia a archivos muy grandes, pero es difícil implementarla. Al igual que con un dispositivo RAID, los tamaños de las franjas tienen que ser consistentes entre las particiones y, como las particiones de LVM son de cualquier tamaño, los tamaños de las franjas tienen que calcularse con cuidado. El método más simple es utilizar una implementación lineal, como un dispositivo RAID 0, que trata al almacenamiento como un disco ordinario grande, con acceso secuencial al almacenamiento.

Existe una restricción y recomendación para los volúmenes lógicos. La partición de arranque no puede ser parte de un volumen lógico. Todavía tiene que crear una partición de disco duro separada como partición de arranque, con el punto de montaje **/boot**, en el que se instalan su kernel y todos los archivos de arranque. Además, se recomienda que no coloque su partición raíz en un volumen lógico. De hacerlo, complicaría cualquier recuperación de datos que se necesite. Por esto es por lo que una configuración de partición predeterminada para muchas distribuciones, que se configura durante la instalación, incluirá una partición **/boot** de 100 megas separada de tipo **ext3**, donde las particiones raíz y de intercambio se instalarán en volúmenes lógicos. Existirán dos particiones, una para el grupo lógico (el volumen físico LVM, **pv**) que almacena volúmenes de intercambio y raíz, y otra para la partición de arranque (**ext3**). A su vez, los volúmenes lógicos serán sistemas de archivos **ext3**.

Creación de LVM durante la instalación

Muchas distribuciones también le permiten crear LVM durante la instalación. Incluso, LVM puede ser la configuración predeterminada. Primero, se crean particiones LVM físicas, después se crean los grupos de volumen donde puso estas particiones y luego, a partir de los grupos de volúmenes, se crean los volúmenes lógicos, para los cuales se especifican después los puntos de montajes y los tipos de sistemas de archivos. Se crean particiones LVM durante el proceso de instalación. Cree una partición física LVM en su disco duro. Una vez hecho esto, cree sus volúmenes lógicos. Primero necesita asignar las particiones físicas LVM a grupos de volúmenes, que son, en esencia, discos duros lógicos. Una vez que se han creado los grupos de volúmenes, está listo para crear sus volúmenes lógicos. Puede crear varios volúmenes lógicos dentro de cada grupo. Los volúmenes lógicos funcionan como particiones. Tendrá que especificar un tipo de sistema de archivos y un punto de montaje para cada volumen lógico que cree, y necesita por lo menos un volumen de intercambio y raíz. El tipo de sistema de archivos para el volumen de intercambio es **swap**, para el volumen raíz es un tipo de sistema de archivos de Linux estándar como **ext3**.

Herramientas de configuración de distribución

Muchas distribuciones tienen sus propias herramientas de configuración de LVM que proporcionan una interfaz GUI fácil de usar para administrar sus sistemas de archivos de LVM. Red Hat y Fedora utilizan **system-config-lvm**. Estas herramientas proporcionan ayuda de contexto detallada, además de tutoriales y manuales en línea. Acceden a los comandos LVM descritos en este capítulo para realizar sus tareas. Por lo general, es preferible utilizar las herramientas de configuración LVM de la distribución para administrar sus sistemas de archivos LVM que intentan utilizar comandos LVM.

Herramientas de LVM: uso de comandos LVM

Se utiliza una colección de herramientas LVM para administrar sus volúmenes LVM, agregando nuevas particiones físicas de LVM y eliminando las actuales (revise la tabla 30-1). Las herramientas

Comando	Descripción
lvm	Abre una shell interactiva para ejecutar comandos LVM
lvmdiskscan	Revisa todos los discos para buscar particiones físicas LVM
lvdisplay	Despliega información detallada acerca de volúmenes lógicos
lvcreate	Crea volúmenes lógicos
lvrename	Cambia el nombre de un volumen lógico
lvchange	Modifica un volumen lógico
lvextend	Extiende el tamaño de un volumen lógico
lvreduce	Reduce el tamaño de un volumen lógico
lvremove	Elimina un volumen lógico
lvs	Presenta una lista de volúmenes lógicos con información detallada
lvresize	Cambia el tamaño de un volumen lógico
lvscan	Revisa el sistema para buscar volúmenes lógicos
pvdisplay	Despliega información detallada acerca de una partición física LVM
pvchange	Modifica una partición física LVM
pvcreate	Crea particiones físicas LVM
pvmove	Mueve el contenido de una partición física LVM a otra partición
pvremove	Elimina particiones físicas LVM
pvs	Presenta una lista de particiones físicas con información detallada
pvresize	Cambia el tamaño de una partición física
pvscan	Revisa el sistema para buscar particiones físicas
vgdisplay	Despliega información detallada acerca de grupos de volúmenes
vgexport	Activa grupos de volúmenes
vgimport	Hace que un nuevo sistema reconozca un grupo de volumen exportado. Es útil para mover un grupo de volumen activado de un sistema a otro
vgmerge	Combina grupos de volúmenes
vgreduce	Elimina particiones físicas de un grupo de volumen
vgremove	Elimina un grupo de volumen
vgs	Presenta una lista de grupos de volúmenes con información detallada
vgslit	Divide un grupo de volumen
vgscan	Revisa el sistema para buscar grupos de volúmenes
vgck	Revisa los grupos de volúmenes
vgrename	Cambia el nombre de un grupo de volumen
vgcfsbackup	Crea una copia de seguridad de la configuración de grupo de volumen (metadatos)
vgcfsrestore	Restaura la configuración de grupo de volumen (metadatos)

TABLA 30-1 Comandos de LVM

GUI de LVM de la distribución, como `system.config-lvm`, realmente son interfaces GUI para las herramientas LVM. Puede utilizar las herramientas LVD directamente o el comando `lvm` para generar una shell interactiva en la cual se ejecutan comandos LVM. Existen páginas Man para todos los comandos de LVM. LVM mantiene información de configuración en el archivo `/etc/lvm/lvm.conf`, donde se configuran opciones LVM como el archivo de registro, el directorio de copia de seguridad de la configuración o el directorio para dispositivos LVM (consulte la página Man `lvm.conf` para conocer más detalles).

Despliegue de información LVM

Se utilizan los comandos `pvdisplay`, `vgdisplay` y `lvdisplay` para mostrar información detallada acerca de una partición física, grupos de volúmenes y volúmenes lógicos. `pvscan`, `vgscan` y `lvscan` presentan una lista de sus volúmenes físicos, de grupo y lógicos.

Administración de volúmenes físicos LVM con comandos de LVM

Un volumen físico es cualquier partición de disco o dispositivo RAID. Un dispositivo RAID se ve como un solo volumen físico. Se crean volúmenes físicos ya sea de un solo disco duro o de particiones en él. En cada sistema grande, con muchos discos duros, lo más probable es que quiera usar un disco completo para cada volumen físico.

Para inicializar un volumen físico en un disco duro completo, se utiliza el nombre de dispositivo del disco duro, como se muestra aquí:

```
pvcreate /dev/sdc
```

Si está utilizando una sola partición para un disco entero, se crea un nuevo volumen físico al utilizar el nombre de dispositivo de la partición, como se muestra aquí:

```
pvcreate /dev/sdc1
```

Esto creará una partición física, pv, llamada `sdc1` en el disco duro `sdc` (el tercer disco Serial ATA, unidad c).

Para inicializar varios discos, sólo inclúyalos. Lo siguiente crea dos particiones físicas, `sdc1` y `sdd1`.

```
pvcreate /dev/sdc1 /dev/sdd1
```

También se utilizan varias particiones en diferentes discos duros. Ésta es una situación en que sus discos duros almacenan varias particiones. Esta condición ocurre a menudo cuando utiliza algunas particiones en su disco duro para diferentes propósitos como sistemas operativos diferentes, o si quiere distribuir su grupo lógico entre varios discos duros. Para inicializar estas particiones juntas, sólo inclúyalas.

```
pvcreate /dev/hda3 /dev/hdb1 /dev/hdb2
```

Una vez que ha inicializado sus particiones, tiene que crear grupos LVM en ellas.

Administración de grupos LVM

Las particiones LVM físicas se utilizan para conformar un grupo de volúmenes. Puede crear manualmente un grupo de volúmenes al utilizar el comando `vgcreate` y el nombre del grupo junto a una lista de particiones físicas que se quieren en el grupo.

Si está creando un nuevo grupo de volúmenes para incluir estas particiones, puede incluirlas en el grupo cuando crea el grupo de volúmenes con el comando `vgcreate`. Este grupo usa una o más particiones físicas. La configuración de instalación predeterminada que se describió antes usó sólo

una partición física para **VolGroup00**. En el siguiente ejemplo, un grupo de volúmenes llamado **mismedios** está integrado por dos volúmenes físicos, **sdb1** y **sdcl**.

```
vgcreate mismedios /dev/sdb1 /dev/sdcl
```

En el ejemplo anterior se configura un grupo lógico en dos discos duros Serial ATA, cada uno con su propia partición. Como opción, se configura un grupo de volumen para abarcar las particiones en varios discos. Si está utilizando particiones para diferentes funciones, este método le da la flexibilidad de utilizar todo el espacio disponible a través de varios discos duros. En el siguiente ejemplo se crea un grupo llamado **conejo** integrado por tres particiones físicas, **/dev/hda3**, **/dev/hdb2** y **/dev/hdb4**.

```
vgcreate conejo /dev/hda3 /dev/hdb2 /dev/hdb4
```

Si después quiere agregar un volumen físico a un grupo de volúmenes, utilice el comando **vgextend**. El comando **vgextend** agrega una nueva partición a un grupo lógico. En el siguiente ejemplo, la partición **/dev/hda3** se agrega al grupo de volúmenes **conejo**. En realidad, está extendiendo el tamaño de un grupo al agregar una nueva partición física.

```
vgextend conejo /dev/hda3
```

Para agregar un nuevo disco a un grupo de volúmenes, se sigue un procedimiento similar. En el siguiente ejemplo se agrega un quinto disco duro Serial ATA, **sde**, al crear primero un volumen físico en éste y después agregar ese volumen, **sde1**, al grupo de volúmenes **mismedios**.

```
pvcreate /dev/sde1
vgextend mismedios /dev/sde1
```

Para eliminar una partición física, primero elimínala de su grupo lógico. Tal vez tenga que utilizar el comando **pvmove** para mover cualquier dato fuera de la partición física. Luego se utiliza el comando **vgreduce** para eliminar la partición del grupo lógico.

A cambio, se elimina un grupo de volúmenes completo al desactivarlo primero con **vgchange -a n** y después utilizar el comando **vgremove**.

Activación de grupos de volúmenes

Mientras en una estructura de sistema de archivos estándar, se montan y desmontan particiones de disco duro, con una estructura LVM se activan y desactivan grupos de volúmenes completos. Los volúmenes de grupo están inaccesibles hasta que los activa con el comando **vgchange** con la opción **-a**. Para activar un grupo, primero reinicie su sistema y después inserte el comando **vgchange** con la opción **-a** y el argumento **y** para activar el grupo lógico (un argumento **n** desactivará el grupo).

```
vgchange -a y conejo
```

Administración de volúmenes lógicos LVM

Para crear volúmenes lógicos, se utiliza el comando **lvcreate** y después formatea su volumen lógico con un comando estándar como **mkfs.ext3**.

Con la opción **-n** se especifica el nombre del volumen, que funciona como una etiqueta de la partición del disco duro. Se utiliza la opción **-L** para especificar el tamaño del volumen. Hay otras opciones para implementar características como si se estableciera un volumen lineal, de franjas o de espejo, o para especificar el tamaño de las extensiones que habrán de usarse. Por lo general, las opciones predeterminadas funcionan bien. En el siguiente ejemplo se crea un volumen lógico llamado **proyectos** en el grupo lógico **conejo** con un tamaño de 20 GB.

```
lvcreate -n proyectos -L 20GB conejo
```

En el siguiente ejemplo se configura un volumen lógico en el grupo de volúmenes **mismedios**, con un tamaño de 540 GB. El grupo de volúmenes **mismedios** está integrado por dos volúmenes físicos, cada uno con discos duros de 320 GB. En efecto, los dos discos duros se ven lógicamente como uno.

```
lvcreate -n misvideos -L 540GB mismedios
```

Una vez que ha creado su volumen lógico, tiene que crear un sistema de archivos para usarlo. Aquí se crea un sistema de archivos **ext3** en el volumen lógico **misvideos**.

```
mkfs.ext3 misvideos
```

Se elimina un volumen lógico con el comando **lvremove**. Con **lvextend**, se incrementa el tamaño de un volumen lógico y con **lvreduce** se reducirá su tamaño.

Nombres de dispositivos LVM: /dev/mapper

LVM utiliza la unidad **device-mapper** para configurar tablas para asignación de dispositivos locales a un disco duro. El nombre de dispositivo para un volumen lógico se almacena en el directorio **/dev/mapper** y tiene el formato *grupo lógico-volumen lógico*. Además, habrá una carpeta de dispositivo correspondiente para el grupo, que contendrá nombres de volumen lógico. Estos nombres de dispositivos son vínculos con los nombres **/dev/mapper**. Por ejemplo, el volumen lógico **misimagenes** en el grupo lógico **mismedios** tiene el nombre de dispositivo, **/dev/mapper/mismedios-misimagenes**. Habrá una carpeta correspondiente llamada **/dev/mismedios**. El nombre de dispositivo **/dev/mismedios/misimagenes** es un vínculo con el nombre de dispositivo **/dev/mapper/mismedios-misimagenes**. Puede utilizar de la misma forma sencilla el vínculo, como se muestra en este capítulo, como el nombre de dispositivo original. El dispositivo de creación de instantáneas que se describe más adelante, en este capítulo, tendría el nombre de dispositivo **/dev/mapper/mismedios-misimagenessnapshot1**, con el vínculo de nombre de dispositivo **/dev/mismedios/misimagenessnapshot**.

NOTA Tiene la opción de crear copias de seguridad de los metadatos del grupo de volúmenes (configuración) al utilizar el comando **vgcfgbackup**. Esto no copia sus volúmenes lógicos (sin contenido). Las copias de seguridad de los metadatos se almacenan en **etc/lvm/backup** y se restauran al utilizar **vgcfgrestore**.

Ejemplo de LVM para varios discos duros

Ahora que los discos duros se han vuelto más baratos y la demanda de almacenamiento ha incrementado, muchos sistemas utilizan varios discos duros. Para administrarlos, las particiones de cada uno solían administrarse individualmente, a menos que implementara un sistema RAID. RAID le permite tratar varios discos duros como un dispositivo de almacenamiento, pero hay restricciones en el tamaño y los tipos de dispositivos que se combinan. Sin RAID, cada disco duro tiene que administrarse de forma separada, y los archivos tienen que caber en el espacio restante a medida que se llenan los discos.

Con LVM, ya no se tienen estas restricciones. Puede combinar discos duros en un solo dispositivo de almacenamiento. Este método también es flexible, permitiéndole reemplazar discos sin perder datos, además de agregar nuevos discos para incrementar automáticamente su espacio almacenamiento (o reemplazar discos duros con otros más grandes).

Por ejemplo, digamos que quiere agregar dos discos duros a su sistema, pero quiere tratar el almacenamiento en ambos de forma lógica en vez de tener que administrar particiones en cada uno. LVM le permite tratar el espacio de almacenamiento combinado de ambos discos duros como un almacén gigante. En efecto, dos discos de 500 GB se tratan como un dispositivo de almacenamiento de 1 terabyte.

En este ejemplo, el sistema Linux hace uso de tres discos duros. El sistema Linux y las particiones de arranque están en el primer disco duro, **sda**. Se agregan dos discos duros a este sistema, **sdb** y **sd**c, que integrarán un dispositivo de almacenamiento LVM que habrá de agregarse al sistema.

Los pasos necesarios para crear y acceder a volúmenes lógicos se describen en los siguientes comandos. En este ejemplo hay dos discos duros que se combinarán en una unidad LVM. Los discos duros son Serial ATA que se identifican en el sistema como **sdb** y **sd**c. Cada disco se particiona primero con una sola partición física LVM. Se utiliza una herramienta de creación de partición como **fdisk** o **parted** para crear las particiones físicas en los discos duros **sdb** y **sd**c. En este ejemplo, se crean las particiones **sdb1** y **sd**c1.

Primero debe inicializar los volúmenes físicos con el comando **pvcreate**. Las particiones **sda1** y **sda2** de la entrada **sda** se reservan para las particiones de arranque y raíz y nunca se inicializan.

```
pvcreate /dev/sda1 /dev/sdb1 /dev/sdc1
```

Luego se crean los grupos lógicos que quiere al utilizar el comando **vgcreate**. En este caso existe un grupo lógico, **mismedios**, que utiliza **sdb1** y **sd**c1. Si después crea un volumen físico y quiere agregarlo al grupo de volúmenes, se utiliza el comando **vgextend**.

```
vgcreate mismedios /dev/sdb1 /dev/sdc1
```

Ahora se crean los volúmenes lógicos en cada grupo de volúmenes, al utilizar el comando **lvcreate**. En este ejemplo, se crean dos volúmenes lógicos, uno para **misvideos** y otro para **misimagenes**. Aquí se muestran los comandos correspondientes **lvcreate**:

```
lvcreate -n misvideos -l 540GB mismedios
lvcreate -n misimagenes -l 60GB mismedios
```

Entonces se activan los volúmenes lógicos. Reinicie y utilice **vgchange** con la opción **-a** y para activar los volúmenes lógicos.

```
vgchange -a y mismedios
```

Ahora se crean sistemas de archivos para cada volumen lógico.

```
mkfs.ext3 misvideos
mkfs.ext3 misimagenes
```

Entonces se montan los volúmenes lógicos. En este ejemplo se montan en subdirectorios del mismo nombre en **/misdatos**.

```
mount -t ext3 /dev/mismedios/misimagenes /misdatos/misimagenes
mount -t ext3 /dev/mismedios/misvideos /misdatos/misvideos
```

Uso de LVM para reemplazar discos

LVM es muy útil cuando necesita reemplazar un disco duro viejo con uno nuevo. Los discos duros se espera que duren cerca de seis años en promedio. Puede que también quiera sólo reemplazar el disco

duro viejo con uno más grande (los tamaños de almacenamiento de disco duro se duplican cada año más o menos). Remplazar un disco duro on-boot es muy sencillo. Remplazar un disco de arranque es mucho más complicado.

Para remplazar un disco, sólo incorpore la nueva unidad en su volumen lógico. El tamaño de su volumen lógico incrementará de manera correspondiente. Se utiliza el comando **pmove** para mover datos de un disco viejo a uno nuevo. Después, se usan comandos para eliminar el viejo. Desde el punto de vista del usuario y el sistema no habrá cambios. Los archivos de su disco viejo se almacenarán en los mismos directorios, aunque el almacenamiento actual se implementará en el nuevo disco.

El reemplazo con LVM se vuelve más complicado, si quiere remplazar su disco de arranque, el disco duro con el que su sistema inicia y que almacena su kernel de Linux. El disco de arranque contiene una partición de arranque especial y el registro de arranque maestro. La partición de arranque no puede ser parte de ningún volumen LVM. Primero, tiene que crear una partición de arranque en el nuevo disco duro, al utilizar una herramienta de partición como **parted** o **fdisk**, que se etiqueta como boot. El disco de arranque suele ser muy pequeño, como 200 MB. Luego se monta la partición en su sistema y se copia el contenido de su directorio **/boot** en ésta. Después se agrega el resto del disco a su volumen lógico y se elimina de forma lógica el disco viejo, al copiar el contenido de ese disco en el nuevo. Todavía tiene que arrancar con el DVD de rescate de Linux (o instalar el DVD en modo de rescate) y enviar el comando **grub-install** para instalar el registro de arranque maestro en su nuevo disco. Entonces podrá arrancar desde el nuevo disco.

Ejemplo de LVM para particiones en diferentes discos duros

En una implementación más compleja, puede utilizar particiones en diferentes discos duros para los mismos volúmenes lógicos. Por ejemplo, si quiere tener volúmenes físicos que constan de particiones de disco duro **hda2**, **hda3**, **hdb1**, **hdb2** y **hdb3** en dos discos duros, **hda** y **hdb**, puede asignar algunas de éstas a un grupo lógico y unas más a otro. Las particiones que integran los diferentes grupos locales pueden provenir de diferentes discos duros físicos. Por ejemplo, **hda2** y **hda3** pueden pertenecer al grupo lógico **tortuga** y **hda3**, **hdb2** y **hdb3** pueden integrar un grupo lógico diferente, digamos **conejo**. El nombre de grupo lógico puede ser cualquier nombre que quiera darle. Es casi igual que asignar un nombre a un disco duro.

NOTA Los ejemplos que se muestran aquí utilizan el prefijo **hd** para hacer referencia a discos duros. Algunas distribuciones, como Fedora, han dejado el prefijo **hd** y utilizan **sd** para discos duros ATA y Serial ATA.

Al utilizar el ejemplo en la figura 30-1, los pasos asociados con la creación y el acceso a volúmenes lógicos se describen con los siguientes comandos. Primero, se utiliza una herramienta de creación de particiones como **fdisk** o **parted** para crear particiones físicas en los discos duros **hda** y **hdb**. En este ejemplo, se crean las particiones **hda1**, **hda2**, **hda3**, **hdb1**, **hdb2**, **hdb3** y **hdb4**.

Enseguida, se inicializan los volúmenes físicos con el comando **pvccreate**. Las particiones **hda1** y **hda2** se reservan para las particiones de arranque y raíz y no se inicializan.

```
pvccreate /dev/hda3 /dev/hdb1 /dev/hdb2
pvccreate /dev/hdb3 /dev/hdb4
```

Luego puede crear los volúmenes lógicos que quiera al utilizar el comando **vgcreate**. En este caso existen dos grupos, **tortuga** y **conejo**. El grupo **tortuga** utiliza **hdb1** y **hdb3**, y **conejo** utiliza **hda3**, **hdb2** y **hdb4**. Si después crea un volumen físico y quiere agregarlo al grupo de volúmenes, utilice el comando **vgextend**.

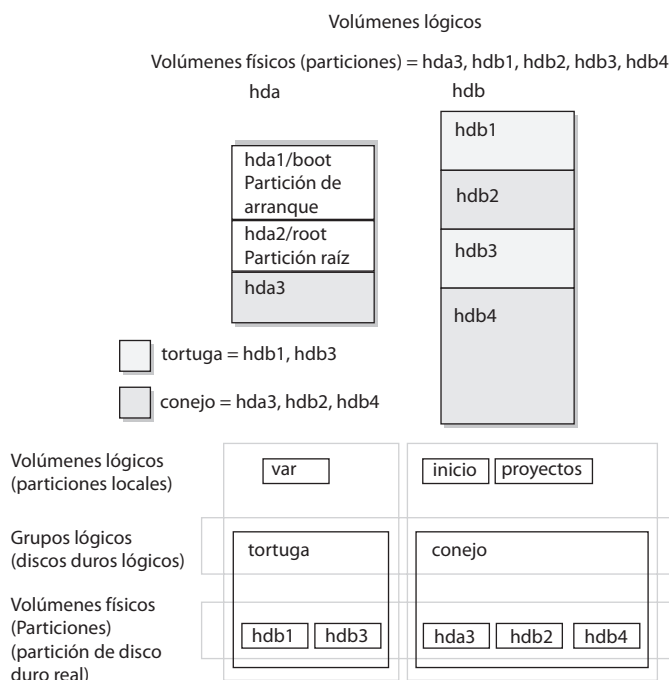


FIGURA 30-1 Volumen lógico que utiliza varias particiones en diferentes discos duros

```
vgcreate tortuga /dev/hdb1 /dev/hdb3
vgcreate conejo /dev/hda3 /dev/hdb2 /dev/hdb4
```

Ahora se crean volúmenes lógicos en cada grupo de volúmenes, al utilizar el comando **lvcreate**.

```
lvcreate -n var -l 2000M tortuga
lvcreate -n inicio -l 50000M conejo
lvcreate -n proyectos -l 20000M conejo
```

Después se activan los volúmenes lógicos. Reinicie y utilice **vgchange** con la opción **-a y** para activar los volúmenes lógicos.

```
vgchange -a y tortuga conejo
```

Ahora se crean sistemas de archivos para cada volumen lógico.

```
mkfs.ext3 var
mkfs.ext3 home
mkfs.ext3 projects
```

Luego se montan los volúmenes lógicos.

```
mount -t ext3 /dev/tortuga/var /var
mount -t ext3 /dev/conejo/home /home
mount -t ext3 /dev/conejo/proyectos /mnt/misproyectos
```

Instantáneas de LVM

Una instantánea registra el estado del volumen lógico en un momento designado. No crea una copia completa de datos en su volumen, sólo cambia desde la última instantánea. Una instantánea define el estado de los datos en un momento dado. Esto le permite respaldar los datos de forma consistente. Además, si necesita restaurar un archivo a su versión previa, se utiliza su instantánea. Las instancias se tratan como volúmenes lógicos y se montan, copian o eliminan.

Para crear una instantánea se utiliza el comando **lvcreate** con la opción **-s**. En este ejemplo a la instantánea se le asigna el nombre **misimagenes-snap1** (opción **-n**). Necesita especificar el nombre de dispositivo completo para el grupo lógico que quiere crear para la instantánea. Asegúrese de que hay espacio libre suficiente en el grupo lógico para la instantánea. En este ejemplo, el volumen lógico de la instantánea se crea en el grupo lógico **/dev/mismedios**. Se puede crear con la misma facilidad en cualquier otro grupo lógico. Aunque una instantánea suele utilizar muy poco espacio, tiene que protegerse contra desbordamientos. Si la instantánea tiene el mismo espacio asignado que el original, nunca se desbordará. En el caso de sistemas en que cambian pocos datos originales, la instantánea es muy pequeña. En el siguiente ejemplo se asigna una tercera parte del tamaño original (60 GB).

```
lv create -s -n misimagenes-snap1 -l 20GB /dev/mismedios
```

Entonces puede montar la instantánea como lo haría con cualquier otro sistema de archivos.

```
mount /dev/mismedios/misimagenessnap1 /missnaps
```

Para eliminar una instantánea se utiliza el comando **lvremove**, como se haría con cualquier volumen lógico.

```
lvremove -f /dev/mismedios/misimagenessnap1
```

Las instantáneas son muy útiles para crear copias de seguridad, mientras un sistema todavía está activo. Se utiliza **tar** o **dump** para crear una copia de seguridad de la montada en un disco o cinta. Todos los datos del volumen original se incluirán, junto con cambios que percibirá la instantánea.

Las instantáneas también le permiten realizar operaciones de deshacer efectivas. Se crea una instantánea de un volumen lógico. Luego se desmonta el original y se monta la instantánea en su lugar. Cualquier cambio que haga se aplicará en la instantánea, no en el original. Si ocurren problemas, desmonte la instantánea y monte el original. Esto restaura el estado original de sus datos. También se hace esto al utilizar varias instantáneas, restaurando una anterior. Con este procedimiento, se puede probar nuevo software en una instantánea, sin tener que poner en peligro sus datos originales. El software operaría en la instantánea, no en el volumen lógico original.

También se utilizan como versiones alternas de un volumen lógico. Tiene la opción de leer y escribir en una instantánea. Una escritura cambiará sólo el volumen de la instantánea, no el original, al crear, en efecto, una versión alterna.

Configuración de dispositivos RAID

RAID es un método de almacenamiento de datos a través de varios discos para proporcionar mucho mejor rendimiento y redundancia. En realidad, se tienen varios discos duros a los que el sistema operativo trata como uno solo. Entonces, RAID almacena y recupera datos de forma eficiente a través de los tres discos, en vez de que el sistema operativo acceda a cada uno como un sistema de archivos separado. Los detalles de nivel más bajo relacionados con el almacenamiento y la recuperación ya no son preocupaciones para el disco duro. Esto permite mucha mayor flexibilidad

en la adición y eliminación de discos duros, además de la implementación de redundancia en el sistema de almacenamiento para proporcionar una mayor confiabilidad. Con RAID, se tienen varios discos duros que se tratan como un disco virtual, donde varios de los discos se utilizan como espejos en tiempo real, que duplican datos. Se utiliza RAID de varias formas, dependiendo del grado de confiabilidad que necesite. Cuando coloca datos en varios discos, las operaciones de entrada/salida se superponen en una forma balanceada, mejorando el rendimiento. Debido a que el uso de varios discos incrementa el tiempo medio entre fallas (MTBF, Mean Time Between Failures), el almacenamiento redundante de datos también incrementa la falta de tolerancia.

RAID se implementa en el nivel de hardware o de software. En un nivel de hardware, se tienen discos duros conectados a un controlador de hardware RAID, por lo general una tarjeta de PC especial. Luego su sistema operativo accede al almacenamiento mediante un controlador de hardware RAID. Como opción, puede implementar RAID como controlador de software, permitiendo que un programa controlador de RAID de software administre el acceso a discos duros tratados como dispositivos RAID. La versión de software le permite utilizar discos duros IDE como RAID. Linux utiliza el controlador MD, compatible con el kernel 2.4, para implementar un controlador RAID de software. El software RAID de Linux permite cinco niveles (lineal, 0, 1, 4, 5 y 6), mientras el hardware RAID permite muchos más. Los niveles de RAID hardware, como los del 7 al 10, proporcionan combinaciones de mayor rendimiento y confiabilidad.

SUGERENCIA Antes de que se utilice RAID en su sistema, asegúrese de que tiene soporte en su kernel, junto con los niveles de RAID que quiere usar. Si no, tendrá que reconfigurar e instalar un módulo RAID para el kernel. Revise el componente Multi-Driver Support en su configuración de kernel. Se especifica el soporte para cualquier nivel de RAID, o para todos.

Soporte a RAID en tarjeta madre: dmraid

Con el kernel 2.6, los dispositivos RAID de hardware tienen soporte con la herramienta Device-Mapper Software (**dmraid**), que en realidad es compatible con un amplio rango de dispositivos RAID en tarjeta madre. Tenga en cuenta que muchos dispositivos RAID “hardware” son, en realidad, software RAID (fakeraid). Aunque se configuran en el BIOS de la tarjeta madre, los discos operan como software, como cualquier otro disco. En este aspecto se pueden considerar menos flexibles que una solución RAID de software de Linux, y también pueden depender directamente del soporte técnico del comercializador para cualquier parche o actualización.

La unidad **dmraid** asignará en su sistema los dispositivos RAID de hardware como los que proporcionan Intel, Promise y Silicon Magic, y a menudo se incluyen en tarjetas madre. La herramienta **dmraid** utiliza el controlador de asignación de dispositivos para configurar una interfaz de sistema de archivos virtual, como se hace con los discos LVM. Los nombres de dispositivo RAID se ubicarán en **/dev/mapper**.

Se utiliza su utilería de configuración de RAID del BIOS para configurar sus dispositivos RAID de acuerdo con su documentación del hardware. Durante una instalación de Linux, los dispositivos RAID se detectan automáticamente y el módulo **dmraid** se carga, al seleccionar los discos apropiados.

Con el comando **dmraid** se detectan y activan dispositivos RAID. El siguiente comando despliega sus dispositivos RAID:

```
dmraid -r
```

Para presentar una lista de dispositivos compatibles, se utiliza **dmraid** con la opción **-l**.

```
dmraid -l
```

La herramienta **dmraid** mejora continuamente y tal vez no trabaje bien con algunos dispositivos RAID.

Niveles de RAID de software en Linux

El software RAID de Linux se implementa en diferentes niveles, dependiendo de si quiere capacidad de organización, eficiencia, redundancia o reconstrucción. Cada capacidad corresponde a diferentes niveles RAID. Para casi todos los niveles, los tamaños de los dispositivos de disco duro deben ser los mismos. Para función de espejo con RAID 1, se requieren discos del mismo tamaño, y se recomiendan para RAID 5. El software RAID de Linux soporta cinco niveles, como se muestra en la tabla 30-2. RAID 5 requiere por lo menos tres discos duros.

Lineal

El nivel *lineal* le permite simplemente organizar varios discos duros en uno lógico, que proporciona un almacén continuo. En lugar de ser forzado para configurar particiones en cada disco duro, en realidad sólo tiene un disco duro. El almacenamiento se administra de forma secuencial. Cuando un disco duro se llena, se utiliza el siguiente. En efecto, está *adjuntando* un disco a otro. Este nivel no proporciona capacidad de recuperación. Si tiene un conjunto RAID con dos discos duros de 80 GB, después de que utilice el almacenamiento para uno, automáticamente iniciará en el siguiente.

RAID 0: franjas

Para mayor eficiencia, RAID almacena datos mediante la *división en franjas* de los discos, donde los datos se organizan en listas que se almacenan a través de discos RAID para acceso más rápido (nivel 0). RAID 0 también organiza sus discos duros en dispositivos RAID comunes pero los trata

Nivel de RAID	Capacidad	Descripción
Lineal	Anexión	Trata a los discos duros RAID como un disco virtual sin franjas, funciones de espejo o reconstrucción de paridad.
0	Franjas	Implementa la separación en franjas del disco a través de unidades de disco sin redundancia.
1	Espejo	Implementa un nivel alto de redundancia. Cada disco se trata como un espejo para todos los datos.
5	Paridad distribuida	Implementa la capacidad de reconstrucción de datos al utilizar información de paridad. Esta información se distribuye a través de todos los discos, en lugar de utilizar un disco separado como en RAID 4. Requiere al menos tres discos duros o particiones.
6	Paridad distribuida	Implementa la capacidad de reconstrucción de datos al utilizar información de paridad distribuida dual. Los conjuntos duales de información de paridad se distribuyen a través de todos los discos. Se considera una forma mejorada de 5.
Multirutas	Acceso múltiple a dispositivos	Permite acceso múltiple al mismo dispositivo.

TABLA 30-2 Niveles de software RAID de Linux

como discos duros individuales, al almacenar datos de forma aleatoria a través de todos los discos. Si tiene un conjunto RAID de disco duro que contiene dos discos de 80 GB, puede acceder a éstos como un solo dispositivo RAID de 160 GB.

RAID 1: espejo

El nivel de RAID 1 implementa redundancia mediante *función de espejo*. En ésta, los mismos datos se escriben en cada disco RAID. Cada disco tiene una copia completa de todos los datos escritos; de este modo, si uno o más discos fallan, los otros todavía tienen sus datos. Aunque es muy segura, la redundancia es muy ineficiente y consume una gran cantidad de almacenamiento. Por ejemplo, en un conjunto RAID de dos discos de 80 GB, un disco se utiliza para almacenamiento estándar y el otro es un respaldo en tiempo real. Esto lo deja con sólo 80 GB para utilizar en su sistema. Las operaciones de escritura también tienen que duplicarse en todos los discos duros de espejo usados por el conjunto RAID, haciendo más lentas las operaciones.

RAID 5 y 6: Paridad distribuida

Como una opción alterna a la función de espejo, los datos se reconstruyen al utilizar *información de paridad*, en caso de una falla de disco duro. La información de paridad se guarda en lugar de una duplicación completa de los datos. La información de paridad ocupa el espacio equivalente a un disco duro, que deja casi todo el espacio en los discos RAID libre para almacenamiento. RAID 5 combina franjas y paridad (consulte lo relacionado con RAID 4), donde la información de paridad se distribuye entre discos duros, en lugar de hacerse en un disco dedicado a ese propósito. Esto permite el uso de un método de acceso más eficiente, división en franjas. Con éste y paridad, RAID 5 proporciona un acceso más rápido y capacidades de recuperación, haciéndolo el nivel RAID más popular. Por ejemplo, un RAID de cuatro discos duros de 80 GB se trataría como un disco duro de 320 GB, con parte de ese almacenamiento (80 GB) usada para tener información de paridad, dejando libres 240 GB. RAID 5 requiere al menos tres discos duros.

RAID 6 opera igual que RAID 5, pero utiliza conjuntos duales de información de paridad para los datos, lo que proporciona una capacidad aún mayor de restauración.

RAID 4: Paridad

Aunque no tiene soporte en algunas distribuciones debido al costo de los recursos, RAID 4, al igual que RAID 5, es compatible con una forma más comprimida de recuperación, al utilizar información de paridad en lugar de datos de espejo. Con RAID 4, la información de paridad se almacena en un disco separado, mientras los otros se utilizan para almacenamiento de datos, de manera muy parecida al modelo lineal.

SUGERENCIA Muchas distribuciones también le permiten crear un formato de discos RAID durante la instalación. En ese momento, se crean sus particiones y dispositivos RAID.

Multirruta

Aunque realmente no es un nivel RAID, Multirruta permite acceso múltiple al mismo dispositivo. Si un controlador falla, otro se utiliza para acceder al dispositivo. En efecto, tiene redundancia en el nivel del controlador. El soporte se implementa al utilizar el daemon **mdadm**. Esto se inicia con la secuencia de comandos **mdadm**.

```
start mdadm start
```

Dispositivos y particiones RAID: md y fd

A un dispositivo RAID se le designa como **md**; utiliza la unidad MD. Estos dispositivos ya están definidos en su sistema Linux, en el directorio **/etc/dev**, a partir de **md0: /dev/md0** es el primer dispositivo RAID; **/dev/md1**, el segundo, etc. En cambio, cada dispositivo RAID utiliza particiones; cada una de ellas contiene un disco duro entero. A estas particiones suele conocerse como discos RAID, mientras que un dispositivo RAID es el conjunto de los discos RAID que utiliza.

Cuando se crea una partición RAID, debe configurar que el tipo de partición sea **fd**, en lugar del 83 de la partición de Linux estándar. El tipo **fd** es el que utiliza RAID para detección automática.

Arranque desde un dispositivo RAID

Por lo general, como parte del proceso de instalación, puede crear dispositivos RAID para arranque de su sistema Linux. Éste se configurará para cargar el soporte de kernel RAID y detectar automáticamente sus dispositivos RAID. El cargador de arranque se instalará en su dispositivo RAID, considerando que todos los discos duros integran ese dispositivo.

Casi ninguna distribución de Linux da soporte a arranque desde RAID 5; sólo RAID 1. Esto significa que si quiere utilizar RAID 5 y todavía arrancar de los discos RAID, necesitará crear al menos dos dispositivos RAID (o más, si quiere), con el uso de las particiones correspondientes para cada dispositivo a través de sus discos duros. Un dispositivo almacenaría su partición **/boot** y se instalaría como un dispositivo RAID 1. Éste sería el primer dispositivo RAID, **md0**, que consta de la primera partición en cada disco duro. Entonces, el segundo dispositivo, **md1**, puede ser un RAID 5 y estaría integrado por las particiones correspondientes en los demás discos duros. Así, su sistema puede arrancar desde un dispositivo RAID 1 pero usar RAID 5.

Si no crea discos RAID durante la instalación, sino que los crea después y quiere arrancar desde éstos, tendrá que asegurarse de que su sistema se configura correctamente. Es necesario crear los dispositivos RAID con superbloques persistentes y el soporte para los dispositivos RAID tiene que ser establecido en el kernel. En las distribuciones de Linux, este soporte se establece como un módulo. Surgen dificultades si está utilizando RAID 5 para su partición **/** (raíz). Esta partición contiene el módulo RAID 5, pero para acceder a la partición, ya tiene que tener cargado el módulo RAID 5. Para sortear esta limitación, puede crear un disco RAM en la partición **/boot** que contiene el módulo RAID 5. Utilice el comando **mkinitrd** para crear el disco RAM y la opción **-with** para especificar el módulo que habrá de incluirse.

```
mkinitrd --preload raid5 --with=raid5 raid-ramdisk 2.6.9-1
```

Administración de RAID: mdadm

Se utiliza la herramienta **mdadm** para administrar y monitorear dispositivos RAID. Se trata de un método general para crear, monitorear, administrar y arreglar dispositivos RAID. Se ejecutan comandos directamente para crear y dar formato a discos RAID. También se ejecuta como un daemon para monitorear y detectar problemas con los dispositivos.

La herramienta **mdadm** tiene siete modos diferentes de operación, cada uno con su propio conjunto de opciones, incluido el monitoreo con la opción **-f** para ejecutarlo como un daemon, o creación con **-l** para configurar un nivel RAID para un disco. En la tabla 30-3 se presentan diferentes modos de operación. Revise la página Man de **mdadm** para conocer una lista detallada de opciones para cada modo.

Modo	Descripción
assemble	Ensambla un conjunto RAID de dispositivos.
build	Genera un conjunto sin superbloques por dispositivo.
create	Genera un conjunto con superbloques por dispositivo.
manage	Administra dispositivos de conjunto, agregando o eliminando discos.
misc	Realiza operaciones específicas en un dispositivo, como hacerlo de sólo lectura.
monitor	Monitorea conjuntos para cambios, y actúa sobre éstos (se utiliza para RAID 1, 4, 5, 6).
grow	Cambia el tamaño del conjunto, como cuando se reemplazan dispositivos más pequeños con otros más grandes.

TABLA 30-3 Modos de **mdadm**

Creación e instalación de dispositivos RAID

Si creó sus dispositivos RAID y sus particiones durante el proceso de instalación, ya debe de tener dispositivos RAID trabajando. Sus dispositivos RAID se configurarán en el archivo **/etc/mdadm.conf**, y el estado de sus dispositivos RAID se encontrará en el archivo **/proc/mdstat**. Se inician o detienen dispositivos RAID con los comandos **raidstart** y **mdadm**. La opción **-a** opera en todos, aunque puede especificar dispositivos particulares, si así lo desea.

Si desea crear manualmente un nuevo dispositivo RAID para un sistema ya instalado, siga estos pasos:

- Asegúrese de que su kernel es compatible con el nivel RAID que quiere asignar al dispositivo que está creando.
- Si no lo ha hecho, cree los discos RAID (particiones) que utilizará para su dispositivo RAID.
- Cree su dispositivo RAID con el comando **mdadm** en el modo **build** o **create**. El conjunto también se activará.
- Como opción, puede configurar su dispositivo RAID (**/dev/mdn**) en el archivo **/etc/mdadm.conf**, que especifica los discos RAID que se utilizarán, y después emplear el comando **mdadm** que especifica el dispositivo RAID que se creará.
- Cree un sistema de archivos en el dispositivo RAID (**mkfs**) y después móntelo.

Adición de un sistema de archivos RAID separado

Si sólo quiere agregar un sistema de archivos RAID a un sistema que ya tiene una partición de arranque estándar, puede emplear la primera partición RAID 1. Si habrán de utilizarse tres discos duros en el sistema de archivos, sólo necesita tres particiones, una para cada disco, **sda1**, **sdb1** y **sdcl**.

```
ARRAY /dev/md0    devices=/dev/sda1, /dev/sdb1    level=1    num-devices=2
```

Luego se crea el conjunto con el siguiente comando:

```
mdadm -C /dev/md0 -n3 /dev/sda1 /dev/sdb1 /dev/sdc1 -l5
```

Entonces puede formar y montar su dispositivo RAID.

Creación de particiones de disco duro: fd

Para agregar nuevos dispositivos RAID, o para crearlos desde el principio, necesita crear de forma manual las particiones de disco duro que utilizarán y después configurar los dispositivos RAID para que utilicen esas particiones. Si desea crear una partición de disco duro para utilizar en un conjunto RAID, utilice **fdisk** o **parted** y especifique **fd** como el tipo de sistema de archivos. Usted invoca **fdisk** o **parted** con el nombre de dispositivo del disco duro donde quiere crear la partición. Asegúrese de especificar **fd** como tipo de partición. En el siguiente ejemplo se invoca a **fdisk** para el disco duro **/dev/hdc** (el primer disco duro en la conexión IDE secundaria):

```
fdisk /dev/hdc
```

Aunque técnicamente se trata de particiones, a estos dispositivos de disco duro se les conoce como discos en la documentación y los archivos de la configuración en RAID.

*NOTA También se utiliza **gparted** o **qtparted** para crear sus particiones de disco. Estas herramientas proporcionan una interfaz GUI para **parted** (menú Aplicaciones | Herramientas del sistema).*

Configuración de RAID: /etc/mdadm.conf

Una vez que ya cuenta con sus discos, debe configurarlos como dispositivos RAID. Éstos se configuran en el archivo **/etc/mdadm.conf**, con opciones como las mostradas en la tabla 30-4. Este archivo se utilizará para el comando **mdadm** en el modo create para crear el dispositivo

Directiva u opción	Descripción
DEVICE <i>lista-dispositivos</i>	Particiones y discos que se utilizan para dispositivos RAID.
ARRAY	Sección de configuración ARRAY para un dispositivo RAID particular.
level=num	El nivel del dispositivo RAID, como 0, 1, 4, 5 y -1 (lineal).
devices=lista-dispositivos-disco	Los dispositivos de disco (particiones) que hacen el conjunto RAID.
num-devices=cuenta	Enumera el número de dispositivos RAID en un conjunto. Cada sección debe tener su directiva. El máximo es 12.
spare-group=nombre	El nombre de texto de un grupo de reemplazo, cuyos dispositivos se utilizan para otros conjuntos.
auto=opción	Crea de forma automática dispositivos especificados, si no existen. Puede crear conjuntos no particionados tradicionales (opción yes o md) o particionables nuevos (opción mdp o part). En el caso de conjuntos particionables, la opción predeterminada es 4, que puede modificarse.
super-minor	Número mínimo de superbloques de conjunto, igual al número de dispositivo md.
uuid=UUID-número	Identificador UUID que se almacena en superbloques de conjuntos, utilizado para identificar el conjunto RAID. Se emplea para hacer referencia a un conjunto con comandos.
MAILADDR	Modo monitor, dirección de correo adonde se envían las alertas.
PROGRAM	Modo monitor, programa que se ejecutará cuando ocurran sucesos.

TABLA 30-4 Opciones de **mdadm.conf**

RAID. En el archivo `/etc/mdadm.conf`, se crean ambas entradas `DEVICE` y `ARRAY`. Las entradas `DEVICE` incluyen una lista de los dispositivos RAID. Las entradas `ARRAY` contienen los conjuntos RAID y sus opciones. En este ejemplo se implementa un conjunto simple de dos discos. Se utilizan discos Serial ATA para que tengan un prefijo de nombre de dispositivo `sd` en vez de `hd`.

```
DEVICE /dev/sda1 /dev/sdb1
```

Puede incluir más de un dispositivo en una entrada `DEVICE`, además de entradas `DEVICE` separadas. También puede especificar varios dispositivos al utilizar símbolos de relación, como `*`, `?` o `[]`. Lo siguiente especifica todas las particiones en un disco `sda` como dispositivos RAID:

```
DEVICE /dev/sda* /dev/sdb1
```

En el caso de una entrada **ARRAY**, se especifica el nombre del dispositivo RAID que está configurando, como `/dev/md0` para el primer dispositivo RAID. Luego se agregan opciones de configuración como `devices` para mostrar las particiones que integran el conjunto, `level` para el nivel de RAID y `num-devices` para el número de dispositivos.

```
ARRAY dev/md0 devices=/dev/sda2,/dev/sdb1,/dev/sdc1 level=5 num-devices=3
```

En el ejemplo anterior se configura el conjunto RAID `/dev/md0` como tres discos (particiones) al utilizar `/dev/sda1`, `/dev/sdb1` y `/dev/sdc1` y se configura para RAID 5 (`level=5`).

Creación de un conjunto RAID

Se crea un conjunto RAID con las opciones especificadas con el comando `mdadm` o las configuraciones incluidas en el archivo `/etc/mdadm.conf`. No es obligatorio el uso del archivo `/etc/mdadm.conf`, aunque hace más manejable la creación de RAID, sobre todo en conjuntos grandes o complejos. Una vez que ha creado sus dispositivos RAID, se activarán automáticamente. Con el siguiente comando se crea un conjunto RAID, `/dev/md1`, al utilizar dos dispositivos, `/dev/sda2`, `/dev/sdb1` y `/dev/sdc1`, en un nivel 5.

```
mdadm --create /dev/md1 --raid-devices=3 /dev/sda1 /dev/sdb1 /dev/sdc1 --level=5
```

Cada opción tiene una versión corta correspondiente, como se muestra en la tabla 30-5. Aquí se muestra el mismo comando con opciones de una sola letra.

```
mdadm -C /dev/md1 -n3 /dev/sda1 /dev/sdb1 /dev/sdc1 -l5
```

Si ha configurado sus dispositivos RAID en el archivo `/etc/mdadm.conf`, se utiliza el comando `mdadm` en el modo de creación para crear sus dispositivos RAID. `mdadm` toma como argumento el nombre del dispositivo RAID, como `/dev/md0` para el primer dispositivo RAID. Después ubica la entrada para ese dispositivo en el archivo `/etc/mdadm.conf` y utiliza tal información de configuración para crear el sistema de archivos RAID en ese dispositivo. Puede especificar un archivo de configuración alternativo con la opción `-c`, si lo desea. `mdadm` opera como una especie de comando `mkfs` para dispositivos RAID, inicializando las particiones y creando el sistema de archivos RAID. Se borrará cualquier dato que haya en las particiones que integran el conjunto RAID.

```
mdadm --create /dev/md0
```


Opción de mdadm -create	Descripción
-n --raid-devices	Número de dispositivos RAID
-l --level	Nivel RAID
-C --create	Modo de creación
-c --chunk	Tamaño de fragmento (franja) en potencias de 2: la opción predeterminada es 64 KB
-x --spare-devices	Número de dispositivos sobrantes en el conjunto
-z --size	Tamaño de bloques utilizados en dispositivos; como opción predeterminada se establece en el dispositivo más pequeño, si no son del mismo tamaño
-p --parity	El algoritmo de paridad; como opción predeterminada se utiliza left-symmetric

TABLA 30-5 Opciones de mdadm -create

Creación de grupos de reemplazo

Ahora el software de RAID de Linux permite que los conjuntos RAID compartan sus dispositivos de reemplazo. Esto significa que, si los conjuntos pertenecen al mismo grupo de reemplazo, un dispositivo que falla en un conjunto utiliza automáticamente el reemplazo en otro conjunto. Los dispositivos de reemplazo de cualquier conjunto se utilizan en otro a medida que se necesitan. El grupo de reemplazo al que pertenece un conjunto se establece con la opción **--spare-group**. El modo de monitoreo de **mdadm** detectará la falla en un dispositivo de un conjunto y lo reemplazará automáticamente con un dispositivo de reemplazo tomado de los conjuntos en el mismo grupo de reemplazo. El primer comando en el siguiente ejemplo crea un disco de reemplazo llamado **/dev/hdd1** para el conjunto **/dev/md0** y le asigna la etiqueta **migrupo**. En el segundo comando, el conjunto **/dev/md1** no tiene un disco de reemplazo pero pertenece al mismo grupo de reemplazo que el conjunto **/dev/md0**. Si un disco en **/dev/md1** falla, se utiliza automáticamente el dispositivo de reemplazo, **/dev/hdd1**, de **/dev/md0**. Las siguientes líneas de código son realmente dos líneas, cada una comienza con **mdadm**:

```
mdadm --create /dev/md0 --raid-devices=3 /dev/hda1 /dev/hdc1 -x
      /dev/hdd1 --level=1 --spare-group=migrupo
mdadm --create /dev/md1 --raid-devices=2 /dev/hda2 /dev/hdc2 --level=1
      --spare-group=migrupo
```

Creación de un sistema de archivos

Una vez que los dispositivos RAID están activados, se crean y se montan los sistemas de archivo en los dispositivos RAID. En el siguiente ejemplo se crea un sistema de archivos de Linux estándar en el dispositivo: **/dev/md0**:

```
mkfs.ext3 /dev/md0
```

En el siguiente ejemplo, el usuario crea un directorio llamado **/miraid** y monta el dispositivo RAID ahí:

```
mkdir /miraid
mount /dev/md0 /miraid
```


Si planea utilizar su dispositivo RAID para administrar sus directorios y archivos de usuario, monte el dispositivo RAID como partición **/home**. Por lo general, este punto de montaje se utilizaría si creara sus dispositivos RAID cuando instala su sistema. Para transferir sus directorios de inicio actuales a un dispositivo RAID, primero debe respaldarlos en otra partición y después montar su dispositivo RAID, copiando sus directorios de inicio en éste.

Administración de conjuntos RAID

Los conjuntos RAID se administran con las operaciones del modo de administración de **mdadm**. En este modo se agregan o eliminan dispositivos en conjuntos, además de que se marcan los dispositivos que fallan. La opción **--add** le permite agregar un dispositivo a un conjunto activo, esencialmente una operación de intercambio activo.

```
mdadm /dev/md0 --add /dev/sdd1
```

Para eliminar un dispositivo de un conjunto activo, primero tiene que marcarlo como fallido con la opción **--fail** y después eliminarlo con **--remove**.

```
mdadm /dev/md0 --fail /dev/sdb1 --remove /dev/sdb1
```

Inicio y detención de conjuntos RAID

Para iniciar un conjunto RAID existente, se utiliza **mdadm** con el modo de ensamblado (los conjuntos recién creados se inician de manera automática). Para hacerlo directamente en la línea de comandos es necesario que también sepa cuáles dispositivos integran el conjunto, al incluirlos después del conjunto RAID.

```
mdadm -A /dev/md1 /dev/sda2 /dev/sdb1
```

Es más fácil configurar sus conjuntos RAID en el archivo **/etc/mdadm.conf**. Con la opción de rastreo, **-s**, **mdadm** leerá la información del conjunto del archivo **/etc/mdadm.conf**. Si no especifica un conjunto RAID, todos los conjuntos se iniciarán.

```
mdadm -As /dev/md0
```

Para detener un conjunto RAID, se utiliza la opción **-S**.

```
mdadm -S /dev/md0
```

Monitoreo de conjuntos RAID

Como es un daemon, **mdadm** se inicia y detiene al utilizar el servicio **mdmonitor** en **/etc/init.d**. Esto invoca **mdadm** en el modo de monitoreo, que detecta cualquier problema que surja y registra informes y toma acciones apropiadas.

```
service mdadm start
```

Se monitorean dispositivos directamente al invocar **mdadm** con el modo de monitoreo.

```
mdadm --monitor /dev/md0
```

Las opciones relacionadas con el monitoreo se establecen en el archivo **/etc/mdadm.conf**. **MAILADDR** establece la dirección de correo a la que se envían las notificaciones de los sucesos de RAID. **PROGRAM** configura el programa que se utiliza, si ocurren sucesos.

Si decide cambiar su configuración RAID o agrega nuevos dispositivos, primero tiene que desactivar sus dispositivos RAID activos. Para desactivar un dispositivo RAID, se utiliza el

comando **mdadm** en el modo misceláneo. Asegúrese de cerrar antes cualquier archivo abierto y desmontar el sistema de archivos en el dispositivo.

```
umount /dev/md0
mdadm -S /dev/md0
```

Configuración de RAID de arranque

Los dispositivos RAID de arranque utilizan el nivel 0 o 1. En el siguiente ejemplo, el primer conjunto sólo almacena la partición de arranque y utiliza el nivel 1 de RAID, mientras que el segundo conjunto utiliza tres particiones y se configura en nivel 5 de RAID.

```
ARRAY /dev/md0    devices=/dev/sda1 level=1 num-devices=1
ARRAY /dev/md1    devices=/dev/sda2,/dev/sdb1/dev/sdc1 level=5 num-devices=3
```

En el ejemplo anterior se configura el conjunto RAID **/dev/md0** como un dispositivo RAID 1 (**level=1**). Dos discos (particiones) integran el segundo conjunto RAID, **/dev/md1**, que utiliza **/dev/sda2** y **/dev/sdb1**. Se configura para RAID 5 (**level=5**).

Puede crear un conjunto RAID utilizando opciones especificadas con el comando **mdadm** o utilizando configuraciones que se incluyen en el archivo **/etc/mdadm.conf**. Ya no es obligatorio el uso del archivo **/etc/mdadm.conf**, aunque permite manejar mejor la creación de RAID, sobre todo en el caso de conjuntos grandes o complejos. Una vez que cree sus dispositivos RAID, se activarán automáticamente. El siguiente comando crea un conjunto RAID, **/dev/md1**, al utilizar dos dispositivos, **/dev/sda2**, **/dev/sdb1** y **/dev/sdc1**, en el nivel 5.

```
mdadm --create /dev/md1 --raid-devices=2 /dev/sda2 /dev/sdb1 --level=1
```

Cada opción tiene una versión corta correspondiente, como se muestra en la tabla 30-5. El mismo comando se muestra aquí con opciones de una sola letra.

```
mdadm --C /dev/md1 -n3 /dev/sda2 /dev/sdb1/dev/sdc1 -l5
```

Para el primer conjunto del ejemplo anterior se utiliza:

```
mdadm -C /dev/md0 -n2 /dev/sdaa -l1
```

Para utilizar de forma más eficiente el espacio, se pueden utilizar las particiones de disco duro correspondientes, como se describe en la siguiente sección.

Particiones de disco duro correspondientes

El término *dispositivo* es confuso, porque también se utiliza para aludir a particiones de un disco duro particular que integran un dispositivo RAID. En realidad, un dispositivo RAID de software es un conjunto de particiones de disco duro, donde cada partición puede ocupar un disco duro completo, aunque no es necesario que la ocupe. En ese caso, se considera que un dispositivo RAID es un conjunto de discos duros (dispositivos). En la práctica, los discos duros de su configuración RAID suelen contener varias particiones de disco duro correspondientes, y cada conjunto tiene el mismo tamaño. Cada conjunto de particiones correspondientes integra un dispositivo RAID, para que pueda tener varios dispositivos RAID que utilizan el mismo conjunto de discos duros. Esto es cierto particularmente en el caso de las configuraciones de particiones de Linux, donde se colocan diferentes directorios de sistema en sus particiones. Por ejemplo, **/boot** puede estar en una partición, **/home** en otra y **/** (la raíz) en una más. Este método también le permite configurar una implementación RAID5 al utilizar sólo dos discos duros físicos. En efecto, puede tener un

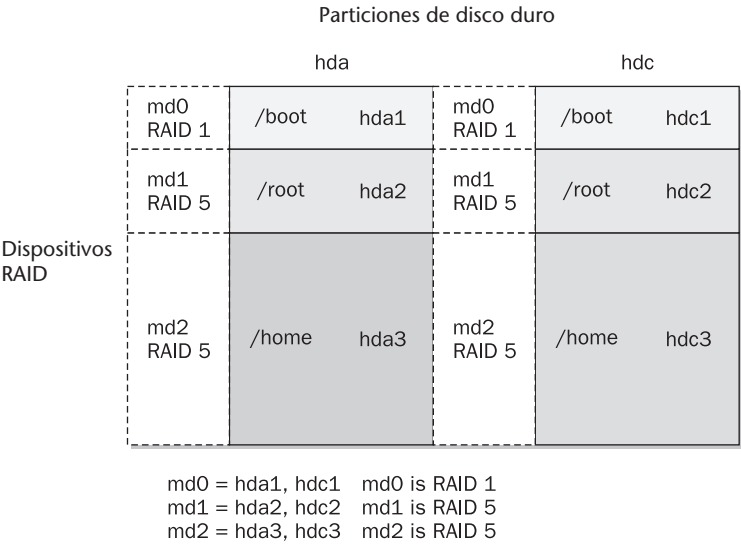


FIGURA 30-2 Dispositivos RAID (RAID 5 requerirá al menos tres particiones)

dispositivo RAID que utiliza cuatro particiones en dos discos duros. Sin embargo, perdería gran parte de la confiabilidad de una implementación RAID5 básica de tres o más discos.

Para configurar dispositivos RAID de modo que tenga particiones separadas para **/boot**, **/home** y **/** (raíz), necesita crear tres dispositivos RAID diferentes, como **md0** para **/boot**, **md1** para **/home** y **md2** para la raíz. Si tiene dos discos duros (por ejemplo **hda** y **hdc**), cada uno tendrá tres particiones, **/boot**, **/home** y **/**. El primer dispositivo RAID, **md0**, constaría de dos particiones **/boot**, la de **hda** y la de **hdc**. De forma similar, el segundo dispositivo RAID, **md1**, estaría integrado por dos particiones raíz, **/**, la de **hda** y otra en **hdc**. **md3** constaría de las particiones **/home** en **hda** y **hdc** (véase la figura 30-2).

Cuando crea las particiones de un dispositivo RAID determinado, es importante asegurarse de que cada partición tenga el mismo tamaño. Por ejemplo, la partición **/** que se utiliza para el dispositivo **md0**, en el disco **hda**, debe tener el mismo tamaño que la partición **md0** correspondiente en el disco **hdc**. Así que si la partición **md1** en **hda** es de 20 GB, entonces su partición correspondiente en **hdc** también debe ser de 20 GB. Si **md2** es de 100 GB en un disco, sus particiones correspo

Ejemplo de RAID

En la figura 30-2 se muestra una configuración simple con tres dispositivos RAID que utilizan particiones correspondientes en dos discos duros para **/boot**, **/** (raíz) y **/home**. La partición de arranque se configura como un dispositivo RAID porque los sistemas sólo pueden arrancar desde un dispositivo RAID 1, no RAID 5. Para simplificar este ejemplo, sólo se utilizan dos particiones para RAID5. Pero tenga en mente que RAID5 requiere al menos tres particiones.

Puede configurar este sistema durante la instalación, al seleccionar y formatear sus dispositivos RAID y sus particiones con Disk Druid. Para los pasos descritos aquí se supone que ya tiene su

sistema instalado en un disco IDE estándar y que los dispositivos RAID están configurados en otras dos unidades de discos IDE. Entonces puede copiar su archivo de su unidad estándar a sus dispositivos RAID.

Primero debe crear las particiones de disco duro al utilizar una herramienta de partición como **parted** o **fdisk**. Después, debe configurar los tres dispositivos RAID en el archivo **/etc/mdadm.conf**.

```
DEVICE    /dev/hda1 /dev/hda2 /dev/hda3 /dev/hdc1 /dev/hdc2 /dev/hdc3

ARRAY /dev/md0    devices=/dev/hda1, /dev/hdc1 level=1 num-devices=2
ARRAY /dev/md1    devices=/dev/hda2, /dev/hdc2 level=5 num-devices=2
ARRAY /dev/md2    devices=/dev/hda3, /dev/hdc3 level=5 num-devices=2
```

A continuación, debe crear sus dispositivos RAID con **mdadm**, que se activarán entonces automáticamente.

```
mdadm --create /dev/md0 /dev/md1 /dev/md2
```

Cree su sistema de archivos en los dispositivos RAID.

```
mkfs.ext3 md0 md1 md2
```

Entonces puede migrar los archivos **/boot**, **/** y **/home** de su disco duro actual a sus dispositivos RAID. Instale su cargador de arranque en el primer dispositivo RAID, **md0**, y cargue el sistema de archivos raíz del segundo dispositivo RAID, **md1**.

Como opción, puede crear primero los conjuntos con el comando **mdadm** y después generar entradas **ARRAY** para un archivo **/etc/mdadm.conf** a partir de la información de RAID para administrar después sus conjuntos, al agregar o eliminar componentes. Los siguientes comandos crean los tres dispositivos RAID del ejemplo anterior:

```
mdadm --create /dev/md0 --raid-devices=2 /dev/hda1 /dev/hdc1 --level=0
mdadm --C /dev/md2 -n3 /dev/hda2 /dev/hdc2 -l5
mdadm --C /dev/md2 -n3 /dev/hda3 /dev/hdc3 -l5
```

Entonces puede generar las entradas de **ARRAY** para el archivo **/etc/mdadm.conf** directamente con el siguiente comando. Todavía tendrá que editar **mdadm.conf** y agregar las entradas de **DEVICE**, además de las de monitoreo, como **MAILADDR**.

```
mdadm --detail --scan > /etc/mdadm.conf
```

El ejemplo anterior también se puede implementar al utilizar RAID 0 para arranque y RAID1 para **md1** y **md2**, haciendo que el disco **hdc** funcione como un disco de espejo para los conjuntos **md0** y **md1**.

Dispositivos y módulos

Todos los dispositivos, como impresoras, terminales y CD-ROM, se conectan a su sistema operativo Linux mediante archivos especiales llamados *archivos de dispositivo*. Estos archivos contienen toda la información que su sistema operativo necesita para controlar el dispositivo especificado. Este diseño introduce gran flexibilidad. El sistema operativo es independiente de los detalles específicos para administrar un dispositivo particular; el archivo de dispositivo maneja las especificaciones. El sistema operativo sólo informa al dispositivo qué tarea va a realizar, y el archivo de dispositivo le dice cómo. Si cambia los dispositivos, sólo tiene que cambiar el archivo de dispositivo, no todo el sistema.

Para instalar un dispositivo en su sistema Linux, necesita un archivo de dispositivo para éste, una configuración de software como la que proporciona una herramienta de configuración y el soporte de kernel (por lo general, proporcionado por un módulo o soporte que ya está compilado en su kernel). Los archivos de dispositivos no se manejan de forma estática, sino udev los genera de forma dinámica, de acuerdo con las necesidades, y HAL los administra. Antes, se creó un archivo para cada dispositivo posible, lo que lleva a que se tenga un gran número de archivos de dispositivos en el directorio `/etc/dev`. Ahora, su sistema sólo detectará los dispositivos que utiliza y para los que crea archivos de dispositivos, dándole una lista mucho más pequeña de archivos de dispositivo. udev y HAL son sistemas de conexión activa, donde se utiliza udev para crear dispositivos y HAL está diseñado para proporcionar información acerca de éstos, además de administrar la configuración de dispositivos extraíbles, como los de los sistemas de archivos para los lectores de tarjetas USB y los CD-ROM.

La administración de los dispositivos es al mismo tiempo más fácil pero mucho más compleja. Ahora tiene que utilizar udev y HAL para configurar dispositivos, aunque gran parte de esto ahora es automático. La información de dispositivo se mantiene en un sistema de archivos de dispositivo especial llamado **sysfs** en `/sys`. Éste es un sistema de archivos virtual como `/proc` y se utiliza para dar seguimiento a todos los dispositivos soportados por el kernel. Varios de los recursos que tal vez necesite consultar y de los directorios que quizá tenga que usar se presentan en la tabla 31-1.

El sistema de archivos sysfs: `/sys`

El sistema de archivos del sistema está diseñado para almacenar información detallada de los dispositivos del sistema. Herramientas de conexión activa, como udev, utilizan esta información para crear interfaces de dispositivo, a medida que se necesiten. En lugar de tener una configuración

Recurso	Descripción
/etc/sysconfig/hwconf	Configuración y lista de hardware de su sistema
/sys	El sistema de archivos sysfs que presenta información de la configuración de todos los dispositivos de su sistema
/proc	Un antiguo sistema de archivos de proceso que presenta información de kernel, incluyendo información de dispositivos
kernel.org/pub/linux/docs/device-list/devices.txt	Nombres de dispositivo de Linux
kernel.org/pub/linux/utls/kernel/hotplug/udev.html	El sitio Web de udev
/etc/udev	El directorio de configuración de udev
freedesktop.org/wiki/Software/hal	El sitio Web de HAL
/etc/hal	El directorio de configuración de HAL
/usr/share/hal/fdi	Los archivos de información del dispositivo HAL, para configurar soporte y directivas de información de HAL
/etc/hal/fdi	Archivos de información del dispositivo del administrador del sistema HAL

TABLA 31-1 Recursos de dispositivo

manual estática y completa para un dispositivo, el sistema **sysfs** mantiene información de configuración del dispositivo, que después es utilizada, de acuerdo con las necesidades, por el sistema de conexión activa para crear dispositivos e interfaces cuando un dispositivo necesite conectarse a su sistema.

Cada vez son más los dispositivos extraíbles, y muchos están hechos para conectarse de forma temporal (por ejemplo, las cámaras). En vez de administrar métodos estáticos y dinámicos separados por dispositivos de configuración, las distribuciones de Linux hacen que todos los dispositivos sean de conexión activa de forma estructural.

El sistema de archivos **sysfs** es virtual y proporciona un mapa jerárquico de sus dispositivos compatibles con el kernel como PCI, buses y dispositivos de bloque, además de módulos de kernel de soporte. El subdirectorio **classes** incluirá todos sus dispositivos compatibles por categoría, como la red y los dispositivos de sonido. Con **sysfs** su sistema determina de manera sencilla el archivo de dispositivo al que un dispositivo particular está asociado. Esto es muy útil para administrar dispositivos extraíbles, además de administrar y configurar dispositivos de forma dinámica, como lo hacen HAL y udev. udev utiliza el sistema de archivos **sysfs** para generar de forma dinámica los archivos de dispositivo necesarios en el directorio **/dev**; HAL lo utiliza para administrar archivos de dispositivos extraíbles, de acuerdo con las necesidades. El tipo de sistema de archivos **/sys** es **sysfs**. Los subdirectorios **/sys** organizan sus dispositivos en diferentes categorías. **systool** utiliza el sistema de archivos para desplegar una lista de sus dispositivos instalados. La herramienta es parte del paquete **sysfsutils**. Con el siguiente ejemplo se presenta una lista de todos sus dispositivos de sistema.

```
systool
```

Como **/proc**, el directorio **/sys** reside sólo en la memoria, pero todavía se monta en el archivo **/etc/fstab**.

```
none    /sys            sysfs            defaults    0            0
```

Archivo	Descripción
/proc/devices	Presenta una lista los controladores de dispositivos configurados para el kernel en ejecución.
/proc/dma	Despliega los canales DMA en uso.
/proc/interrupts	Despliega los IRQ (interruptores) en uso.
/proc/ioports	Muestra los puertos de entrada y salida en uso.
/proc/pci	Muestra una lista de los dispositivos PCI.
/proc/asound	Presenta una lista de los dispositivos de sonido.
/proc/ide	Directorio para dispositivos IDE.
/proc/net	Directorio para dispositivos de red.

TABLA 31-2 Archivos de información de dispositivo **/proc**

El sistema de archivos proc: /proc

El sistema de archivos **/proc** (véase el capítulo 29) es un sistema de archivos más antiguo, que se utilizaba para mantener información de procesos de kernel, incluidos los dispositivos. Mantiene archivos de información especiales para sus dispositivos, aunque muchos de éstos ahora tienen soporte en el sistema de archivos **sysfs**. El archivo **/proc/devices** incluye una lista de sus dispositivos de carácter y bloque instalados, junto con sus números principales. Los IRQ, DMA y puertos de entrada y salida utilizados por los dispositivos se incluyen en los archivos **interrupts**, **dma** e **ioports**, respectivamente. Ciertos archivos incluyen información que cubre varios dispositivos, como **pci**, que contiene todos sus dispositivos PCI, y **sound**, que incluye todos sus dispositivos de sonido. El archivo **sound** contiene información detallada acerca de su tarjeta de sonido. Varios subdirectorios, como **net**, **ide** y **scsi**, contienen archivos de información para diferentes dispositivos. Ciertos archivos almacenan información de configuración que cambia de forma dinámica, como la capacidad de reenviar paquetes IP y el número máximo de archivos. Es posible cambiar estos valores con la herramienta **sysctl** (Kernel Tuning, en el menú Herramientas del sistema) o mediante la edición manual de ciertos archivos. En la tabla 31-2 se presentan varios archivos **/proc** relacionados con dispositivos.

udev: archivos de dispositivo

Los dispositivos ahora son de *conexión activa*, lo que significa que se conectan y desconectan de forma sencilla. Su configuración se detecta automáticamente y no depende de configuraciones administrativas manuales. La herramienta de conexión activa que se utiliza para detectar archivos de dispositivos es **udev** (user devices, dispositivos de usuarios). Cada vez que su sistema arranca, **udev** detecta automáticamente sus dispositivos y genera archivos de dispositivo para éstos en el directorio **/etc/dev**. Esto significa que el directorio **/etc/dev** y sus archivos se vuelven a crear cada vez que arranca. Es un directorio dinámico, no estático. **udev** utiliza un conjunto de reglas para determinar cómo se generan los archivos de dispositivo, incluido cualquier vínculo simbólico correspondiente. Éstos se ubican en el archivo **/etc/udev/rules.d**. Encontrará más información acerca de **udev** en kernel.org/pub/linux/utils/kernel/hotplug/udev.html.

Como parte del sistema de conexión activa, **udev** detecta automáticamente dispositivos kernel que se agregan o eliminan del sistema. Cuando la interfaz de dispositivo se crea por primera vez, se localiza y lee su archivo **sysfs** correspondiente, determinando cualquier atributo adicional como

números de serie y números mayores o menores de dispositivos que se utilizan para identificar de forma única el dispositivo. Éstos se utilizan como claves en las reglas de udev para crear la interfaz de dispositivo. Una vez que el dispositivo se crea, se incluye en la base de datos udev, que da seguimiento a los dispositivos instalados.

Si se agrega un dispositivo, se llama a udev para conexión activa. Revisa el archivo **sysfs** en busca del dispositivo con los números mayor y menor, si se proporcionan. Luego utiliza las reglas de sus archivos de reglas para crear el archivo de dispositivo y cualquier vínculo simbólico para crear el archivo de dispositivo en **/dev**, con permisos especificados para el dispositivo en las reglas de permisos. Una vez que el archivo de dispositivo se crea, udev ejecuta los programas en **/etc/dev.d**.

NOTA Cuando el sistema inicia, invoca a **/sbin/udevstart**, que ejecuta udev y crea todos los dispositivos de kernel, al crear archivos de dispositivos en el directorio **/dev**.

Como ahora **/etc/dev** es dinámico, cualquier cambio que haga manualmente al directorio **/etc/dev** se perderá cuando reinicie. Esto incluye la creación de cualquier vínculo simbólico como **/dev/cdrom** que muchas aplicaciones de software utilizan. En cambio, tales vínculos simbólicos tienen que configurarse en los archivos de reglas de udev, que se ubican en el directorio **/etc/udev/rules.d**. Las reglas predeterminadas ya están establecidas para los vínculos simbólicos, pero puede crear y agregar sus propias reglas.

Configuración de udev

El archivo de configuración para udev es **/etc/udev/udev.conf**. Aquí se establecen opciones de udev globales como la ubicación de la base de datos de udev; las opciones predeterminadas para los permisos de dispositivo, el propietario y el grupo; y la ubicación de los archivos de reglas de udev. La herramienta udev utiliza el archivo **rules.d** para crear de forma dinámica sus archivos de dispositivo. Tenga cuidado al hacer cualquier cambio, sobre todo a las ubicaciones de los archivos de reglas. El soporte para todos los dispositivos en su sistema depende de estas reglas. El archivo **udev.conf** se muestra aquí:

```
# udev.conf
# The main config file for udev
#
# This file can be used to override some of udev's default values
# for where it looks for files, and where it places device nodes.
#
# WARNING: changing any value, can cause serious system breakage!

# udev_root - where in the filesystem to place the device nodes
udev_root="/dev/"

# udev_db - The name and location of the udev database.
udev_db="/dev/.udev.tdb"

# udev_rules - The name and location of the udev rules file
udev_rules="/etc/udev/rules.d/"
#udev_permissions - The name,location of the udev permission file
udev_permissions="/etc/udev/permissions.d/"

# default_mode - set the default mode for all nodes that have no
# explicit match in the permissions file
default_mode="0600"
```

```
#default_owner - set the default owner for all nodes that have no
# explicit match in the permissions file
default_owner="root"

#default_group - set the default group for all nodes that have no
# explicit match in the permissions file
default_group="root"

# udev_log - set to "yes" if you want logging, else "no"
udev_log="no"
```

La ubicación de los archivos de dispositivo se establece con **udev_root** a **/dev**. Éste es el directorio oficial en sistemas Linux y no debe cambiarse. El archivo **udev_rules** especifica dónde se ubican los archivos de reglas que udev utiliza para generar los archivos de dispositivo. La opción **udev_log** le permite activar o desactivar los registros, lo que es útil para detectar errores. La opción **udev_permissions** especifica la ubicación de los archivos de permisos que almacena los permisos que se aplican a ciertos dispositivos. Los permisos predeterminados ya aparecen en la lista. En el caso en que un dispositivo no tenga permisos en el archivo correspondiente, se utilizan las opciones predeterminadas que se establecen con **default_mode**, **default_owner** y **default_group**. Esto funciona más a manera de seguro contra fallos. Un permiso de dispositivo se debe establecer en un archivo de permiso udev.

Nombres de dispositivos y reglas de udev: **/etc/udev/rules.d**

El nombre de un archivo de dispositivo se designa para reflejar la tarea del dispositivo. Los de impresora comienzan con **lp** (de Line Print, impresión de línea). Debido a que puede tener más de una impresora conectada a su sistema, los archivos de dispositivo de impresora se distinguen por dos o más números o letras después del prefijo **lp**, como **lp0**, **lp1**, **lp2**. Lo mismo pasa con los archivos de dispositivo de terminal, que comienzan con el prefijo **tty** (de Teletype, teletipo), que se diferencian por números y letras como **tty0**, **tty1**, **ttyS0**, etc. Obtendrá una lista completa de nombres de archivos de dispositivos actuales y los dispositivos para los que se usan en el sitio Web **kernel.org** en kernel.org/pub/linux/docs/device-list/devices.txt.

Con udev, los nombres de dispositivos se determinan de forma dinámica mediante reglas que se incluyen en los archivos de reglas udev. Éstos se ubican en **/etc/udev/rules.d**. Durante la instalación, su sistema genera los archivos de reglas que encontrará en este directorio. Nunca debe editarlos. Si necesita agregar reglas propias, debe crear su propio archivo de reglas. Los nombres de los archivos de reglas se asignan empezando con un número para establecer la prioridad. Se leen de forma secuencial, y las primeras reglas invalidan cualquier conflicto futuro. Todos los archivos de reglas tienen una extensión **.rules**. Las reglas se organizan en tres categorías generales: nombres, permisos/proprietario y vínculos simbólicos. Además, existen varias categorías especializadas para ciertos tipos de dispositivos como faxes y ratones. En Red Hat, Fedora y distribuciones similares, las tres categorías se incluyen en un archivo de reglas primarias llamado **50-udev.rules**. Aquí encontrará las reglas para la mayoría de sus dispositivos de sistema. En Debian, Ubuntu y distribuciones similares, las categorías generales se almacenan en archivos separados: **20-names.rules**, **40-permissions.rules** y **60-symlink.rules**.

Otros archivos de reglas pueden configurarse para dispositivos más especializados como **60-rules-libsane** para escáneres, **60-rules-libmtp** para reproductores de música, **60-pcmcia.rules** para dispositivos PCMCIA y **90-alsa.rules** para el controlador de sonido. Los archivos de reglas que ya están presentes en el directorio **rules.d** han sido proporcionados para su distribución de Linux y están diseñados específicamente para ésta. Nunca debe modificar estas reglas. Para personalizar

su configuración, cree sus propios archivos de reglas en **/etc/udev/rules.d**. En su archivo de reglas normalmente se definen sólo symlinks, al utilizar sólo los campos SYMLINK, como se describe en las siguientes secciones. Esto configura vínculos simbólicos con dispositivos, permitiéndole acceder a éstos con otros nombres de dispositivo. Los campos NAME se utilizan para crear la interfaz de dispositivo original, tarea que se suele dejarse al propio udev.

Cada línea asigna un atributo de dispositivo a un nombre, además de especificar cualquier nombre simbólico (vínculos). Los atributos se especifican al utilizar claves, de las que puede haber más de una. Si todas las claves coinciden con un dispositivo, entonces el nombre asociado se utiliza para éste y se genera un archivo de dispositivo con ese nombre. Se utiliza una clave asignable, como NAME para el nombre de dispositivo o SYMLINK para un nombre simbólico, para asignar el valor relacionado. En vez de mostrar un nombre de dispositivo, puede especificarse un programa o una secuencia de comandos, en lugar de generar el nombre. Así suele pasar con dispositivos DVD/CD-ROM, donde el nombre de dispositivo puede ser un dvdwriter, cdwriter, cdrom o dvdrom.

Las reglas constan de una lista de campos separados por comas. Un campo está integrado por una relación o clave asignable. Las claves de relación utilizan los operadores == y != para determinar si es igual o no. Los operadores *, ? y [] relacionan caracteres, cualquier carácter solo o una clase de caracteres, como en la shell. Las claves asignables utilizan operadores =, += y := para asignar valores. El operador = asigna un solo valor, += adjunta el valor a los ya asignados y el operador := hace una asignación final, evitando cambios futuros. Las claves asignables también dan soporte a las claves udev que aparecen en la tabla 31-3. Revise la página Man de udev para conocer descripciones detalladas.

Los campos de clave como KERNEL permiten la inclusión de patrones para especificar colecciones de dispositivos. Por ejemplo, mouse* buscará todos los dispositivos que comiencen con el patrón "Mouse". En el siguiente campo se utiliza la clave KERNEL para encontrar todos los dispositivos mouse que aparecen en el kernel:

```
KERNEL=="mouse*"
```

La siguiente clave encontrará todos los dispositivos de impresora numerados de **lp0** a **lp9**. Utiliza corchetes para especificar un rango de números o caracteres, en este caso de 0 a 9, **[0-9]**:

```
KERNEL=="lp[0-9]*"
```

Los campos NAME, SYMLINK y PROGRAM permiten códigos de sustitución de cadena, de manera similar a como trabajan los códigos de impresión. Este código empieza con un símbolo %. El código permite que se haga referencia a varios dispositivos y nombres en la misma regla. En la tabla 31-4 se presentan los códigos soportados.

La página Man de udev proporciona varios ejemplos de reglas de udev mediante el uso de varios campos. Los archivos **50-udev.rules** y **20-udev.rules** almacenan reglas que utilizan principalmente claves KERNEL para designar dispositivos. La clave KERNEL es seguida por una clave NAME para especificar el nombre del archivo de dispositivo o una clave SYMLINK para configurar un vínculo simbólico con un archivo de dispositivo. La siguiente regla utiliza la clave KERNEL para relacionar todos los dispositivos mouse que aparecen en el kernel. Los nombres de dispositivo correspondientes se colocan en el directorio **/dev/input**, y el nombre que se utiliza es el nombre de kernel para el dispositivo (**%k**):

```
KERNEL=="mouse*", NAME="input/%k"
```

Esta regla utiliza una clave BUS y una clave KERNEL para configurar archivos de dispositivos para impresoras USB, cuyos nombres de kernel se utilizarán para crear archivos de dispositivo en **/dev/usb**:

Claves de relación	Descripción
ACTION	Busca una coincidencia con la acción del suceso.
DEVPATH	Busca una coincidencia con la ruta del dispositivo.
ENV{clave}	Busca una coincidencia con un valor de variable de entorno.
BUS	Busca una coincidencia con el tipo de bus del dispositivo. (El bus de dispositivos sysfs debe permitir que sea determinado por un symlink “device”).
DRIVER	Busca una coincidencia con el nombre de controlador de dispositivo.
ID	Busca una coincidencia con el número de dispositivo en el bus, como el ID de bus PCI.
KERNEL	Busca una coincidencia con el nombre de dispositivo de kernel.
PROGRAM	Utiliza un programa externo para determinar el dispositivo. Esta clave es válida si el programa tuvo éxito. La cadena regresada por el programa puede coincidir adicionalmente con la clave RESULT.
RESULT	Busca una coincidencia con la cadena regresada de la última llamada a PROGRAM. Esta clave puede utilizarse en cualquier regla siguiente, después de una llamada a PROGRAM.
SUBSYSTEM	Busca una coincidencia con el subsistema de dispositivo.
SYSFS{nombredearchivo}	Busca una coincidencia con el atributo de dispositivo, como una etiqueta, un vendedor, el número de serie USB, UUID de SCSI o la etiqueta de sistema de archivos.
Claves asignables	Descripción
NAME	El nombre del nodo que habrá de crearse, o el nombre al que debe cambiarse la interfaz de red.
OWNER, GROUP, MODE	Los permisos para el dispositivo.
PLACE	Busca una coincidencia con la ubicación en el bus, como el puerto físico de un dispositivo USB.
ENV{clave}	Exporta la variable al entorno.
IMPORT{tipo}	Importa los resultados de un programa, el contenido de un archivo de texto o claves almacenadas en un dispositivo principal. El tipo es program , file o parent .
SYMLINK	El nombre del vínculo simbólico (symlink) para el dispositivo.
RUN	Agrega un programa a la lista de programas que ejecuta el dispositivo.

TABLA 31-3 Claves de reglas udev

```
BUS=="usb", KERNEL=="lp[0-9]*", NAME="usb/%k"
```

Vínculos simbólicos

Ciertos archivos de dispositivo son, en realidad, vínculos simbólicos que tienen nombres de dispositivo comunes que suelen vincularse con el archivo de dispositivo real utilizado. Un *vínculo simbólico* es otro nombre para un archivo que se utiliza como acceso directo, que hace referencia a ese archivo. Los dispositivos comunes como impresoras, unidades CD-ROM, discos duros,

Código de sustitución	Descripción
%n	El número de kernel de un dispositivo
%k	El nombre de kernel del dispositivo
%M	El número mayor del kernel
%m	El número menor del kernel
%p	La ruta del dispositivo
%b	El nombre de dispositivo relacionado con la ruta del dispositivo
%c	La cadena devuelta por un campo PROGRAM (no se utiliza en un campo PROGRAM)
%s {nombredearchivo}	Contenido del atributo sysfs
%E{clave}	Valor de variable de entorno
%N	Nombre de un nodo de dispositivo temporal, para proporcionar acceso antes de que se cree el nodo real
%%	Cita el carácter %, en caso de que sea necesario en el nombre de dispositivo

TABLA 31-4 Códigos de sustitución udev

dispositivos SCSI y dispositivos de sonido, junto con muchos otros, tendrán vínculos simbólicos correspondientes. Por ejemplo, un vínculo simbólico **/dev/cdrom** vincula con el archivo de dispositivo real que se utiliza para su CD-ROM. Si su CD-ROM es un dispositivo IDE, es probable que utilice el archivo de dispositivo **hdc**. En este caso, **/dev/cdrom** es un vínculo con **/dev/hdc**. En efecto, **/dev/cdrom** es otro nombre para **/dev/hdc**. Las unidades de DVD/CD Serial ATA se vincularán con los dispositivos **sdc**, como **sdc0** para la primera unidad CD/DVD Serial ATA. Si su unidad funciona como escritor y lector de CD y DVD, tendrá varios vínculos con el mismo dispositivo. En este caso los vínculos **cdrom**, **cdrw**, **cdwriter**, **dvd**, **dvdrw**, **dvdwriter** se vincularán con el mismo dispositivo CD/DVD RW-ROM.

También existe un archivo de vínculo **/dev/modem** para su módem. Si éste se encuentra conectado al segundo puerto serial, su archivo de dispositivo será **/dev/ttyS1**. En este caso, **/dev/modem** será un vínculo con el archivo de dispositivo. Entonces las aplicaciones utilizarán **/dev/modem** para acceder a su módem, en lugar de tener que conocer el archivo de dispositivo real utilizado. En la tabla 31-5 se muestra una lista de vínculos de dispositivos de uso común.

udev crea los vínculos simbólicos mediante el uso del campo SYMLINK. Los vínculos simbólicos de un dispositivo se despliegan con la misma regla que crea un archivo de dispositivo (clave NAME) o en una regla separada que sólo especificará un vínculo simbólico. La inclusión de la clave NAME no tiene que ser específica, si se utiliza el nombre de dispositivo predeterminado. El signo + agregado al símbolo = creará automáticamente el dispositivo con el nombre predeterminado, sin requerir una clave NAME explícita en la regla. La siguiente regla es para impresoras paralelas. Incluye el nombre predeterminado y la clave NAME implícita al crear el dispositivo (+) y un vínculo simbólico, **par**. **%n** agregará un número al vínculo simbólico, como **par1**, **par2**, etc.

```
KERNEL=="lp[0-9]*",               SYMLINK+="par%n"
```

Si quiere crear más de un vínculo simbólico para un dispositivo, se muestran en el campo SYMLINK. De la siguiente manera se crean dos vínculos simbólicos, uno llamado **cdrom** y otro **cdrom-** con el nombre de kernel adjunto (**%k**).

Vínculo	Descripción
cdrom	Vincula con su archivo de dispositivo de CD-ROM, establecido en /etc/udev/rules.d
dvd	Vincula con su archivo de dispositivo de DVD-ROM, establecido en /etc/udev/rules.d
cdwriter	Vincula con su archivo de dispositivo de CD-R o CD-RW, establecido en /etc/udev/rules.d
dvdwriter	Vincula con su archivo de dispositivo de DVD-R o DVD-RW, establecido en /etc/udev/rules.d
modem	Vincula con su archivo de dispositivo de módem, establecido en /etc/udev/rules.d
floppy	Vincula con su archivo de dispositivo de disco flexible, establecido en /etc/udev/rules.d
tape	Vincula con su archivo de dispositivo de cinta, establecido en /etc/udev/rules.d
scanner	Vincula con su archivo de dispositivo de escáner, establecido en /etc/udev/rules.d
mouse	Vincula con su archivo de dispositivo de ratón, establecido en /etc/udev/rules.d
tape	Vincula con su archivo de dispositivo de cinta, establecido en /etc/udev/rules.d

TABLA 31-5 Vínculos simbólicos de dispositivo

```
SYMLINK+="cdrom cdrom-%k"
```

Si decide configurar una regla separada que especifica sólo un vínculo simbólico, éste se mantendrá en una lista esperando la creación de este dispositivo. Esto también le permite agregar otros vínculos simbólicos con un dispositivo en otros archivos de reglas. Esta situación es confusa porque se crean vínculos simbólicos para dispositivos que no se han generado. Los vínculos simbólicos se definirán y mantendrán hasta que se necesiten, cuando el dispositivo se genere. Por esto es por lo que hay más reglas SYMLINK que NAME en udev que realmente configuren archivos de dispositivos. En el caso de los dispositivos extraíbles, no se generará un nombre de dispositivo sino hasta que se conecten.

En los archivos **50-rules.udev** o **60-symlink.rules** encontrará varias reglas SYMLINK para dispositivos ópticos; una de ellas se muestra aquí, al utilizar el nombre predeterminado implícito:

```
KERNEL=="sd*[0-9]*", SYMLINK+="cdrom cdrom-%k"
```

En casi todos los casos, sólo necesitará vínculos simbólicos para dispositivos, al utilizar los nombres simbólicos oficiales. Muchos de éstos ya están definidos. Si necesita crear un solo vínculo simbólico, puede crear una regla SYMLINK para éste. Sin embargo, es necesario colocar una nueva regla SYMLINK antes de las reglas que asignan un nombre a ese dispositivo. udev lee las reglas SYMLINK para un dispositivo, y se mantienen hasta que el dispositivo se nombra. Después, esos nombres simbólicos se utilizan para ese dispositivo. Puede tener todos los vínculos simbólicos con dispositivos que desee, lo que significa que puede tener varias reglas SYMLINK para el mismo dispositivo. Cuando se encuentra la regla NAME para el dispositivo, sólo se adjuntan las claves SYMLINK previas.

Casi todos los nombres simbólicos estándar ya están definidos en el archivo **50-udev.rules** o **60-symlink.rules** (dependiendo de la distribución), como **audio** para el dispositivo de audio. En el siguiente ejemplo, se hace referencia al dispositivo por su clave KERNEL y el vínculo simbólico se aplica con la clave SYMLINK. Ésta es sólo una regla SYMLINK. La clave NAME está implícita:

```
KERNEL=="audio0", SYMLINK+="audio"
```

Campos de programa, claves **IMPORT{program}** y **/lib/udev**

Varios tipos de dispositivos utilizan secuencias de comandos especiales para determinar el nombre que se utiliza para éste. Resulta cierto, sobre todo, en el caso de lectores y grabadores de CD/DVD, para los cuales existen varios dispositivos de tipos muy diferentes, como CD-ROM, DVD-ROM o CD-RW. El vínculo simbólico que se utiliza es **cdwriter** para CD-RW, **cdrom** para un CD-ROM e incluso **dvd** para un DVD-ROM. Para determinar el vínculo simbólico correcto, se invoca un programa para determinar el dispositivo. Muchos son programas especializados y secuencias de comandos que se mantienen en el directorio **/lib/udev**, como **check-cdrom.sh**, que determina el tipo de CD-ROM. Un programa se invoca con la clave **PROGRAM** o **IMPORT{program}**. La clave **PROGRAM** ejecutará el programa y regresará 0 si es exitoso. Los valores regresados se almacenan en la clave **RESULTS**. La clave **IMPORT{program}** regresa valores asignables.

En Red Hat, Fedora y distribuciones similares, la secuencia de comandos **check-cdrom.sh** se utiliza para ver si un dispositivo IDE (hd) es DVD. En la siguiente regla, la secuencia de comandos pasa el nombre de kernel y el parámetro DVD y regresará un valor positivo, si el dispositivo es un DVD-ROM. Entonces se asigna **dvdn** al vínculo simbólico, como en **dvd1**. En este ejemplo se utilizan otras dos claves. **BUS** revisa si el dispositivo es un CD-ROM IDE, y **ATTRS{removable}=="1"** confirma si es extraíble. Las siguientes líneas son en realidad una sola.

```
KERNEL=="hd[a-z]", BUS=="ide", ATTRS{removable}=="1", PROGRAM=="check-cdrom.sh %k
DVD", SYMLINK+="dvd dvd-%k"
```

La siguiente regla se utiliza para CD-ROM simples. El campo **ATTRS** regresará el nombre del dispositivo IDE en el sistema de archivos **sys** y su salida se prueba para ver si es un CD-ROM. Después, se asigna el vínculo simbólico, como en **cdrom** o **cdrom-**, con su nombre de kernel adjunto. Las siguientes líneas son una sola.

```
KERNEL=="hd[a-z]", BUS=="ide", ATTRS{media}== cdrom, SYMLINK+="cdrom cdrom-%k"
```

Se utiliza la clave **IMPORT(program)** para importar y ejecutar un programa externo e importar sus resultados directamente como un valor asignado. En Debian, el programa que se utiliza para detectar los dispositivos de unidad DVD/CD es **cdrom_id**.

```
KERNEL=="sr[0-9]*|hd[a-z]*|pcd[a-z]*", IMPORT(program)="cdrom_id -export $temp-
node"
```

En Debian, se utilizan reglas **symlink** y de permiso separadas. Las reglas **symlink** utilizan la clave **ENV** para revisar un valor en las variables **cdrom_id** asociadas, como **ID-CDROM** para dispositivos CDROM.

```
ENV{ID_CDROM}== "?*" , SYMLINK+="cdrom"
```

En el caso de permisos, los CDROM se asignan al grupo **cdrom**.

```
ENV{ID_CDROM}== "?*" , GROUP+="cdrom"
```

Creación de reglas de udev

udev coloca las reglas predeterminadas para dispositivos en los archivos **/etc/udev/rules.d/50-udev.rules** o **20-names.rules**. Nunca debe editar este archivo, aunque se revise para ver si maneja los

nombres de dispositivos. Este archivo creará los archivos de dispositivo al utilizar los nombres de kernel oficiales. A menudo aplicaciones que esperan encontrar dispositivos con estos nombres particulares, como **lp0** para el dispositivo de impresora, hacen referencia directa a estos nombres.

Si quiere crear reglas propias, debe colocarlas en archivos de reglas separados. Las reglas NAME que dan nombre a los dispositivos se leen en orden alfabético, donde la primera regla tendrá precedencia sobre cualquiera que esté después. Sólo se usará la primera regla NAME de un dispositivo. Las posteriores para ese mismo dispositivo se ignorarán. Tenga en cuenta que una regla SYMLINK con un += incluye una regla NAME para el dispositivo predeterminado, aunque el campo NAME no se muestra explícitamente. Revise reactivated.net/writing_udev_rules.html para conocer un tutorial sobre escritura de reglas udev.

Como se están creando reglas para reemplazar a las predeterminadas, éstas tendrán que ejecutarse primero. Para ello, se colocan en un archivo de reglas que comienza con un número menor, digamos 10. Este archivo de reglas se ejecutaría antes que los archivos **50-udev.rules** o **20-names.rules**, que almacenan las reglas predeterminadas. Los archivos de reglas se leen en orden alfabético, y los números menores se leen primero. Puede crear un archivo llamado **10-user.rules** en el directorio **/etc/udev/rules.d** para colocar sus propias reglas. Por el contrario, si quiere reglas que se ejecutarán sólo si las predeterminadas fallan por alguna razón, utilice un archivo de reglas numerado después de 50, como **90-mispredeterminadas.rules**.

En la sección siguiente “Nombres persistentes: udevinfo”, se describe cómo crear una regla canon-pr para reemplazar a la regla de impresora predeterminada. La nueva regla canon-pr de usuario se colocaría en un archivo **10-user.rules** para que se ejecute antes de las reglas de impresora en el archivo **50-udev.rules** o **20-names.rules**, por lo que tendrá precedencia. La regla de impresora predeterminada en los archivos **50-udev.rules** y **20-names.rules** (que se muestra aquí) no se aplicaría a la impresora Canon.

```
BUS="usb", KERNEL="lp[0-9]*", NAME="usb/%k"
```

Reglas SYMLINK

En casi todos los casos, sólo necesitará crear vínculos simbólicos para dispositivos, al utilizar el nombre oficial. También puede crear reglas que sólo creen vínculos simbólicos. Sin embargo, éstas necesitan colocarse antes de las reglas de nombre que asignan un nombre a los dispositivos. udev lee estas reglas SYMLINK, que se mantienen hasta que se asigna nombre a un dispositivo, y después todos los nombres simbólicos se utilizarán para éste, que puede tener todos los vínculos simbólicos que se se quieran, lo que significa que habrá varias reglas SYMLINK para el mismo dispositivo. Cuando se encuentra la regla NAME para el dispositivo, sólo se adjuntan las claves SYMLINK.

Casi todos los nombres simbólicos ya están definidos en los archivos **50-udev.rules** y **60-symlink.rules**, como audio para el dispositivo de audio. En el siguiente ejemplo, se hace referencia al dispositivo por su clave KERNEL y el vínculo simbólico se aplica con el campo SYMLINK. El campo NAME para el dispositivo predeterminado está implícito:

```
KERNEL="audio0", SYMLINK+="audio"
```

Si quiere saber siempre el nombre para un dispositivo, basta con agregar una regla SYMLINK. Por ejemplo, si sabe que su DVD-ROM está conectado a la primera conexión IDE secundaria (**hdc**), cree un nombre simbólico propio con una regla SYMLINK. En el siguiente ejemplo, se crea un nuevo vínculo simbólico, **midvdrom**, para el DVD-ROM en el dispositivo **/dev/hdc**.

```
KERNEL=="hdc", SYMLINK="midvdrom"
```


Para que se utilice una regla SYMLINK, tiene que encontrarse antes de la regla NAME que nombra al dispositivo. Debe colocar estas reglas en un archivo que antecederá a los archivos **50-udev.rules** o **60-symlink.rules**, como **10-user.rules**.

Nombres persistentes: udevinfo

Las reglas udev predeterminadas denominarán a sus dispositivos mediante el uso de los nombres simbólicos oficiales reservados para ellos, como **lp n** para impresora, donde n es el número de impresora. En el caso de dispositivos fijos, como impresoras fijas, esto suele ser adecuado. Sin embargo, en el caso de dispositivos extraíbles, como impresoras USB que pueden estar conectadas a puertos USB en diferentes secuencias en momentos distintos, tal vez los nombres que se utilizan no hagan referencia a la misma impresora. Por ejemplo, si tiene dos impresoras USB, una Epson y una Canon, y conecta la Epson primero y la Canon después, a la impresora Epson se le dará el nombre **usb/lp0** y, a la Canon, **usb/lp1**. Sin embargo, si después las desconecta y luego conecta la Canon primero y la Epson después, entonces la Canon tendrá el nombre **usb/lp0** y la Epson **usb/lp1**. Si quiere que siempre tengan el mismo nombre, digamos **epson-pr** y **canon-pr**, tendrá que crear su propia regla para detectar estas impresoras y darles sus propios nombres simbólicos. Las reglas persistentes se almacenan en los archivos **65-input-persistent.rules** y **65-storage-persistent.rules**. La tarea clave en la creación de un nombre persistente consiste en utilizar información única para identificar el dispositivo. Puede crear una regla que haga referencia al dispositivo con la información única, identificándolo, y que después le asigne un nombre oficial pero que le da un nombre simbólico único. Luego se utiliza el nombre simbólico único, como **canon-pr**, para hacer referencia siempre a esa impresora y no a otra, cuando está conectada. En este ejemplo, se utiliza información, como el número de serie de la impresora Canon, para identificar a ésta. Después se le asigna el nombre oficial, **usb/lp0**, o **usb/lp1** si otra impresora se conectó primero, y después se le da un nombre simbólico único, **canon-pr**, que hará referencia al nombre oficial, cualquiera que sea. Al mantener el nombre oficial, como **lp0**, se preserva el acceso estándar al dispositivo, tal como lo utilizan muchas aplicaciones.

Se utiliza la información del sistema de archivos **/sys** acerca del dispositivo para detectar el dispositivo correcto al que se hará referencia con el vínculo simbólico. Puede usarse información de dispositivo **/sys** única, como el número de serie del vendedor o el nombre del vendedor, como referencia única del dispositivo. Para obtener esta información, primero necesita consultar el sistema de archivos **/sys**. Se hace esto con el comando **udevinfo**.

Primero necesitará saber dónde se ubica el dispositivo en el sistema de archivos **/sys**. Conecte su dispositivo, que se configurará y nombrará automáticamente, utilizando el nombre oficial. Por ejemplo, al conectar una impresora USB se creará un nombre de dispositivo **/dev/usb/lp0** para éste. Se utiliza este nombre de dispositivo para saber más acerca de dónde está la información de la impresora USB en **/sys**. Utilice el comando **udevinfo** con la opción **-q path** para consultar el nombre de ruta de **/sys**, y agregue la opción **-n** con el nombre de ruta completo del dispositivo para identificar el dispositivo que está buscando. El siguiente comando desplegará la ruta **/sys** para la impresora con el nombre de dispositivo **lp0**. En este caso, el dispositivo está en el subdirectorio **class**, bajo **usb**. Se supone que la ruta es **/sys**.

```
udevinfo -q path -n /dev/usb/lp0
/class/usb/lp0
```

Una vez que tenga la ruta **/sys** del dispositivo, puede utilizarla para desplegar información acerca del dispositivo. Utilice el comando **udevinfo** de nuevo con la opción **-a** para desplegar toda la información acerca del dispositivo y la opción **-p** para especificar su ruta en el sistema de archivo **/sys**. La lista es extensa, así que debe canalizar la salida a **less** o redirigirla a un archivo.

```
udevinfo -a -p /sys/class/usb/lp0 | less
```

Algunas de las salidas de clave que deben buscarse son el bus utilizado e información como el número de serie, el nombre del producto y el fabricante. Busque la información que identifica de forma única el dispositivo, como el número de serie. Algunos dispositivos son compatibles con diferentes buses, y la información puede ser distinta para cada uno. Asegúrese de utilizar la información para ese bus cuando configure sus claves en la regla udev.

```
BUS="usb"
ATTRS{serial}="300HCR"
ATTRS{manufacturer}="Canon"
ATTRS{idproduct}="1074"
ATTRS{product}="S330"
```

Tiene la opción de utilizar mucha de esta información en una clave ATTRS de una regla udev para identificar el dispositivo. La clave ATTRS (atributos) se utiliza para obtener información de `/sys` acerca de un dispositivo. Se utiliza la clave ATTRS con el campo al que quiere hacer referencia entre corchetes. Luego se busca en ese campo un valor que haga referencia al dispositivo particular que quiere. Utilice el signo = y un valor de campo válido contra el cual se comparará. Una vez que conozca el número de serie `/sys` de un dispositivo, puede utilizarlo en claves ATTRS de reglas udev para hacer referencia de forma única al dispositivo. La siguiente clave revisa el número de serie del campo de dispositivos para el número de serie de la impresora Canon:

```
ATTRS{serial}=="300HCR"
```

Ahora puede crearse una regla de usuario para la impresora Canon.

En otro archivo de reglas puede agregar sus propios vínculos simbólicos al utilizar información de `/sys` para identificar de forma única la impresora y el nombre de dispositivo con su nombre de kernel oficial. Las primeras dos claves, BUS y ATTRS, especifican la impresora particular. En este caso el número de serie de la impresora se utiliza para identificarla de forma única. La clave NAME nombrará la impresora al utilizar el nombre de kernel oficial, al que siempre se hace referencia con el código `%k`. Como se trata de una impresora USB, su archivo de dispositivo se colocará en el subdirectorio `usb`, `usb/%k`. Después la clave SYMLINK define el nombre simbólico único que habrá de utilizarse, en este caso `canon-pr` en el directorio `/dev/usb`.

```
BUS=="usb", ATTRS{serial}=="300HCR", NAME="usb/%k", SYMLINK="usb/canon-pr"
```

Las reglas se aplican de forma dinámica en tiempo real. Para ejecutar una nueva regla, sólo conecte su impresora USB (o desconéctela y vuelva a conectarla). Verá los archivos de dispositivo generados automáticamente.

Campos de permiso: MODE, GROUP, OWNER

Los permisos que se darán a diferentes archivos de dispositivos se determinan mediante los campos de permisos en las reglas udev. En Red Hat, Fedora y distribuciones similares, las reglas de permisos se ubican en el archivo `50-udev.rules`, mientras en Debian y distribuciones similares se ubican en el archivo `40-permissions.rules`. El campo MODE es una configuración de permiso de bits octales, al igual que los utilizados para permisos de archivo. Por lo general se asigna 660, con permiso de propietario y grupo de lectura/escritura. Se permite la búsqueda con patrón mediante los operadores `*`, `?` y `[]`. En el siguiente ejemplo se configuran los dispositivos de audio al propietario y grupo con permisos de lectura/escritura para el propietario y el grupo:

```
KERNEL=="audio*",      MODE="0660"
```

La entrada de dispositivo floppy especifica un grupo de disco flexible.

```
KERNEL=="fd[01]*",      GROUP="floppy", MODE="0660"
```

Los dispositivos de impresora USB utilizan el grupo **lp** con MODE 660.

```
KERNEL=="usb/lp*",      GROUP="lp", MODE="0660"
```

Los dispositivos de cinta utilizan el grupo **disk**.

```
KERNEL=="npt*",      GROUP="disk", MODE="0660"
```

La configuración predeterminada establece OWNER y GROUP en root con permisos de lectura/escritura para el propietario (600).

```
KERNEL=="*",      OWNER="root" GROUP="root", MODE="0600"
```

Capa de abstracción de hardware: HAL

El objetivo de la capa de abstracción de hardware (HAL, Hardware Abstraction Layer) es volver abstracto el proceso de acceso a las aplicaciones por parte de los dispositivos. No es necesario que las aplicaciones sepan algo de un dispositivo, ni siquiera su nombre simbólico. La aplicación sólo tiene que pedir un dispositivo de cierto tipo y después un servicio, como HAL, debe proporcionar el que está disponible. La implementación del dispositivo se oculta a las aplicaciones.

HAL hace que los dispositivos estén disponibles fácilmente para escritorios y aplicaciones al utilizar una estructura D-BUS (bus de dispositivo). Los dispositivos se administran como objetos a los que las aplicaciones acceden de manera sencilla. El daemon de HAL, **haldaemon**, proporciona el servicio D-BUS. El servicio HAL, **freedesktop.org**, administrado por **/org/freedesktop/HAL/Manager**, proporciona la interacción con el dispositivo.

HAL es un servicio de información para dispositivos. El daemon de HAL mantiene una base de datos dinámica de dispositivos de hardware conectados. Programas especializados de llamada utilizan esta información para mantener ciertos archivos de configuración de dispositivo. Así sucede con la administración de los dispositivos de almacenamiento extraíbles. HAL invocará a los programas especializados de llamada que utilizarán información de HAL para administrar dispositivos de forma dinámica. Los dispositivos extraíbles, como discos CD-ROM o lectores de tarjetas USB, se administran mediante llamadas especializadas con información de HAL, que detectan cuando esos elementos se conectan. La situación es confusa. Los programas de llamada realizan la tarea real, pero HAL proporciona la información de dispositivo. Por ejemplo, aunque el llamado **hal-system-storage-mount** monte un dispositivo, las opciones y puntos de montaje que se utilizan para las entradas CD-ROM se especifican en archivos de información de dispositivo de HAL que establecen directivas para administración de almacenamiento.

HAL tiene un impacto clave en el archivo **/etc/fstab**, que se utiliza para administrar sistemas de archivos. Ya no se trata de entradas que se mantienen en el archivo **/etc/fstab** para dispositivos extraíbles como CD-ROMs. HAL administra directamente estos dispositivos al utilizar su conjunto de llamados de almacenamiento como **hal-storage-mount** para montar un dispositivo o **hal-storage-eject** para eliminar uno. En efecto, ahora tiene que utilizar los archivos de información de dispositivo de HAL para administrar sus sistemas de archivos extraíbles.

HAL es un proyecto de software de **freedesktop.org**, que se especializa en herramientas de escritorio de fuente abierta. Revise la documentación de especificación de HAL más reciente en **freedektop.org**, bajo la página **software/HAL**, para conocer explicaciones detalladas de la manera en que trabaja HAL (véase el vínculo de especificaciones en la página de HAL: Latest HAL Specification). La documentación es muy detallada y completa.

El daemon de HAL y hal-device-manager (hal-gnome)

El daemon de HAL, **hald**, se ejecuta como el proceso **haldaemon**. La información proporcionada por el daemon de HAL para todos sus dispositivos se despliega al utilizar el administrador de dispositivos de HAL. El administrador de dispositivos de HAL es parte del paquete **hal-gnome**. Se accede a éste, una vez instalado, desde la entrada del menú Sistema | Administración | Hardware. El administrador de dispositivos de HAL se llama **hal-device-manager**.

Cuando ejecuta el administrador, despliega un árbol expandible de sus dispositivos, organizado por categorías en el panel de la izquierda. El panel de la derecha despliega información acerca del dispositivo seleccionado. Un panel Dispositivo mostrará la información básica del dispositivo, como el vendedor y el tipo de bus. El panel Avanzado presenta las propiedades de dispositivos de HAL definidas por este dispositivo, como se describe en secciones posteriores, además de las rutas del sistema de archivos **/sys** para este dispositivo. En el caso de los controladores de dispositivos, también habrá un panel USB o PCI. Por ejemplo, una grabadora de DVD puede tener una entrada para la propiedad **storage.cdrom.cdr** que indica que escribe discos CD-R. Puede encontrar un dispositivo IDE CD/DVD-ROM bajo IDE (tal vez algunos controladores IDE de terceros estén etiquetados como dispositivos SCSI). Una entrada típica se vería así (**bool** es el tipo de entrada, particularmente boolean):

```
storage.cdrom.cdr      bool      true
```

Los valores numéricos pueden utilizar un tipo **int** o **strlist**. La siguiente propiedad **write_speed** tiene un valor de 7056:

```
storage.cdrom.write_speed  strlist      7056
```

La ruta del sistema de archivos **/sys** también será una cadena. Se precederá con una categoría de propiedad de Linux. Las cadenas utilizarán un tipo **strlist** para varios valores y **string** para valores únicos. La siguiente entrada ubica la ruta de sistema de archivos **/sys** en **/sys/block/hdc**:

```
linux.sysfs_path        strlist      /sys/block/hdc
```

Configuración de HAL: /etc/hal/fdi y /usr/share/hal/fdi

La información de los dispositivos y las directivas para administrar dispositivos se almacena en archivos de información de dispositivo en los directorios **/etc/hal/fdi** y **/usr/share/hal/fdi**. Es en estos directorios donde se establecen propiedades como las opciones que se utilizan para CD-ROM en **/etc/fstab**.

La implementación de HAL en Linux configura administración de almacenamiento al enfocarse en métodos de almacenamiento para volúmenes montables, en vez de dispositivos particulares. Las propiedades de volumen definen acciones que habrán de tomarse y opciones válidas que pueden utilizarse. Las llamadas especiales realizan las acciones directamente, como **hal-storage-mount** para montar medios o **hal-storage-eject** para extraerlos.

Archivos de información de dispositivos: fdi

Las propiedades de HAL para estos dispositivos se manejan mediante archivos de información de dispositivo (fdi) en los directorios `/usr/share/hal/fdi` y `/etc/hal/fdi`.

El directorio `/usr/share/hal/fdi` se utiliza para configuraciones proporcionadas por la distribución, mientras que `/etc/hal/fdi` se utiliza para establecer configuraciones administrativas. En ambos se incluyen subdirectorios para los diferentes tipos de información que administra HAL, como **policy**, cuyos subdirectorios tienen archivos con directivas para la administración de dispositivos. Los archivos, conocidos como archivos de información de dispositivo, tienen reglas para obtener información acerca de dispositivos, además de detectar y asignar opciones para dispositivos extraíbles. Los archivos de información de dispositivo tienen la extensión **.fdi**, como en **storage-methods.fdi**. Por ejemplo, el directorio **policy** tiene dos subdirectorios: **10osvendedor** y **20thirdparty**. **10osvendedor** almacena los archivos fdi que tienen reglas de directivas para administrar dispositivos extraíbles en Linux (**10osvendedor** reemplaza a **90defaultpolicy** en versiones anteriores de HAL). Este directorio mantiene el archivo de directiva **20-storage-methods.fdi** que se utiliza para dispositivos de almacenamiento. Aquí encontrará las propiedades que especifican opciones para dispositivos de almacenamiento extraíbles como CD-ROM. El directorio comienza con números; los números menores se leen primero. A diferencia de udev, la última propiedad leída invalida cualquier configuración de propiedad anterior, así que se da prioridad a los directorios con números mayores y los archivos fdi que almacenan. Por esto, las directivas predeterminadas son **10osvendedor**, mientras que las directivas de usuario, que invalidan las predeterminadas, están en un directorio con número mayor, como **30user**, al igual que las políticas de terceros, **20thirdpolicy**.

En la actualidad existen tres directorios de archivos de información de dispositivo configurados en los directorios de archivo de información de dispositivo, cada uno con diferentes tipos de información: **information**, **policy** y **preprobe**:

- **Information** Para información acerca de dispositivos.
- **Policy** Para configurar directivas como las de almacenamiento. Las directivas predeterminadas para un dispositivo de almacenamiento están en un archivo **20-storage-methods.fdi**, del directorio **policy/10osvendedor**.
- **Preprobe** Maneja dispositivos difíciles, como unidades inusuales o configuraciones de unidades, como los que están en **preprobe/10osvendedor/10-ide-drives.fdi**. Contiene información que es necesaria incluso antes de que se explore el dispositivo.

Dentro de estos subdirectorios se encuentran todavía otros subdirectorios que indican de dónde vienen los archivos de información de dispositivo, como **vendedor**, **thirdparty** o **user**, y su prioridad. Ciertos archivos críticos se presentan aquí:

- **information/10freedesktop** Información proporcionada por **freedesktop.org**.
- **policy/10osvendedor** Directivas predeterminadas (se configuran por el administrador de sistema y el sistema operativo de la distribución).
- **preprobe/10usevendedor** Directivas **preprobe** para dispositivos difíciles.

Propiedades

La información de un dispositivo se especifica con una entrada *property*. Estas entradas constan de un par clave/valor, donde la clave especifica el dispositivo y su atributo y el valor es el que corresponde a ese atributo. Existen varios tipos de valores, como verdadero/falso booleano, valores de cadena, como los que se utilizan para especificar puntos de montaje de directorio o valores enteros.

Las propiedades se clasifican de acuerdo con metadatos, conexiones físicas, funciones y directivas. Los metadatos proporcionan información general acerca de un dispositivo, como el bus que utiliza, su controlador o su ID de HAL. Las propiedades de metadatos comienzan con la clave **info**, como en **info.bus**. Las propiedades físicas describen conexiones físicas, sobre todo los buses utilizados. La información de bus IDE, PCI y SCSI se presenta en las claves **ide**, **pci** y **scsi**. Las propiedades **usb_device** se utilizan para el bus USB; un ejemplo es **usb-device.number**.

Las propiedades funcionales aplican a tipos específicos de dispositivos. Aquí encontrará propiedades para dispositivos de almacenamiento, como las claves **storage.cdrom** que especifican si un dispositivo óptico tiene capacidad de escritura. Por ejemplo, si se asigna a la clave **storage.cdrom.cdr** el valor **true** se especificará que una unidad óptica graba discos CD-R.

Las directivas no son propiedades en sí. Indican la manera en que se manejarán los dispositivos. Son, en efecto, las directivas que utilizarán los programas de llamada para llevar a cabo tareas. Las directivas para medios de almacenamiento se mantienen al utilizar propiedades **Volume**, que especifican los métodos (las llamadas) que servirán para utilizar y validar opciones, como las opciones de montaje. HAL utiliza secuencias de comandos en el directorio **/usr/share/hal/scripts** para administrar realmente medios. Las siguientes entradas abreviadas vienen del archivo de directiva **20-storage-methods.fdi**. La primera especifica la acción que habrá de tomarse y la segunda la secuencia de comandos de llamado que habrá de ejecutarse, **hal-storage-mount**.

```
<append key="Volume.method_names" type="strlist">Mount</append>
<append key="Volume.method_execpaths" type="strlist">hal-storage-mount</append>
```

Se designan las opciones de montaje al utilizar **volume.mount.valid_options**, como se muestra aquí para **ro** (Read Only, sólo lectura). Las opciones que se utilizarán se determinarán cuando se ejecute la llamada al montaje.

```
<append key="volume.mount.valid_options" type="strlist">ro</append>
```

En la tabla 31-6 se presentan varias de las propiedades de directiva de volumen de uso común.

Propiedad	Descripción
volume.method.execpath	Secuencia de comandos de llamada para que la ejecute un dispositivo.
volume.policy.desired_mount_point (string)	El punto de montaje preferido para el dispositivo de almacenamiento.
volume.mount.valid_options.* (bool)	Opciones de montaje para utilizarse para un dispositivo específico, donde * es cualquier opción de montaje, como noauto o exec .
volume.method_names	Acción que se tomará.
volume.policy.mount_filesystem (string)	Sistema de archivos que se utilizará cuando se monte un volumen.
volume.mount.valid.mount_options.* (bool)	Opciones de montaje predeterminadas para volúmenes, donde * es cualquier opción de montaje, como noauto o exec .

TABLA 31-6 Dispositivos de almacenamiento de HAL

Directivas de archivo de información de dispositivo

Las propiedades se definen en directivas presentes en archivos de información de dispositivo. Como se observa, estos archivos tienen extensiones **.fdi**. Una directiva se encierra entre picoparéntesis. Existen tres directivas:

- **merge** Combina una nueva propiedad en una base de datos de información del dispositivo.
- **append** Adjunta o modifica una propiedad para ese dispositivo que ya está en la base de datos.
- **match** Prueba los valores de información de dispositivo.

Una directiva incluye un tipo de atributo que designa el tipo de valor que se almacena, como string, bool, init y double. El tipo **copyy_property** copia una propiedad. En el siguiente análisis del archivo **storage-methods.fdi** se muestran varios ejemplos de directivas merge y match.

storage.fdi

El archivo **20-storage-methods.fdi**, en el directorio **/usr/share/hal/fdi/policy/10osvendor**, incluye las directivas para sus dispositivos de almacenamiento extraíbles. Aquí es donde realmente se especifican sus opciones para las entradas de volúmenes de almacenamiento (por ejemplo, CD-ROM). El archivo se organiza en secciones que comienzan con tipos particulares de dispositivos para opciones predeterminadas estándar. Las claves se utilizan para definir opciones, como **volume.mount.valid_options**, que especificará una opción de montaje para un dispositivo de almacenamiento, como un CD-ROM. Las claves también se utilizan para especificar excepciones, como dispositivos de conexión activa.

El archivo **20-storage-methods.fdi** comienza con las propiedades predeterminadas y después incluye las de tipos específicos de dispositivos. A menos que después se vuelvan a definir en una clave, permanecerá aplicada la predeterminada. Las opciones que verá para los volúmenes de almacenamiento predeterminados aplicarán a CD-ROMs. Por ejemplo, la opción **noexec** se establece como valor predeterminado. Lo siguiente establece **noexec** como la opción de montaje predeterminada para un dispositivo de almacenamiento. También existen entradas para **ro** y **quiet**. La operación **append** agrega la opción de directiva.

```
<append key="volume.mount.valid_options" type="strlist">noexec</append>
```

Ahora el directorio raíz del punto de montaje predeterminado para dispositivos de almacenamiento se establece mediante la secuencia de comandos de llamada a montaje, **hal-storage-mount**. Actualmente es **/media**. El punto de montaje predeterminado es **disk**. HAL intentará utilizar la información de propiedad Volume para generar un punto de montaje.

El siguiente ejemplo muestra la administración de discos en blanco. En vez de que se les monte, sólo se expulsan estos discos. Para determinar las acciones posibles, HAL utiliza **method_names**, **method_signatures** y **method_execpaths** para las propiedades de Volume (se ha eliminado de este ejemplo el prefijo **org.freedesktop.Hal** para hacerlo más legible, como en **org.freedesktop.Hal.Volume.method_names**).

```
<match key="volume.disc.is_blank" bool="true">
<append key="info.interfaces" type="strlist">Volume</append>
<append key="Volume.method_names" type="strlist">Eject</append>
<append key="Volume.method_signatures" type="strlist">as</append>
<append key="Device.Volume.method_execpaths" type="strlist">hal-storage-eject<
/append>
</match>
```

Después de tratar con casos especiales, se buscan los dispositivos de sistema de archivos de la siguiente manera:

```
<match key="volume.fsusage" string="filesystem">
```

También se especifican los dispositivos de almacenamiento que habrán de ignorarse, como `vfat`.

```
<merge key="volume.ignore" type="bool">false</merge>
```

Después se especifican las acciones que habrán de tomarse y las secuencias de comandos de llamada que se utilizarán, como la empleada para `Unmount`, que utiliza **hal-storage-mount**.

```
<append key="Device.Volume.method_names" type="strlist">Mount</append>
<append key="Device.Volume.method_signatures" type="strlist">ssas</append>
<append key="Device.Volume.method_execpaths" type="strlist">hal-storage-mount<
/append>
```

Luego se especifican las opciones con **volume.mount.valid_options**, que comienzan con las predeterminadas y continúan con los casos especiales, como **ext3** que se muestra aquí:

```
<!-- allow these mount options for ext3 -->
<match key="volume.fstype" string="ext3">
<append key="volume.mount.valid_options" type="strlist">data=</append>
</match>
```

Llamadas de HAL

Las *llamadas* son programas que se invocan cuando el objeto de dispositivo se modifica o cuando un dispositivo cambia. Como tales, las llamadas se utilizan para mantener directivas en todo el sistema (que pueden ser específicas del sistema operativo) como cambiar permisos en nodos de dispositivo, administrar dispositivos extraíbles o configurar el subsistema de red. Hay tres tipos diferentes de llamadas para dispositivos, capacidades y propiedades. Las llamadas *device* se ejecutan cuando se agrega o extrae un dispositivo. Las llamadas *capability* agregan o eliminan capacidades de dispositivo, y las *property* agregan o eliminan una propiedad del dispositivo. En la publicación actual, las llamadas se implementan al utilizar las reglas de propiedad `info.callout`, como quien invoca al llamado **hal-storage-mount** cuando se insertan o extraen CD/DVD-ROM, como se muestra aquí:

```
<append key="org.freedesktop.Hal.Device.Volume.method_execpaths"
type="strlist">hal-storage-mount</append>
```

Las llamadas se colocan en el directorio `/usr/libexec` con las llamadas de HAL con un prefijo **hal-**. Aquí encontrará muchas llamadas a almacenamiento utilizadas por HAL como **hal-storage-eject** y **hal-storage-mount**. HAL utiliza estas llamadas para administrar dispositivos extraíbles como DVD/CD-ROM directamente, en vez de editar entradas en el archivo `/etc/fstab` (**fstab-sync** ya no se utiliza). La herramienta **gnome-mount** que se utiliza para montar discos CD/DVD en el escritorio GNOME utiliza los llamados HAL. Otras secuencias de comandos soportadas se encuentran en el directorio `/usr/lib/hal/scripts`.

Dispositivos manuales

Todavía es necesario crear varios dispositivos de forma manual: por ejemplo, los puertos paralelos de impresora. Casi todos estos dispositivos ya están configurados con **MAKEDEV** y los archivos `/etc/makedev.d`. Para que `udev` cree estos dispositivos, sus nombres se colocan en archivos de

configuración en el directorio `/etc/udev/makedev.d`. El archivo **50-udev.nodes** contiene una lista de nombres de dispositivo que udev utilizará con **MAKEDEV** para construir de forma manual cuando udev genere el directorio de dispositivos `/dev`. Aquí, encontrará entradas para puertos paralelos como **parport0** a **parport3**.

Si quiere, puede crear interfaces de archivos de dispositivos de forma manual al utilizar los comandos **MAKEDEV** o **mknod**. Para que udev los agregue al directorio `/dev`, colóquelos en `/etc/udev/devices`; udev los copiará en el directorio `/dev` cuando los genere. En el caso de algunos dispositivos, como ISDN, tal vez tenga que hacer esto. En el siguiente ejemplo se crea un dispositivo ISDN con **MAKEDEV** y se coloca en el directorio `/etc/udev/devices`:

```
/sbin/MAKEDEV -d /etc/udev/devices isdn
```

Tipos de dispositivo

Linux implementa varios tipos de dispositivos; de ellos, los más comunes son de bloque y carácter. Un *dispositivo de bloque*, como un disco duro, transmite datos bloque por bloque. Un *dispositivo de carácter*, como una impresora o módem, transmite datos de carácter en carácter, o más bien como un flujo de datos, no como bloques separados. Los archivos de controlador de dispositivo para dispositivos de carácter tienen una *c* como primer carácter en el segmento de permisos desplegado por el comando **ls**. Los archivos de controlador de dispositivo para dispositivos de bloque tienen una *b*. En el siguiente ejemplo, **lp0** (la impresora) es un dispositivo de carácter y **sda1** (el disco duro) es un dispositivo de bloque:

```
# ls -l sda1 lp0
brw-rw---- 1 root disk 3, 1 Jan 30 02:04 sda1
crw-rw---- 1 root lp    6, 0 Jan 30 02:04 lp0
```

El tipo de dispositivo puede ser *b*, *c*, *p* o *u*. Como ya se mencionó, la *b* indica un dispositivo de bloque y *c* uno de carácter. La *u* es para un dispositivo de carácter sin búfer y la *p* para un dispositivo FIFO (First In, First Out: primero en entrar, primero en salir). Los dispositivos del mismo tipo a menudo tienen el mismo nombre; por ejemplo, todas las interfaces seriales tienen el nombre **ttyS**. Entonces los dispositivos del mismo tipo se identifican de forma única con un número adjunto al nombre. Este número tiene dos componentes: el número mayor y el menor. Los dispositivos pueden tener el mismo número mayor, pero en ese caso el número menor siempre es diferente. Esta estructura de números mayor y menor está diseñada para tratar con situaciones en que varios dispositivos son dependientes de uno más grande, como varios módems conectados a la misma tarjeta de entrada/salida. Todos los módems tendrán el mismo número mayor que hace referencia a la tarjeta, pero cada módem tendrá un número menor único. Ambos números, el menor y el mayor, se requieren para dispositivos de bloque o carácter (*b*, *c* y *u*). Sin embargo, no se utilizan para dispositivos FIFO.

Los nombres de dispositivo válidos, junto con sus números mayores o menores, se incluyen en el archivo **devices.txt** ubicado en el directorio `/Documentation` para el código fuente de kernel, `/usr/src/linux-ver/Documentation`. Cuando se crea un dispositivo, se utilizan números mayores y menores, además del prefijo de nombre de un tipo particular de dispositivo que se está creando. Ya casi todos estos dispositivos están creados y se encuentran en el directorio `/etc/dev`.

MAKEDEV

Se utiliza **MAKEDEV** para crear archivos de dispositivo. **MAKEDEV** utiliza archivos de configuración de dispositivos ubicados en el directorio `/etc/makedev.d` para determinar opciones de dispositivos, como los números mayores y menores del dispositivo o cualquier vínculo simbólico que debe

crearse para éste. Por ejemplo, el archivo `/etc/makedev.d/01sound` incluye dispositivos de sonido. Un archivo de configuración **MAKEDEV** tiene tres tipos diferentes de registros; cada uno comienza con un operador diferente.

- **b o c** Crea un dispositivo de bloque (b) o carácter (c). Estas entradas almacenan varias opciones: modo (permisos), propietario, grupo, números mayor y menor, inc, conteo (número de dispositivos creado) y ftn (el nombre del dispositivo). La opción ftn es, técnicamente, una cadena de formato que incluye un especificador de formato para incrementar de forma numérica nombres de dispositivos similares, como `cdrom%d` para `cdrom0`, `cdrom1`, etc. La opción inc establece el incremento.
- **l** Crea un vínculo simbólico de un dispositivo.
- **a** Un alias que aplica los comandos utilizados por un dispositivo a otro. Esto le permite crear un dispositivo de sonido, que a cambio crea automáticamente dispositivos de audio, MIDI y mezcladora, etc.

En el archivo `/etc/makedev.d/01sound`, existen varias entradas de alias para sonido, como la siguiente:

```
a sound audio
```

Una entrada de vínculo creará un vínculo simbólico llamado **audio0** para el archivo de dispositivo de audio.

```
l audio0 audio
```

La creación real del archivo de dispositivo de sonido se configura en el archivo **alsa**. Los dispositivos de sonido utilizan ALSA. Aquí encontrara varias entradas c con permisos, propietario, valores de grupo, etc.

Con gran parte de las opciones manejadas en los archivos de configuración de dispositivo **MAKEDEV**, el comando para crear dispositivos es muy simple. Sin embargo, tenga en cuenta que con **udev**, los archivos de dispositivo no se crean en `/dev`. **udev** regenera este directorio automáticamente. Para que **udev** coloque su archivo de dispositivo en `/dev` cuando lo genera, ubique el archivo de dispositivo que hizo en `/etc/udev/devices`. Utilice la opción **-d** para especificar el directorio de dispositivo **udev**. Lo siguiente crea un dispositivo ISDN:

```
MAKEDEV -d /etc/udev/devices isdn
```

mknod

Aunque el comando **MAKEDEV** es preferible para crear archivos de dispositivo, sólo crea archivos para los que está configurado. En el caso de dispositivos que no están configurados para que se utilicen con **MAKEDEV**, tendrá que recurrir al comando **mknod**. Se trata de un comando de nivel bajo que requiere configuración manual de todas las opciones. Con el comando **mknod** se crea un archivo de dispositivo en el modo tradicional, sin nada del soporte de configuración que proporciona **MAKEDEV**.

El comando **mknod** crea un dispositivo de tipo bloque o carácter. El comando **mknod** tiene la siguiente sintaxis:

```
mknod opciones dispositivo tipo-dispositivo num-mayor num-menor
```

Como **MAKEDEV** cubre fácilmente casi todos los dispositivos, además de que **udev** los genera de forma automática, en raras ocasiones, o nunca, tendrá que utilizar **mknod**. Como un ejemplo simple,

aquí se analiza la creación de un archivo de dispositivo con **mknod** para un puerto de impresora. Los sistemas de Linux suelen proporcionar archivos de dispositivo para puertos de impresora (**lp0-2**). Como ejemplo, puede ver la manera en que puede crearse manualmente un puerto adicional con el comando **mknod**. Los de impresora son dispositivos de carácter y deben pertenecer al root y al daemon. El propietario y el grupo, 660, leen y escriben los permisos para dispositivos de impresora. El número de dispositivo mayor se establece en 6, mientras el de dispositivo menor será el número de puerto de la impresora, como 0 para LPT1 y 1 para LPT2. Una vez que se crea el dispositivo, se utiliza **chown** para cambiar su propietario al usuario **root**, porque sólo el administrador debe controlarlo. Cambie el grupo a **lp** con el comando **chgrp**.

Casi todos los dispositivos pertenecen a sus propios grupos, como **disks** para particiones de disco duro, **lp** para impresoras, **floppy** para discos flexibles y **tty** para terminales. En el siguiente ejemplo, un dispositivo de impresora se crea en un cuarto puerto paralelo, **lp3**. La opción **-m** especifica los permisos (en este caso, 660). El dispositivo es de carácter, como lo indica el argumento **c** después del nombre de dispositivo. El número mayor es 6, y el menor es 3. Si estuviera generando un dispositivo en **lp4**, el número mayor aún sería 6, pero el menor sería 4. Una vez que se cree el dispositivo, el comando **chown** cambia el propietario del dispositivo de impresora paralelo a **root**. En el caso de impresoras, asegúrese de que se ha creado el directorio spool para su dispositivo. Si no, necesita generar uno. Los directorios spool contienen archivos para datos que varían de acuerdo con la salida o entrada del dispositivo, como en el caso de impresoras o escáneres.

Al igual que con todos los dispositivos manuales, el archivo de dispositivo tiene que colocarse en el directorio **/etc/udev/devices**; udev lo pondrá después en **/dev**.

```
# mknod -m 660 /etc/udev/devices/lp3 c 6 3
# chown root /etc/udev/devices/lp3
# chgrp lp /etc/udev/devices/lp3
```

Instalación y administración de terminales y módems

En Linux, varios usuarios pueden estar en una sesión al mismo tiempo. Cada uno necesita su propia terminal para acceder al sistema Linux, por supuesto. El monitor en su PC actúa como una terminal especial, llamada *consola*, pero se agregan otras terminales a través de los puertos seriales de su PC o de una tarjeta especial con varios puertos instalada en su PC. Las otras terminales o PC son independientes y utilizan programas de emulación de terminal. Para conocer una explicación detallada de la instalación de terminales, consulte el archivo **Term_HOWTO** en **/usr/share/doc/HOWTO** o en el sitio de Linux Documentation Project (**tldp.org**). Aquí se proporciona una explicación breve.

Puertos seriales

A los puertos seriales de su PC se les conoce como COM1, COM2, COM3 y COM4. Estos puertos seriales corresponden a los dispositivos de terminal de **/dev/ttyS0** a **/dev/ttyS3**. Tome en cuenta que tal vez ya los estén usando otros dispositivos de entrada como su ratón y otros dispositivos de comunicación como su módem. Si tiene una impresora serial, uno de estos dispositivos seriales ya estará en uso. Si instala una tarjeta de varios puertos, tiene muchos puertos más para escoger. Para cada terminal que se agrega, udev creará el dispositivo de carácter apropiado en su sistema Linux. Los permisos para un dispositivo de terminal suelen ser normalmente 660. Los *dispositivos de terminal* son dispositivos de caracteres con un número mayor de 4 y números menores que suelen comenzar en 64.

SUGERENCIA La entrada `/dev/pts` en el archivo `/etc/fstab` monta un sistema de archivos `devpts` en `/dev/pts` para Pseudo-TTY de Unix 98. Estas pseudoterminales se identifican con dispositivos numerados.

mingetty, mgetty yagetty

Su sistema administra los dispositivos de terminal mediante el programa **getty** y un conjunto de archivos de configuración. Cuando su sistema inicia, lee una lista de terminales conectadas en el archivo **inittab** y después ejecuta un programa **getty** apropiado para cada uno, ya sea **mingetty**, **mgetty** o **agetty**. Estos programas **getty** configuran comunicaciones entre su sistema Linux y terminales específicas. **mingetty** proporciona soporte mínimo a consolas virtuales, mientras que **agetty** proporciona soporte mejorado a conexiones de terminal. **agetty** también incluye parámetros para la cantidad de baudios y el intervalo entre éstos. **mgetty** está diseñado para conexiones de fax/módem, lo que le permite configurar parámetros de marcación, inicio de sesión y fax. Los archivos de configuración **mgetty** se almacenan en el directorio `/etc/mgetty+sendfax`. La información de conexión de módem se almacena en el archivo `/etc/mgetty+sendfax/mgetty.conf`. Todos los programas **getty** leen un mensaje inicial colocado en el archivo `/etc/issue`, que contiene códigos especiales para proporcionar el nombre de sistema y la fecha y hora actuales.

Archivos termcap e inittab

El archivo `/etc/inittab` almacena instrucciones para su sistema relacionados con la manera de administrar dispositivos de terminal. Una línea en el archivo `/etc/inittab` tiene cuatro componentes básicos: un ID, un nivel de ejecución, una acción y un proceso. Los dispositivos de terminal se identifican con números de ID, que comienzan con 1 para el primer dispositivo. El nivel de ejecución al que opera la terminal suele ser 1. La acción suele ser *respawn*, que significa que se ejecutará el proceso continuamente. El proceso es una llamada a **mingetty**, **mgetty** o **agetty** con el nombre de dispositivo de terminal. El archivo `/etc/termcap` almacena las especificaciones para diferentes tipos de terminales. Se trata de los diferentes tipos de terminal que usan los usuarios para iniciar sesión en su sistema. Su archivo `/etc/termcap` ya está lleno con especificaciones para casi todas las terminales producidas actualmente. Una entrada en el archivo `/etc/termcap` consta de varios nombres que se utilizan para una terminal separada por una línea vertical (|) y después una serie de especificaciones de parámetro, cada una terminada en dos puntos. Aquí encontrará el nombre utilizado para un tipo específico de terminal. Se utiliza **more** para desplegar su archivo `/etc/termcap`, y después se utiliza una búsqueda, `/`, para ubicar el tipo de terminal. Se establecen muchas opciones para un dispositivo de terminal. Para cambiar estas opciones, se utiliza el comando **stty** en lugar de cambiar directamente los archivos de configuración. El comando **stty** sin argumentos despliega la configuración actual de la terminal.

tset

Cuando un usuario inicia sesión, resulta útil tener el dispositivo de terminal inicializado con el comando **tset**. Por lo general, el comando **tset** se coloca en el archivo `.bash_profile` del usuario y se ejecuta automáticamente siempre que el usuario inicia sesión en el sistema. Se utiliza el comando **tset** para configurar el tipo de terminal y cualquier otra opción que requiera el dispositivo de terminal. A continuación se presenta una entrada común de **tset** para un archivo `.bash_profile`. La opción **-m dialup**: pide al usuario insertar un tipo de terminal. El tipo especificado aquí es uno predeterminado que se despliega entre paréntesis. El usuario oprime ENTER para seleccionar el predeterminado. La petición se ve como ésta: **TERM=(vt100)?**.

```
eval `tset -s -Q -m dialup:?vt00`
```

Dispositivos de entrada

Los dispositivos de entrada, como los ratones y teclados, se despliegan en varios niveles. La detección inicial se lleva a cabo durante la instalación, donde selecciona los tipos de ratón y teclado. HAL detectará automáticamente el teclado y el ratón. Puede realizar una configuración detallada con sus herramientas de configuración de escritorio, como las herramientas de configuración de ratón de GNOME o KDE. En GNOME, seleccione Sistema | Preferencias | Hardware | Ratón para configurar su ratón. Existe una entrada Teclado en el mismo menú para teclados.

Instalación de tarjetas de sonido, red y otras

Para instalar una nueva tarjeta, primero debe configurarse su kernel para que sea compatible con ella. El soporte para la mayoría de las tarjetas se proporciona en forma de módulos que se cargan de forma dinámica en el kernel. La instalación de soporte para una tarjeta suele ser sólo cuestión de cargar un módulo que incluye los controladores. Por ejemplo, los controladores para la tarjeta de sonido Sound Blaster están en el módulo **sb.o**. Al cargar este módulo, Linux puede acceder a su tarjeta de sonido. Casi todas las distribuciones de Linux detectan automáticamente las tarjetas instaladas en su sistema y cargan los módulos necesarios. Si cambia las tarjetas de sonido, la nueva tarjeta se detecta automáticamente. También se cargan de forma manual módulos que necesita, al eliminar los viejos que crean conflictos. En la sección “Módulos”, que se encuentra más adelante, en este capítulo, se describe este proceso.

Los archivos de dispositivo para casi todas las tarjetas ya están configurados en el directorio **/dev** por **udev**. Por ejemplo, el nombre de dispositivo para su tarjeta de sonido es **/dev/audio**. Sin embargo, los nombres de dispositivo para tarjetas de red son alias de módulos de red, en lugar de archivos de dispositivo. Por ejemplo, el nombre de dispositivo de su tarjeta Ethernet comienza con **eth**, y la numeración con **0**, como en **eth0** para la primera tarjeta Ethernet de su sistema. Se utiliza un alias para hacer referencia al módulo utilizado para esa tarjeta en particular; por ejemplo, una tarjeta Etherlik 3Com coloca alias en el módulo de red 3c59x, cuyo alias sería **eth0** si es la primera tarjeta Ethernet. Los módulos por sí solos se almacenan en el módulo del kernel ubicado en **/lib/modules**, como se describe en la sección “Archivos y directorios de módulo: **/lib/modules**”.

Dispositivos de sonido

Su distribución de Linux suele proporcionar una herramienta para configurar la mayor parte de las tarjetas de sonido. Ahora, **udev** y **HAL** detectan y administran casi todas las tarjetas de sonido. En la tabla 31-7 se presenta una lista de los diferentes dispositivos de sonido. Tal vez algunas tarjetas

Dispositivo	Descripción
/dev/sndstat	Estado de controlador de sonido
/dev/audio	Dispositivo de salida de audio
/dev/dsp	Dispositivo de sampleo de sonido
/dev/mixer	Mezcladora de control en la tarjeta de sonido
/dev/music	Secuenciador de alto nivel
/dev/sequencer	Secuenciador de bajo nivel
/dev/midi	Puerto MIDI directo

TABLA 31-7 Dispositivos de sonido

de sonido requieran un soporte más especializado. Puede saber cuál es la configuración de sonido actual al desplegar el contenido del archivo `/proc/asound/oss/sndstat`. Se prueba la tarjeta de sonido con sólo redirigir un archivo de sonido a ésta, como se muestra aquí:

```
cat sample.au > /dev/audio
```

Para el kernel 2.4, casi todos los controladores de sonido de Linux se desarrollaron como parte del sistema de sonido abierto (OSS, Open Sound System) y se distribuyen de manera gratuita como OSS/Free. Éstos se instalan como parte de las distribuciones de Linux. Los controladores de dispositivo OSS están hechos para proporcionar un API uniforme para todas las plataformas Unix, incluido Linux. Se da soporte a tarjetas de sonido compatibles con Sound Blaster (y Windows Sound System), ISA y PCI. OSS también está disponible por un precio nominal y presenta interfaces de configuración para instalación de dispositivo.

Advanced Linux Sound Architecture (ALSA) reemplazó a OSS en el kernel de Linux 2.6; apunta a ser una mejor opción que OSS, mientras mantiene compatibilidad con éste. ALSA proporciona un controlador de sonido modular, una API, y un administrador de configuración ALS es un proyecto GNU, y es completamente gratis; su sitio Web en alsa-project.org contiene documentación extensa, aplicaciones y controladores. Actualmente está disponible el controlador de sonido ALSA, ALSA Kernel API, la biblioteca de ALSA para dar soporte a desarrollo de aplicaciones, y el administrador de ALSA, para proporcionar una interfaz de configuración para el controlador. ALSA evolucionó del proyecto Linux Ultra Sound Project. El proyecto ALSA actualmente da soporte a casi todas las tarjetas de sonido de Creative.

La interfaz digital de instrumentos musicales (MIDI, Musical Instrument Digital Interface) y Sound Pages de Linux, actualmente en linux-sound.org, incluyen vínculos con sitios para MIDI y software de audio de Linux.

Dispositivos de video y televisión

Los nombres de dispositivo que se utilizan para dispositivos de televisión, video y DVD aparecen en la tabla 31-8. Los controladores para decodificadores de DVD y televisión ya están desarrollados, y [mga4linux \(marvel.sourceforge.net\)](http://mga4linux.marvel.sourceforge.net) está desarrollando soporte de video para las tarjetas Matrox Multimedia. General ATI TV and Overlay Software (GATOS) (gatos.sourceforge.net) ha desarrollado controladores para las características de tarjetas de video ATI que no cuentan con soporte, específicamente las características de televisión. El proyecto BTTV Driver Project ha desarrollado controladores para el chip de video Booktree. Creative Labs patrocina controladores de Linux para la línea Creative de decodificadores DVD DXR2 (opensource.creative.com).

Nombre de dispositivo	Tipo de dispositivo
<code>/dev/video</code>	Interfaz de captura de video
<code>/dev/vfx</code>	Interfaz de efectos de video
<code>/dev/codec</code>	Interfaz de codec de video
<code>/dev/vout</code>	Interfaz de salida de video
<code>/dev/radio</code>	Dispositivos de radio AM/FM
<code>/dev/vtx</code>	Chips de interfaz de teletexto
<code>/dev/vbi</code>	Interfaz de servicios de datos

TABLA 31-8 Controladores de dispositivo de video y televisión

Dispositivos PCMCIA

Los dispositivos PCMCIA son lectores de tarjetas que suelen encontrarse en computadoras portátiles para conectar dispositivos como módems o tarjetas inalámbricas, aunque también se han vuelto un estándar en muchos sistemas de escritorio. El mismo dispositivo PCMCIA da soporte a muchos tipos diferentes de dispositivos, incluidas las tarjetas de red, módems, discos duros y dispositivos Bluetooth.

HAL y udev ahora administran el soporte a PCMCIA; ya no se utiliza el servicio `cardmgr/pcmcia`. Ahora los dispositivos PCMCIA se consideran de conexión activa administrados directamente por HAL y udev. Ahora `pccardctl` maneja la información de tarjeta y control. Las reglas udev de PCMCIA se incluyen en `60-pcmcia.rules`, que examina e instala automáticamente las tarjetas. Revise la página kernel.org/pub/linux/utils/kernel/pcmcia/pcmcia.html para conocer más información.

Puede obtenerse información acerca de un dispositivo PCMCIA con el comando `pccardctl`, además de expulsar e insertar manualmente un dispositivo. Las opciones `status`, `config` e `ident` desplegarán el estado del conector del dispositivo y la configuración y la identificación de este último. Las opciones `insert` y `eject` le permitirán agregar y eliminar un dispositivo. El comando `cardinfo` también proporciona información de dispositivo.

No es aconsejable usar conexión activa con dispositivos IDE o SCSI. Para éstos, primero debe apagar manualmente el dispositivo con el comando `pccardctl`.

```
pccardctl eject
pccardctl scheme home
```

Módulos

El kernel de Linux emplea módulos para dar soporte a diferentes características de sistemas operativos, incluido el que se da a varios dispositivos como tarjetas de sonido y red. En muchos casos, tiene la opción de implementar soporte a un dispositivo como módulo o mediante la compilación directa como característica integrada de kernel, que requiere que reconstruya éste. Una solución más segura es utilizar módulos. Los *módulos* son componentes del kernel de Linux que se cargan de acuerdo con las necesidades. Para agregar soporte a un nuevo dispositivo, ahora simplemente se instruye al kernel que cargue un módulo para ese dispositivo. En algunos casos, tal vez tenga que recopilar sólo ese módulo para proporcionar soporte a su dispositivo. El uso de módulos tiene la ventaja agregada de reducir el tamaño del programa de kernel, además de hacer su sistema más estable. El kernel carga los módulos en la memoria sólo cuando es necesario. Si un módulo falla, sólo se detiene la ejecución de éste, lo que no afecta a todo el sistema. Por ejemplo, el módulo para la interfaz de red PPP utilizada para un módem sólo necesita utilizarse cuando se conecta a un ISP.

Herramientas de módulo de kernel

Los módulos que su sistema necesita suelen determinarse durante la instalación, de acuerdo con el tipo de información de configuración que proporciona y la detección automática realizada por su distribución de Linux. Por ejemplo, si su sistema utiliza una tarjeta Ethernet cuyo tipo se especifica durante la instalación, el sistema carga el módulo para esa tarjeta. Sin embargo, puede controlar de forma manual cuáles módulos cargará su sistema. En efecto, esto le permite personalizar su kernel de la forma que quiera. Se utilizan varios comandos, herramientas de configuración y daemons para administrar módulos de kernel. El kernel de Linux 2.6 incluye el cargador de módulo de kernel (Kmod, Kernel Module Loader), que tiene la capacidad de cargar módulos automáticamente,

de acuerdo con las necesidades. El módulo de kernel que carga el soporte también debe habilitarse en el kernel, aunque esto suele considerarse parte de la configuración estándar. Además, varias herramientas utilizan ciertos comandos de kernel para realizar las tareas de cargar o descargar módulos de forma manual, si se prefiere. El cargador de módulo de kernel usa ciertos comandos de kernel para realizar la tarea de cargar y descargar módulos. El comando **modprobe** es de propósito general y llama a **insmod** para cargar módulos y a **rmmod** para descargarlos. Estos comandos se muestran en la tabla 31-9. Las opciones para módulos particulares, configuración general e incluso carga de módulos específicos se detallan en el archivo **/etc/modprobe.conf**. Se utiliza este archivo para cargar y configurar módulos automáticamente. También puede especificar que se carguen módulos en el indicador de comandos de arranque o en **grub.conf**.

Archivos y directorios de módulo: /lib/modules

El nombre de archivo de un módulo tiene la extensión **.o**. Los módulos de kernel residen en el directorio **/lib/modules/versión**, donde *versión* es el número de versión de su kernel actual, con la extensión FC7. El directorio para el kernel 2.6.20-1.2054_FC7 es **/lib/modules/2.6.20-1.2054_FC5**. Conforme instala nuevos kernel en su sistema, se generan nuevos directorios de módulos para éstos. Un método para acceder al directorio del kernel actual consiste en utilizar el comando **uname -r** para generar el número de versión del kernel. Este comando necesita comillas invertidas.

```
cd /lib/modules/`uname -r`
```

En este directorio, los módulos del kernel residen en el directorio **/kernel**. Dentro del directorio **/kernel** existen varios subdirectorios, incluido el directorio **/drivers** que almacena subdirectorios de módulos como controladores de sonido o de video. Estos subdirectorios sirven para ordenar sus módulos en categorías, facilitando su localización. Por ejemplo, el directorio **kernel/drivers/net** almacena módulos para sus tarjetas Ethernet, y el directorio **kernel/drivers/sound** contiene los módulos de tarjeta de sonido.

SUGERENCIA Observará que no hay entradas para los dispositivos Ethernet en el archivo **ldev**, como **eth0** o **eth1**. Esto se debe a que en realidad son alias de los módulos de kernel definidos en el archivo **letc/modprobe.conf**, o dispositivos que el kernel maneja directamente. No son archivos de dispositivo.

Comando	Descripción
lsmod	Presenta una lista de módulos cargados.
insmod	Carga un módulo en el kernel. No revisa dependencias.
rmmod	Descarga un módulo cargado. No revisa dependencias.
modinfo	Despliega información acerca de un módulo: -a (autor), -d (descripción), -p (parámetros de módulo), -f (nombre de archivo de módulo), -v (versión de módulo).
depmod	Crea un archivo de dependencia que presenta una lista de todos los demás módulos de los que puede depender el módulo especificado.
modprobe	Carga un módulo con cualquier módulo dependiente que también podría necesitar. Utiliza el archivo de lista de dependencias generado por depmod: -r (descarga un módulo), -l (presenta una lista de módulos).

TABLA 31-9 Comandos de módulo kernel

Administración de módulos con modprobe

Como ya se observó, se utilizan varios comandos para administrar módulos. El comando **lsmod** presenta una lista de los módulos cargados en el kernel, y **modinfo** proporciona información acerca de módulos particulares. Aunque se utilizan los comandos **insmod** y **rmmod** para cargar y descargar módulos directamente, sólo debe utilizar **modprobe** para estas tareas. Sin embargo, a menudo un módulo determinado requiere otros módulos para cargarse. Por ejemplo, el módulo para la tarjeta de sonido Sound Blaster, **sb.o**, requiere que también se cargue el módulo **sound.o**.

El comando depmod

En lugar de tratar de determinar manualmente de qué módulos depende uno determinado, se utiliza el comando **depmod** para detectar las dependencias. El comando **depmod** genera un archivo con una lista de todos los módulos de los que depende un módulo. El comando **depmod** genera una lista jerárquica, que indica los módulos que deben cargarse primero y en qué orden. Después, para cargar el módulo, se utiliza el comando **modprobe** con ese archivo. **modprobe** lee el archivo generado por **depmod** y carga cualquier módulo dependiente en el orden correcto, junto con el que quiere. Necesita ejecutar **depmod** con la opción **-a** una vez, antes de que utilice **modprobe**. La inserción de **depmod -a** crea una lista completa de todas las dependencias de módulo. Este comando crea un archivo llamado **modules.dep** en el directorio de módulos para su versión de kernel actual, **/lib/modules/versión**.

```
depmod -a
```

El comando modprobe

Para instalar un módulo de forma manual, se utiliza el comando **modprobe** y el nombre del módulo. Se agrega cualquier parámetro que requiera el módulo. El siguiente comando instala el módulo de sonido de alta definición de Intel. **modprobe** también es compatible con el uso del carácter ***** para permitir el empleo de un patrón para seleccionar varios módulos. En este ejemplo se utilizan varios valores comunes para tarjetas de sonido. Debe utilizar los valores recomendados para su tarjeta de sonido en su sistema. El proyecto ALSA da soporte a casi todos los controladores de tarjeta de sonido. Revise su sitio Web para conocer qué módulo de controlador se utiliza con su tarjeta.

```
modprobe snd-hda-intel
```

Para descubrir cuáles parámetros toma un módulo, se utiliza el comando **modinfo** con la opción **-p**.

La opción **-l** se utiliza para presentar una lista de los módulos; la opción **-t**, para buscar módulos en un subdirectorio específico. Los módulos de sonido se ordenan en diferentes subdirectorios, de acuerdo con la interfaz de dispositivo que utilizan, como **pci**, **isa** o **usb**. Casi todas las tarjetas de sonido internas utilizan **pci**. Dentro del directorio de la interfaz, tal vez haya más directorios, como **emu10k1**, que se utilizan para tarjetas Audigy de Creative y **hda** para controladores de alta definición. En el siguiente ejemplo, el usuario despliega una lista con todos los módulos en el directorio **sound/pci/hda**:

```
# modprobe -l -t sound/pci/hda
/lib/modules/2.6.15-1.3059_FC7/kernel/sound/pci/hda/snd-hda-intel.o
/lib/modules/2.6.15-1.3059_FC7/kernel/sound/pci/hda/snd-hda-codec.o
/lib/modules/2.6.15-1.2054_FC7/kernel/drivers/sound/sound.o
/lib/modules/2.6.15-1.2054_FC7/kernel/drivers/sound/soundcore.o
```

Las opciones para el comando **modprobe** se colocan en el archivo **/etc/modprobe.conf**. Aquí se insertan opciones de configuración, como los directorios y alias predeterminados. Un alias proporciona un nombre simple para un módulo. Por ejemplo, la siguiente entrada le permite hacer referencia al módulo de la tarjeta Ethernet **3c59x.o** como **eth0** (Kmod detecta automáticamente la tarjeta Ethernet 3Com y carga el módulo 3c59x):

```
alias eth0 3c59x
```

En sistemas Nvidia, se utiliza el módulo **forcedeth**.

```
alias eth0 forcedeth
```

El comando **insmod**

El comando **insmod** realiza la carga real de módulos. **modprobe** y Kernel Module Loader hacen uso de este comando para cargar módulos. Aunque es preferible **modprobe** porque revisa dependencias, se pueden cargar y descargar módulos particulares de forma individual con los comandos **insmod** y **rmmod**. El comando **insmod** toma como argumento el nombre del módulo, al igual que **rmmod**. El nombre de base, como **sb** para el módulo **sb.o**. Para especificar el nombre de archivo de módulo completo se utiliza la opción **-o**. Otras opciones útiles son **-p**, que le permite examinar antes su sistema para ver si el módulo se carga correctamente, y **-n**, que realiza todas las tareas, excepto cargar realmente el módulo (una ejecución simulada). La opción **-v** (verbose) enumera todas las acciones, a medida que van ocurriendo. En los casos raros en que tal vez tenga que forzar un módulo para cargarlo, se utiliza la opción **-f**. En el siguiente ejemplo, **insmod** carga el módulo **sb.o**:

```
# insmod -v sb
```

El comando **rmmod**

El comando **rmmod** realiza la descarga real de los módulos. Es el comando que utilizan **modprobe** y Kernel Module Loader para descargar módulos. Se utiliza el comando **rmmod** para eliminar un módulo particular, siempre y cuando no lo utilicen o lo necesiten otros módulos. Se elimina un módulo y todos sus módulos dependientes con la opción **-r**. La opción **-a** elimina todos los módulos que no se utilizan. Con la opción **-e**, cuando **rmmod** descarga un módulo, guarda cualquier dato persistente (parámetros) en el directorio de datos persistentes, que suele ser **/var/lib/modules/persist**.

Configuración de **modprobe**

Para cargar módulos se necesita cambiar el nombre del sistema, además de especificar las opciones que se utilizarán cuando se cargan módulos específicos. Aunque se elimine o instale un módulo, tal vez se tengan que ejecutar ciertos programas adicionales o se deban agregar otras opciones específicas. Estos parámetros se definen en un archivo **/etc/modprobe.conf** o en archivos ubicados en un directorio **/etc/modprobe.d**. La configuración de **modprobe** permite cuatro acciones: alias, options, install y remove.

- **alias** *nombre de módulo* Proporciona otro nombre para el módulo; se utiliza para dispositivos de red y sonido.
- **options** *opciones de módulo* Especifica cualquier opción que pueda necesitar un módulo.
- **install** *comandos de módulo* Utiliza los comandos especificados para instalar un módulo, permitiéndole controlar la carga del módulo.

- **remove** *comandos de módulo* Especifica comandos para ejecutarse cuando un módulo se descarga.
- **include** *archivo-config* Archivos de configuración adicionales.
- **blacklist** *módulo* Ignora cualquier alias interno que un módulo interno puede definir para sí mismo. Esto le permite utilizar sólo alias definidos por modprobe. También evita conflictos de módulos donde dos módulos pueden tener el mismo alias definido internamente. Las entradas de listas negras predeterminadas se almacenan en uno o más archivos de listas negras en el directorio **/etc/modprobe.d**. Sus nombres comienzan con **blacklist**. Utilice el comando **modinfo** para presentar una lista de los alias internos de un módulo.

Entre las entradas más comunes se encuentran alias que se utilizan para tarjetas de red. Observe que no hay un nombre de dispositivo para dispositivos Ethernet en el directorio **/dev**. Esto es porque el nombre de dispositivo es realmente un alias para un módulo de red Ethernet que ha sido definido en un archivo de configuración modprobe (esto fue llamado **modules.conf** en publicaciones anteriores). Para agregar otra tarjeta Ethernet del mismo tipo, se coloca un alias para éste en el archivo de configuración modprobe. Para una segunda tarjeta Ethernet, se utiliza el nombre de dispositivo **eth1** como el alias. De esta forma, se hace referencia al segundo dispositivo Ethernet con el nombre **eth1**. Una entrada de configuración modprobe se muestra aquí:

```
alias eth1 ne2k-pci
```

SUGERENCIA Después de hacer cambios a los archivos **/etc/modprobe.conf** o **modprobe.d**, debe ejecutar **depmod** nuevamente para grabar cualquier cambio en dependencias de módulos.

Si agrega un modelo de tarjeta Ethernet diferente, tiene que especificar el módulo utilizado por ese tipo de tarjeta. En el siguiente ejemplo, la segunda tarjeta es una Realtek PCI estándar. Kmod ya ha detectado automáticamente la nueva tarjeta y cargado el módulo **ne2k-pci**. Sólo tiene que identificar esto como la tarjeta **eth1** en un archivo de configuración modprobe como **/etc/modprobe.conf**.

```
alias eth0 forcedeth
alias eth1 ne2k-pci
```

NOTA En lugar de un solo archivo **modprobe.conf**, la configuración modprobe se implementa al utilizar archivos separados en un directorio **/etc/modprobe.d**.

Aquí se muestra un ejemplo de archivo **modprobe.conf**. Observe el alias para el controlador Serial ATA Nvidia, **sata_nv** y el adaptador ATA paralelo AMD, **pata_amd**. Ambos tienen alias como adaptadores host SCSI. Se hace referencia a la tarjeta de sonido por su nombre de módulo, **snd-hda-intel**, en futuras operaciones de instalación y eliminación. Las últimas dos líneas son una sola, que comienza con **remove**.

```
alias eth0 forcedeth
alias eth1 ne2k-pci
alias scsi_hostadapter sata_nv
alias scsi_hostadapter1 pata_amd
alias snd-card-0 snd-hda-intel
options snd-card-0 index=0
options snd-hda-intel index=0
```

SUGERENCIA En algunos casos, tal vez Kmod no detecte un dispositivo en la forma que quiere y, por lo tanto, no cargue el módulo kernel que le gustaría. En este caso, los parámetros de kernel se especificaron en el cargador de arranque GRUB para cargar los módulos correctos.

Instalación de nuevos módulos de comercializadores: paquetes de controladores

A menudo, encontrará que los módulos de Linux no son compatibles con su dispositivo de hardware. En este caso, tal vez tenga que descargar controladores del comercializador de hardware o un grupo de desarrollo de fuente abierta para crear su propio controlador e instalarlo para que lo utilice su kernel.

Los controladores pueden ser archivos RPM o comprimidos. El proceso para instalar controladores difiere, dependiendo de la manera en que un comercializador dé soporte al controlador. Aquí se presentan diferentes tipos de paquetes:

- **Paquetes RPM o Deb** Algunos sitios de soporte técnico proporcionarán controladores ya empaquetados en archivos RPM o Deb para instalación directa.
- **Controladores compilados en archivos** Algunos proporcionarán controladores ya compilados para su distribución, pero empaquetados en archivos comprimidos. En este caso, una simple operación de instalación colocará el módulo de soporte en el directorio **modules** y se generará si está disponible para utilizarse con el kernel.
- **Código fuente** Otros proporcionan sólo el código fuente, que, cuando se compile, detectará la configuración de su sistema y compilará el módulo correspondiente.
- **Secuencias de comandos con código fuente** Algunos proporcionarán secuencias de comandos personalizadas, que pueden plantearle preguntas básicas acerca de su sistema y después instala y compila el módulo.

En el caso de controladores que vienen en forma de archivos comprimidos (**tar.gz** o **tar.bz2**), las operaciones de compilación e instalación normalmente utilizan una secuencia de comandos Makefile operada por el comando **make**. Una instalación simple suele requerir la ejecución del siguiente comando en el directorio de software del controlador:

```
make install
```

En el caso de sitios que sólo proporcionen el código fuente, tal vez tenga que realizar las operaciones de configuración y compilación como lo haría para cualquier software.

```
./configure
make
make install
```

En el caso de paquetes que no tienen una opción de instalación, compilada o de código fuente, tendrá que mover manualmente el módulo al directorio del módulo kernel, **/lib/modules/version**, y utilizar **depmod** y **modprobe** para cargarlo (consulte la sección anterior).

Si un sitio le da una secuencia de comandos personalizada, sólo ejecútela. Por ejemplo, las interfaces de red LAN Marvel gigabit que se encuentran en casi todas las tarjetas madre utilizan los controladores SysKonnect de Linux que se mantienen en el módulo **sk98lin.o**. La configuración de kernel estándar generará e instalará este módulo. Pero si está utilizando una nueva tarjeta madre, tal vez necesite descargar e instalar el controlador más reciente de Linux. Por ejemplo, algunos comercializadores proporcionan una secuencia de comandos, **install.sh**, que se ejecuta para configurar, compilar e instalar el módulo.

```
./install.sh
```

NOTA En Red Hat y Fedora, si sólo se le proporciona un archivo de código fuente para el módulo, como un archivo `.c`, se utilizan los archivos de encabezado de kernel en el directorio `/lib/modules/versión/build` para compilar el módulo. Consulte las notas de versión de Fedora o Red Hat para conocer más detalles de la manera de crear un Makefile personalizado para crear módulos. No tendrá que descargar e instalar el código fuente.

Instalación de nuevos módulos para el kernel

El código fuente para su kernel de Linux contiene un conjunto extenso de módulos; no todos se utilizan en su sistema. Los binarios de kernel proporcionados por casi todas las distribuciones vienen con un conjunto extenso de módulos ya instalados. Sin embargo, si instala un dispositivo que no tiene instalado soporte para el kernel, tendrá que configurar y compilar el módulo del kernel que proporciona los controladores. Esto requiere el uso del código fuente de kernel para seleccionar el módulo que necesita de una lista en una herramienta de configuración kernel y después regenerar sus módulos del kernel con el nuevo módulo incluido (consulte el capítulo 32). Entonces el nuevo módulo se copia en la biblioteca de módulos, instalándolo en su sistema. También se inserta en el archivo `/etc/modprobe.conf` con cualquier opción, o se utiliza `modprobe` para instalarlo de forma manual.

Descargue la versión de código fuente original del kernel en el archivo comprimido de **kernel.org**, después desempáquelo en cualquier directorio (pero no utilice `/usr/src/linux`). Como opción, se utiliza una versión empaquetada del código de kernel proporcionada por su distribución.

Ahora vaya al directorio del kernel y utilice el comando `make` con el argumento `gconfig` o `menuconfig` para desplegar los menús de configuración del kernel, que los invoca con los siguientes comandos. El comando `make gconfig` empieza con una interfaz X Window System que necesita ejecutarse en su escritorio desde una ventana de terminal.

```
make gconfig
```

Al utilizar los menús, seleccione los módulos que necesite. Asegúrese de que cada uno se marque como un módulo, al hacer clic en la casilla de verificación Módulo en `gconfig` o al escribir `m` para `menuconfig`. Una vez que el kernel se configure, guárdelo y salga de los menús de configuración. Después se compilan los módulos al crear los archivos binarios de módulo con el siguiente comando:

```
make modules
```

Esto coloca los módulos en el directorio de módulos fuente del kernel. Puede copiar el que quiera al directorio de módulos de kernel, `/lib/modules/versión/kernel`, donde versión es el número de versión de su kernel de Linux. Un método más simple es reinstalar todos sus módulos, al utilizar el siguiente comando. Esto copia todos los módulos compilados al directorio `/lib/modules/versión/kernel`:

```
make modules_install
```

NOTA Si está utilizando Red Hat o Fedora, primero asegúrese de que ha instalado el código fuente en el directorio `/usr/src/redhat/BUILD`.

Administración del kernel

El *kernel* es el sistema operativo que realiza tareas básicas como administración de memoria y acceso al disco, además de interactuar con el hardware que conforma su sistema. Por ejemplo, el *kernel* hace posibles características estándar de Linux como soporte a multitareas y multiusuario. También maneja comunicaciones con dispositivos como CD-ROM o discos duros. Los usuarios envían peticiones para acceder a estos dispositivos a través del *kernel*, que después maneja la tarea de nivel inferior de enviar las instrucciones apropiadas al dispositivo. Dada la gran variedad de dispositivos disponibles, habrá mucha variación entre los tipos de dispositivos conectados al sistema Linux. Cuando éste se instala, el *kernel* se configura apropiadamente para sus dispositivos conectados. Sin embargo, si agrega un nuevo dispositivo, tal vez tenga que habilitar el soporte a éste en el *kernel*. Para esto es necesario reconfigurar el *kernel* para que dé soporte al nuevo dispositivo a través de un procedimiento que suele conocerse como *generar o compilar el kernel*. Además, de manera continua se lanzan nuevas versiones del *kernel* para proporcionar soporte mejorado a sus dispositivos, además de permitir nuevas características y confiabilidad mejorada para un sistema que se ejecuta sin problemas. Usted puede bajar, compilar e instalar todas estas nuevas versiones en su sistema.

Versiones de kernel

El número de versión de un *kernel* de Linux consta de cuatro segmentos: los números mayor, menor, de revisión y seguridad/depuración. El *número mayor* se incrementa con cambios mayores en el *kernel* y se carga en muy pocas ocasiones. El *número menor* indica una revisión mayor del *kernel*. El número de revisión se utiliza para dar soporte a nuevas características. El número de seguridad/depuración se utiliza para seguridad y depuración. Las versiones de desarrollo aparecerán primero como candidatos de versión, que tendrán un *rc* en el nombre. A medida que se descubren y corrigen errores, y que se introducen nuevas características, se lanzan nuevas revisiones del *kernel*. Por ejemplo, 2.6.21.1 tiene el número mayor 2 y el número menor 6, con el número de revisión 21 y el número de seguridad/depuración 1. Una versión candidata de un nuevo *kernel* tendrá un nombre como **2.6.22-rc3**.

Las distribuciones a menudo agregan otro número que alude a un conjunto específico de parches aplicados al *kernel*, además de una distribución inicial. Por ejemplo, el *kernel* es 2.6.21-1.3116, donde **3116** es el número de parche. En distribuciones que soportan paquetes RPM, puede utilizar una consulta RPM para ver qué versión está instalada, como se muestra aquí:

```
rpm -q kernel
```

Puede tener más de una versión del kernel instalada en su sistema. Para ver cuál se está ejecutando actualmente, se utiliza el comando **uname** con la opción **-r** (la opción **-a** proporciona información más detallada).

```
uname -r
```

Se trabaja constantemente en el kernel de Linux, y nuevas versiones se lanzan cuando están listas. Las distribuciones pueden incluir diferentes versiones de Linux. Los kernel de Linux están disponibles en **kernel.org**. Además, a menudo están disponibles paquetes RPM para un nuevo kernel en sitios de actualización de distribución. Una de las posibles razones para actualizar su kernel es proporcionar soporte a nuevo hardware o a características que son compatibles con su versión de la distribución. Por ejemplo, tal vez necesite soporte a un nuevo dispositivo que no se proporciona en su versión de la distribución del kernel. Tal vez no se incluyan ciertas características en una versión de la distribución, porque se consideran experimentales o un riesgo de seguridad.

NOTA *En muchos casos, no necesita compilar e instalar un nuevo kernel sólo para agregar soporte a un nuevo dispositivo. Los kernels proporcionan casi todo el soporte a dispositivos en forma de módulos que se cargan; de ellos, sólo se instalan con el kernel los que son necesarios. Lo más probable es que su kernel actual tenga el módulo que necesita; sólo requiere compilarlo e instalarlo.*

SUGERENCIA *Muchos módulos se compilan de forma separada al utilizar fuentes proporcionadas por comercializadores, como controladores de dispositivo de red actualizados. Para éstos, sólo necesita los encabezados de kernel, que ya están instalados en el directorio `/usr/lib/modules/versión/build`, donde la versión es la instalada en el kernel. En estos casos, no tiene que instalar todo kernel completo para agregar y modificar módulos.*

Referencias

Para conocer más acerca del kernel de Linux, visite **kernel.org**, el depósito oficial para los kernels de Linux. Ahí encontrará el código fuente más actual, además de la documentación. El sitio Web de su distribución también proporcionará documentación en línea para instalación y compilación del kernel en sus sistemas. También existen varios HOWTO de Linux relacionados con el tema. Los paquetes de software de código fuente del kernel también incluyen documentación extensa. Los archivos del código fuente de kernel también se instalan siempre, ya sea directamente en un directorio local o en el directorio que utilizan los paquetes del kernel de la distribución. La fuente por sí sola será un directorio etiquetado **linux-versión**, donde la *versión* es la del kernel, como en **linux-2.6.21**. En este directorio, se encuentra un subdirectorio llamado **/Documentation**, que contiene un conjunto extenso de archivos y directorios, que documentan características, módulos y comandos del kernel. La siguiente lista de recursos de kernel también contiene más información:

- **kernel.org** El sitio Web oficial del kernel de Linux; todos los nuevos kernels se originan de aquí.
- **linuxhq.com** El sitio central de Linux, código fuente de kernel y parches.
- **kernelnewbies.org** Fuentes e información del kernel de Linux.
- **en.tldp.org** Proyecto de documentación de Linux.

Afinación del kernel: parámetros de runtime del kernel

Varias características del kernel, como el reenvío IP o el número máximo de archivos, se activan o desactivan sin tener que compilar e instalar un nuevo kernel o módulo. Estos parámetros se afinan mediante los archivos del directorio `/proc/sys`. Los parámetros se establecen en el archivo `/etc/sysctl.conf`. Se utiliza el comando `sysctl` directamente. La opción `-p` causa que `sysctl` lea parámetros del archivo `/etc/sysctl.conf` (también se puede especificar un archivo diferente). Se utiliza la opción `-w` para cambiar parámetros específicos. Se hace referencia a este parámetro con su clave. Una *clave* es el nombre de parámetro que ya está fijo para sus categorías de sistema `proc` (directorios), como `net.ipv4.ip_forward` para el parámetro `ip_forward` ubicado en `/proc/sys/net/ipv4/`. Para desplegar el valor de un parámetro particular, sólo utilice su clave. La opción `-a` presenta una lista de todos los parámetros que se cambian. En el siguiente ejemplo, el usuario cambia el parámetro de nombre de dominio, al hacer referencia a éste con la clave `kernel.domainname` (el comando `domainname` también le permite establecer el parámetro `kernel.domainname`):

```
# sysctl -w kernel.domainname="mipista.com"
```

El siguiente ejemplo activa el reenvío de IP:

```
# sysctl -w net.ipv4.ip_forward=1
```

Si sólo utiliza la clave, se despliega el valor actual del parámetro:

```
# sysctl net.ipv4.ip_forward
net.ipv4.ip_forward = 1
```

Instalación de una nueva versión del kernel

Para instalar un nuevo kernel, debe descargar los paquetes de software del kernel que corresponde a su sistema. Se instala un nuevo kernel al descargar una versión binaria de su sitio Web de la distribución e instalarlo, o al descargar el código fuente, compilar el kernel y después instalar el archivo binario resultante junto con los módulos. La versión binaria del kernel se proporciona en un paquete RPM, y puede instalar un nuevo kernel como lo haría con cualquier otro paquete de software RPM.

La forma más sencilla de instalar un kernel nuevo es mediante la utilidad de instalación del software de su distribución para descargar e instalar automáticamente un paquete de kernel preparado para la distribución. La instalación creará una entrada GRUB para que, cuando inicie, el nuevo kernel se muestre como una de las opciones, por lo general la predeterminada.

Si quiere descargar manualmente los paquetes RPM del kernel, tenga en cuenta que la instalación completa del kernel suele incluir una serie de paquetes RPM que comienzan con la palabra *kernel*. También hay otros paquetes que tal vez necesite y que contienen archivos de configuración de sistema actualizados para el nuevo kernel. También se utilizan como guía los paquetes ya instalados en su sistema. Utilice el comando `rpm` con la opción `-qa` para desplegar una lista de todos los paquetes y después canalice esa lista a través del comando `grep` con el patrón `kernel` para desplegar sólo los paquetes kernel:

```
rpm -qa | grep kernel
```


La versión en código fuente del kernel está disponible para su descarga en sitios FTP de la distribución, en el directorio fuente, y se incluye en el CD-ROM de código fuente de la distribución. También se descargan las fuentes más recientes directamente de **kernel.org**. No importa de dónde descargue una versión del kernel, siempre será la misma. El código fuente de una versión de kernel descargado de un sitio de distribución es el mismo que el de **kernel.org**. Los parches para esa versión se aplican a cualquier distribución.

Paquetes del kernel de CPU

Las distribuciones proporcionarán diferentes paquetes del kernel optimizados para varias CPU populares. Seleccione el apropiado para su equipo. Todos los kernels incluyen soporte a multiprocesadores. La distribución x86 incluirá la versión x86, y la distribución de 64 bits almacenará las versiones x86_64. Todos los paquetes tienen el nombre de kernel, pero cada uno cuenta con un calificador diferente. Para el x86, hay dos paquetes de kernel diferentes, uno para las nuevas CPU Pentium 2, 3 y 4, y otro para las Pentium antiguas, las CPU AMD K6 y otros sistemas más viejos. Cada paquete tendrá una CPU de referencia en su nombre de archivo: 686 para Pentium 2, 3 y 4; 586 para Pentium, K6 y otros sistemas. En el caso de sistemas de 64 bits, como la serie Athlon 64, la distribución de 64 bits sólo incluirá un paquete x86_64. También hay paquetes para el kernel de virtualización Xen y existe el kernel PAE para soporte de memoria de sistema extendida. kernel-kdump es una nueva versión mínima para volcados de sistema, útiles para depuración (el soporte a kdump se incluye en los kernel de 32 bits).

Por lo general, para cada kernel existen paquetes correspondientes de encabezado de kernel (también conocidos como construcciones), denominados con el término **devel**, que sólo contiene los encabezados de kernel. Éstos se utilizan para compilar módulos de kernel o aplicaciones de software que no necesitan el código fuente completo del kernel, sólo los encabezados. Los encabezados de su kernel actual ya están instalados. Los encabezados de kernel estarán instalados en el directorio **/etc/src/kernels**, con un vínculo **build** en el directorio **/lib/modules** del kernel.

Instalación de paquetes de kernel: /boot

En casi todas las distribuciones, los kernels se instalan en el directorio **/boot**. Al realizar una operación **ls -l** en este directorio se presenta una lista de los kernels instalados. Existe un archivo para su kernel antiguo y uno para el nuevo. Si tomó las precauciones descritas en la sección anterior, tal vez ya haya cambiado el nombre del kernel viejo. Si está utilizando un cargador de arranque como GRUB, necesita cambiar su archivo de configuración (**/boot/grub/menu.lst**) para agregar la entrada que invoque el nuevo kernel. El kernel arranca usando el archivo de kernel **boot/vmlinuz-version**. En su archivo **/boot/grub/menu.lst**, necesita una línea de kernel para hacer referencia a este archivo kernel. También necesita incluir una línea para el disco RAM, **initrd**.

```
kernel /boot/vmlinuz-version ro root=/dev/hda3
initrd /boot/initrd-version.img
```

Si su sistema tiene un controlador SCSI u otro hardware especializado, RPM también creará un disco RAM para almacenar módulos de soporte apropiados (se crea un disco RAM manualmente con el comando **mkinitrd**). El disco RAM recibirá el nombre **initrd-version-kernel.img** y se ubica en el directorio **/boot**, como en **/boot/initrd-2.6.version.img**.

PRECAUCIÓN *User-mode Linux (UML) es una versión opcional del kernel diseñada para ejecutarse como un programa independiente separado del kernel. En efecto, crea una máquina virtual con almacenamiento de disco implementado en un archivo de usuario. UML suele utilizarse para probar software o experimentar con configuraciones del kernel sin tener que dañar el sistema real. También se utiliza UML para implementar hospedaje virtual, al ejecutar varias máquinas virtuales en un host físico. Con una máquina virtual, se controla el acceso al sistema host, que proporciona mayor seguridad. Encuentre más acerca del modo de usuario de Linux en user-mode-linux.sourceforge.net.*

Pasos preventivos para modificar un kernel de la misma versión

Si quiere modificar su configuración del kernel y generar una nueva, debe retener una copia de su kernel actual. En caso de que algo salga mal con su versión modificada, siempre se puede arrancar desde la copia que almacenó. No necesita preocuparse de que esto pase si está instalando una nueva versión del kernel. Se les da un nombre diferente a los nuevos kernel, para que los anteriores no se sobrescriban.

Para retener una copia de su kernel actual, se hace una copia de respaldo, permitiendo que el original se sobrescriba. Una versión instalada de un kernel usa varios archivos en el directorio **/boot**. Cada archivo termina con el número de la versión del kernel. Éstos incluyen el archivo **vmlinuz**, que es la imagen de archivo real del kernel, junto con varios archivos de soporte, **System.map**, **config** e **initrd**. El archivo **System.map** contiene símbolos de kernel que necesitan los módulos para iniciar funciones del kernel. Por ejemplo, el archivo de imagen del kernel se llama **vmlinuz-versión**, donde *versión* es el número de versión adjunto, como en **vmlinuz-2.6.versión**. El archivo **System.map** para este kernel tiene el nombre **System.map-2.6.versión**. Aquí están los archivos de kernel para la versión **2.6.v**:

```
/boot/vmlinuz-2.6.versión
/boot/System.map-2.6.versión
/boot/initrd-2.6.versión.img
/boot/config-2.6.versión
```

Si, por otra parte, está creando una versión modificada del mismo kernel, el archivo del kernel, aquí llamado **vmlinuz-2.6.versión**, se sobrescribirá con el nuevo archivo de imagen del kernel, junto con los archivos **System.map** y **config**. Para mantener su versión de trabajo actual, primero tiene que hacer una copia de estos archivos. Se hace una copia del archivo **/boot/vmlinuz-2.6.versión**, al darle otro nombre, como se muestra aquí:

```
cp /boot/vmlinuz-2.versión/boot/vmlinuz-2.6.versión.orig
```

Además, se hacen copias de seguridad de los archivos **System.map** y **config**. También debe respaldar sus módulos ubicados en el directorio **/lib/modules/versión**; aquí *versión* es el número de versión del kernel. De otra forma, perderá los módulos que ya se configuraron para funcionar con el kernel original. Para la versión **2.6.versión**, las bibliotecas se cargan en **/lib/modules/2.6.versión**. Si está compilando una versión diferente, esas bibliotecas se colocan en un nuevo directorio, que incluye el nuevo número de versión en el nombre.

Cargador de arranque

La instalación de un paquete de kernel creará automáticamente una entrada de cargador de arranque GRUB para el nuevo kernel. Será capaz de seleccionarlo en el arranque. Las entradas para su kernel más antiguo permanecerán.

Si crea una versión personalizada de su kernel actual mientras mantiene la versión original como respaldo, entonces necesitaría crear una nueva entrada para el kernel original en el archivo de configuración del cargador de arranque. Es aconsejable dejar la entrada para el kernel original, en caso de que algo salga mal con el nuevo kernel. De esta forma, siempre se reinicia y se selecciona el kernel original. Por ejemplo, en `/boot/grub/menu.lst`, agregue una nueva entrada, similar a la del kernel anterior, que hace referencia al nuevo kernel en su instrucción `kernel`. La entrada `menu.lst` buscará algo como el siguiente código. Entonces puede seleccionar la entrada con el título “Old Linux (2.6.versión.orig)” en el menú GRUB para lanzar el kernel original.

```
title Linux original (2.6.versión.orig)
root (hd0,2)
kernel /boot/vmlinuz-2.6.versión.orig root=/dev/hda3
initrd /boot/initrd-2.6.versión.orig.img
```

Si utiliza una etiqueta para la partición de arranque, la opción `root` para la instrucción `kernel` se verá así para una partición de arranque etiquetada `/`.

```
kernel /boot/vmlinuz-2.6.versión.orig ro root=LABEL=
```

Disco de arranque

También debe tener un CD-ROM de arranque listo, sólo en caso de que algo salga mal con la instalación (por lo general, habrá creado uno durante la instalación). Con un CD-ROM de arranque, se inicia su sistema sin tener que utilizar el cargador de arranque. En Red Hat, Fedora y distribuciones similares, se crea un CD-ROM de arranque al utilizar la utilidad `mkbootdisk`, pero necesita conocer el número de versión completo para su kernel. En realidad, se tienen varios kernels instalados y se crea un CD-ROM para cada uno (su archivo `/boot/grub/menu.lst` incluye el número de versión de su kernel). Si la versión del kernel es 2.6.versión, se utiliza como argumento en el comando `mkbootdisk` para crear un CD-ROM de arranque de su sistema.

Para crear un CD-ROM de arranque, se utiliza la opción `--iso` con la opción `--device` para especificar el archivo de imagen de CD. Luego se graba el archivo de imagen en un CD-ROM con una aplicación como K3b. En el siguiente ejemplo, el usuario crea un archivo de imagen de CD-ROM, llamado `miimagen.iso`, para un CD-ROM de arranque del kernel 2.6.versión:

```
mkbootdisk --iso --device miimagen.iso 2.6.versión
```

Compilación del kernel desde el código fuente

En vez de instalar versiones binarias ya compiladas del kernel, se instala el código fuente en su sistema y se utiliza para que usted mismo cree los archivos binarios. Los archivos de código fuente del kernel se compilan con el compilador `gcc`, igual que cualquier otro archivo de código fuente. Una ventaja de compilar el kernel es que se mejora su configuración, al agregar soporte para ciertos tipos de dispositivos como Bluetooth.

NOTA Muchas distribuciones instalan recursos del kernel al utilizar sus propios métodos de paquete de software. Esto es verdad sobre todo para Fedora y Red Hat, cuyos paquetes del kernel SRPMS se instalan en el directorio `/usr/src/redhat` y utilizan archivos SPEC para extraer la versión de kernel que quiere. Revise su documentación de la distribución en paquetes de código fuente de kernel para conocer más detalles.

Instalación de fuentes del kernel: archivos y parches del kernel

También se descargan las fuentes del kernel originales de **kernel.org**. Esta versión no estará optimizada para su distribución (los paquetes de kernel de la distribución estarán optimizados). Se debe colocar en el directorio de su elección, pero no en el directorio `/usr/src/linux`.

Estas versiones suelen ser mucho más recientes que las disponibles en el sitio de su distribución, pero tal vez no se hayan probado a fondo en su plataforma de distribución. Las fuentes del kernel están en forma de archivos comprimidos (**.tar.gz**). Tienen el prefijo **linux** con el nombre de versión como sufijo. Se descomprime y extrae el archivo con los siguientes comandos. Primero vaya al directorio local que elija y después desempaquete el archivo con File Roller o con el comando **tar**. Crea un directorio con el prefijo **linux** donde se colocan los archivos fuente. En el siguiente ejemplo se extrae el kernel 2.6.21.1:

```
cd mykernel
tar -xzf linux-2.6.21.1.tar.gz
```

Asegúrese de desempaquetar el archivo para la versión **kernel.org** en un directorio que seleccione, como **mikernel**, en el directorio home. La fuente residirá dentro de un subdirectorio que tiene el prefijo **linux** y el sufijo que contiene la versión del kernel, como en **linux-2.6.21** para el kernel 2.6, revisión 21. El directorio local en este ejemplo sería **mikernel/linux-2.6.21**.

SUGERENCIA Si está utilizando la fuente kernel original, debe también revisar cualquier parche.

Configuración del kernel

Una vez que la fuente se instala, debe configurar el kernel. La configuración consiste en determinar las características para las que quiere proporcionar soporte en el nivel del kernel. Entre éstas se incluyen controladores para diferentes dispositivos, como tarjetas de sonido y dispositivos SCSI. Puede configurar las características como se incluyen directamente en el kernel o como módulos que el kernel carga de acuerdo con sus necesidades. También se excluyen características específicas. Las características que se incorporan directamente en el kernel forman un programa de kernel mayor. Las características que se configuran como módulos separados también se actualizan fácilmente. La documentación de muchos dispositivos que proporcionan soporte a sonido, video y red se encuentra en el directorio `/usr/share/doc`. Revise el paquete **kernel-doc** para encontrar una lista de documentación proporcionada.

NOTA Si configuró su kernel antes y ahora quiere iniciar desde las configuraciones predeterminadas, se utiliza el comando **make mrproper** para restaurar las configuraciones predeterminadas del kernel.

Herramientas de configuración del kernel

Para configurar el kernel se utiliza una de varias herramientas de configuración disponibles: **config**, **menuconfig**, **xconfig** (qconf) y **gconfig** (gkc). También se puede editar directamente el archivo de configuración. Estas herramientas realizan las mismas tareas de configuración pero utilizan diferentes interfaces. La herramienta **config** es una simple secuencia de comandos de configuración que proporciona indicadores de línea de comandos para diferentes opciones de configuración. La herramienta **menuconfig** proporciona un menú basado en cursor, que aún puede ejecutar desde la línea de comandos. Existen entradas de menú para diferentes categorías de configuración y puede seleccionar las que quiera. Para marcar una característica que habrá de incluirse en el kernel, muévela y oprima BARRA ESPACIADORA. Un asterisco aparece entre los

paréntesis vacíos, a la izquierda de la entrada. Si quiere que sea un módulo, oprima M, y una M aparecerá en los paréntesis. La opción **xconfig** ejecuta qconf, la herramienta de configuración del kernel GUI basada en QT (KDE) y requiere que las bibliotecas QT (KDE) se instalen primero. La opción **gconfig** ejecuta la herramienta gkc, que utiliza una interfaz GTK, que requiere que GNOME se instale primero. qconf y gkc proporcionan árboles de menús expansibles, paneles que se seleccionan y ventanas de ayuda. Entre las características que es posible seleccionar se incluyen casillas de verificación en que puede hacer clic. Todas estas herramientas guardan sus configuraciones en el archivo **.config** del directorio de la fuente del kernel. Si quiere eliminar por completo una configuración, utilice la opción **mrproper** para eliminar el archivo **.config** y cualquier archivo binario, para empezar desde cero.

```
make mrproper
```

Se inicia una herramienta de configuración al antecederla con un comando **make**. Asegúrese de que se encuentra en el directorio del kernel (ya sea la ubicación de la distribución de sus paquetes de kernel o en el directorio local que utilizó para el archivo comprimido, como **tar.gz**). El proceso de iniciar una herramienta de configuración es una operación **make** que utiliza el Makefile del kernel de Linux. La herramienta **xconfig** debe iniciarse en una ventana de terminal de su administrador de ventanas. Las herramientas **menuconfig** y **config** se inician en una línea de comandos de shell. En el siguiente ejemplo se presenta una lista de los comandos para iniciar **xconfig**, **gconfig**, **menuconfig** y **config**:

```
make gconfig
make xconfig
make menuconfig
make config
```

gconfig (gkc)

La herramienta de configuración de kernel de GTK (gkc) se invoca con la opción **gconfig**. Esto utiliza una interfaz basada en GNOME que es similar a qconf (**xconfig**). La herramienta gkc abre una ventana Linux Kernel Configuration con submenús expansibles como los de qconf. Muchas categorías se organizan en algunos encabezados principales, y ahora se incluyen varios bajo el menú Device Drivers. Las entradas de los botones Load y Save y el menú File se utilizan para guardar la configuración o copiarla en un archivo. Los botones de vista Single, Split y Full le permiten desplegar menús en una ventana, en un panel de despliegue con otro panel que contenga un árbol expansible para seleccionar entradas, o como un sólo árbol expansible de entradas. El botón Expand expandirá todos los encabezados y subencabezados, mientras que Collapse le permitirá expandir sólo los que quiera desplegar. Utilice los triángulos hacia abajo y hacia los lados para expandir o colapsar las subentradas.

Al hacer clic en una entrada, se abre una ventana con una lista de diferentes características incluidas. Las entradas se ordenan en columnas que presentan la opción, el nombre actual, su rango (yes, module o no), y sus datos (yes, no o module status). Las entradas del menú Options le permiten determinar cuáles columnas se desplegarán: Name para el nombre de módulo actual; Range para las entradas que se seleccionan entre yes, no y module; y Data para el estado de las opciones, denominadas Value.

Las entradas Range se denominan N, M y Y y se utilizan para seleccionar si no se incluye una opción (N), si se agrega como un módulo (M) o se compila directamente en el kernel (Y). Las entradas que se seleccionan desplegarán un subrayado. Al hacer clic en el subrayado cambiará su entrada a Y para módulo o para inclusión directa del kernel, y N para no incluirlo. La columna Value mostrará cuál está seleccionado.

La columna Option incluirá un estado que muestra si la opción se incluye directamente (una marca de verificación), se incluye como un módulo (marca de línea) o no se incluye (vacío)

Para seleccionar o dejar de seleccionar una entrada, haga doble clic en el nombre de la opción en el campo Options. Verá sus casillas de verificación con una marca de verificación, una línea (módulo) o vacías. Las entrada N, M y Y corresponden a que no se incluye, módulo o se incluye en el kernel. La preferencia predeterminada para el módulo o la inclusión directa en el kernel para esa opción está seleccionada automáticamente. El cambio puede ser manual, si así lo desea.

xconfig (qconf)

La opción **xconfig** involucra la herramienta **qconf**, que está basada en las bibliotecas QT de KDE. Es necesario instalar primero KDE. La herramienta **qconf** abre una ventana Linux Kernel Configuration que presenta una lista de diferentes categorías de configuración. Tiene una interfaz un poco más simple, sin los botones de expandir o contraer o las columnas para estado de módulo o fuente.

Características importantes de la configuración del kernel

Las herramientas **xconfig**, **menuconfig** y **gconfig** proporcionan una ayuda excelente sensible al contexto para cada entrada. A la derecha de cada entrada está un botón Help. Haga clic en éste para desplegar una explicación detallada sobre qué hace la característica y por qué la incluiría directamente o como un módulo, o incluso por qué la excluiría. Cuando tenga duda acerca de una característica, siempre utilice el botón Help para saber exactamente qué hace y por qué querría utilizarla. Muchas de estas características se describen aquí. La categoría principal de una característica se incluye entre paréntesis.

***SUGERENCIA** Como regla, las características en uso continuo, como el soporte de red y sistema de archivos, deben compilarse directamente en el kernel. Las características que cambian fácilmente, como las tarjetas de sonido, o las que se usan con menos frecuencia deben compilarse como módulos. De otra forma, su archivo de imagen de kernel puede volverse muy grande y hará más lenta la ejecución.*

- **Loadable Module Support** En casi todos los casos, debe asegurarse de que en su kernel es posible cargar módulos. Haga clic en el botón Loadable Module Support para desplegar una lista de varias opciones de administración de módulos. Asegúrese de que Enable Loadable Module Support esté marcado en Yes. Esta característica le permite a su kernel agregar módulos según sea necesario. Kernel Module Loader también debe estar en Yes, porque esto permite a sus daemon, como su servidor Web, cargar cualquier módulo que necesite.
- **Processor Type And Features** La ventana Processor Type And Features le permite configurar soporte a su sistema particular. Aquí, se selecciona el tipo de procesador que tiene (486, 586, 686, Pentium III, Pentium IV, etc.), además de la cantidad máxima de memoria en su soporte de sistema (hasta 64 GB con el kernel 2.4).
- **General Setup** La ventana General Setup le permite seleccionar características generales, como red, soporte a PCI en BIOS y administración de energía, además del soporte a ELF y binario **a.out**. También se da soporte a **sysctl** para cambiar de forma dinámica parámetros de kernel especificados en los archivos **/proc**. Se utiliza **redhat-config-proc** (la herramienta Kernel Tunning en el menú Herramientas del sistema) para hacer estos cambios dinámicos al kernel. En el menú de soporte de controlador de dispositivo adicional, se habilitan características especializadas como Crypto IP Encapsulation (CIPE) y SSL acelerado.

- **Block Devices (Device Drivers)** La ventana Block Devices incluye entradas que habilitan el soporte a sus dispositivos IDE, unidades de disco flexible y dispositivos de puerto paralelos. Las características especiales, como soporte a discos RAM y el dispositivo para montar archivos de imagen de CD-ROM, también están ahí.
- **Multi-Device Support (RAID y LVM) (Device Drivers)** La ventana Multi-Device Support presenta entradas que habilitan el uso de dispositivos RAID. Puede seleccionar el nivel de soporte RAID que quiera. Aquí también puede habilitar el soporte a Logical Volumen Management (LVM), que le permite combinar particiones en volúmenes lógicos que se administran de forma dinámica.
- **Networking Options (Device Drivers/Networking Support)** La ventana Networking Options muestra un conjunto extenso de capacidades de red. La entrada TCP/IP Networking debe estar habilitada para permitir cualquier tipo de red de Internet. Aquí, se especifican características que permiten a su sistema operar como una puerta de enlace, una firewall o un enrutador. Network Packet Filtering habilita el soporte a firewall IPtables. También existe el soporte a otros tipos de redes, incluidas AppleTalk e IPX. Apple Talk debe estar habilitado si quiere utilizar NetTalk para conectarse a un sistema Macintosh en su red (Filesystems).
- **ATA/IDE/MFM/RLL Support (Device Drivers)** En la ventana ATA/IDE/MFM/RLL Support, puede hacer clic en el botón IDE, ATA and ATAPI Block Device para abrir una ventana donde se selecciona el soporte a discos duros ATA IDE y CD-ROM ATAPI.
- **SCSI Support (Device Drivers)** Si tiene cualquier dispositivo SCSI en su sistema, asegúrese de que las entradas en la ventana SCSI Support estén establecidas en Yes. Aquí se habilita el soporte a discos SCSI, unidades de cinta y CD-ROM. La ventana SCSI Low-Level Drivers despliega una lista extensa de dispositivos SCSI compatibles con Linux. Asegúrese de que estén seleccionados los que tiene.
- **Network Device Support (Device Drivers/Networking Support)** La ventana Network Device Support presenta una lista de varias características generales para soporte a dispositivo de red. Aquí existen entradas para ventanas que incluyen soporte a tipos particulares de dispositivos de red, incluidos Ethernet y token ring, además de interfaces WAN y dispositivos AppleTalk. Muchos de estos dispositivos se crean como módulos que se cargan cuando se necesitan. Seleccione que se reconstruya su kernel con soporte a cualquiera de estos dispositivos generados directamente en el kernel.
- **Multimedia Devices (Device Drivers)** Los dispositivos multimedia proporcionan soporte a varias tarjetas multimedia, al igual que Video4Linux.
- **File Systems** En esta ventana se presentan los diferentes tipos de sistemas de archivos que Linux soporta. Éstos incluyen sistemas de archivos de Windows como DOS, VFAT y NTFS, además de archivos de CD-ROM como ISO y UDF. Se incluyen sistemas de archivos de red como NFS, SMB (Samba) y NCP (NetWare), además de varios sistemas de archivos como HFS (Macintosh).
- **Character Devices (Device Drivers)** La ventana Character Devices presenta características de dispositivos como su teclado, ratón y puertos seriales. Existe soporte a ratón serial y de bus.

- **Sound (Device Drivers)** Para el kernel 2.4, la ventana Sound presenta una lista de diferentes tarjetas de sonido compatibles con el kernel. Seleccione la de su sistema. En el caso de sistemas más viejos, tal vez tenga que proporcionar IRQ, DMA y E/S base que usa su tarjeta de sonido. Éstas se compilan como módulos separados, entre los que se selecciona si quiere que se incluyan directamente en el kernel. Para el kernel 2.6, se selecciona el soporte de sonido Advanced Linux Sound Architecture, que lo expande a controladores para dispositivos de sonido particulares (también se incluye Open Sound System, aunque esté descontinuado).
- **Bluetooth Devices (Device Drivers/Networking Support)** Aquí hay soporte a periféricos compatibles con Bluetooth, al presentar una lista de controladores para interfaces USB, serial y tarjetas PC.
- **Kernel Hacking** La ventana Kernel Hacking presenta una lista de características de interés para desarrolladores que trabajan en el nivel de kernel y necesitan modificar el código de éste. Puede hacer que el kernel incluya información de depuración y también proporcione algunas medidas de control durante fallas.

Una vez que configure sus opciones, guarde su configuración. Al seleccionar la entrada Save en el menú File se sobrescribe su archivo de configuración **.config**. La opción Guardar como le permite guardar su configuración en un archivo particular.

Compilación e instalación del kernel

Ahora que la configuración está lista, se compila su kernel. Tendrá que limpiar cualquier archivo de objeto y dependencia que pueda permanecer de una compilación previa. Utilice el siguiente comando para eliminar estos archivos:

```
make clean
```

Se utilizan varias opciones para compilar el kernel (consulte la tabla 32-1). La opción **bzImage** sólo genera un archivo de kernel llamado **bzImage** y lo coloca en el directorio **arch**. En el caso de

Opción	Descripción
zImage	Crea el archivo de kernel llamado zImage ubicado en el directorio arch or arch/i386/boot .
install	Crea el kernel y lo instala en su sistema.
zdisk	Crea el archivo de kernel y lo instala en un disco flexible (crea un disco de arranque, 1.44 MB).
bzImage	Crea el archivo de kernel comprimido y lo llama bzImage .
bzdisk	Crea el kernel y lo instala en un disco flexible (crea un disco de arranque). Es útil sólo para construcciones pequeñas de kernel, 1.44 MB.
fdimage	Crea una imagen de disco flexible con el kernel, 1.44 MB (de arranque).
fdimage288	Crea una imagen de disco flexible 2.88 con el kernel (de arranque).

TABLA 32-1 Compilando opciones para el comando de kernel **make**

sistemas Intel y AMD, **bzImage** se encuentra en el subdirectorio **i385/boot**, **arch/i386/boot**. Para una fuente de kernel, esto es un **arch/i386/boot**.

```
make bzImage
```

Las opciones que se muestran en la Tabla 32-1 crean el kernel, pero no los módulos (tales características del kernel se tienen que compilar en módulos separados). Para compilar sus módulos, se utiliza el comando **make** con el argumento **modules**.

```
make modules
```

El comando **make** sin argumentos creará **bzImage** y los módulos.

```
make
```

Para instalar sus módulos, se utiliza el comando **make** con la opción **modules_install**. Esto instala los módulos en el directorio **/lib/modules/num-versión**, donde *num-versión* es el número de la versión del kernel. Debe hacer una copia de respaldo de los módulos viejos antes de instalar los nuevos.

```
make modules_install
```

La opción **install** genera los archivos del kernel y los instala en su sistema como **vmlinuz**, al incorporar el paso **make bzImage**. Esta operación reemplazará los archivos del kernel como **bzImage** en el directorio **/boot**, al darles los nombres y los números de versión apropiados de kernel.

```
make install
```

Aquí se muestran los comandos para una compilación e instalación simple:

```
make clean
make
make modules_install
make install
```

Si quiere, se insertan éstos en menos líneas, separando los comandos con punto y coma, como se muestra aquí:

```
make clean; make; make modules_install; make install
```

Una forma más segura de realizar estas operaciones en líneas simples es hacerlas condicionalmente dependientes una de la otra, al utilizar el comando **&&**. En el método anterior, si una operación tiene un error, la siguiente todavía se ejecutará. Al hacer las operaciones condicionales, cada operación sólo se ejecuta si la anterior tiene éxito.

```
make clean && make && make modules_install && make install
```

Instalación manual de la imagen del kernel

Para instalar un archivo **bzImage** de kernel, copie el archivo **bzImage** al directorio donde reside el kernel y asígnele el nombre utilizado en su distribución, como **vmlinuz-2.6.versión**. Recuerde respaldar primero el archivo de kernel antiguo, como se observó en los pasos preventivos. **vmlinuz** es un vínculo simbólico con un archivo de kernel real que tiene el término **vmlinuz** en el nombre de

versión. Así, para instalar de forma manual un archivo **bzImage**, se copia al directorio **/boot** con el número de versión adjunto como **vmlinuz-2.6.versión**.

```
make bzImage
cp arch/i386/boot/bzImage /boot/vmlinuz-2.6.versión
```

SUGERENCIA La opción **bzImage**, y las que comienzan con la letra **b**, crean una imagen de kernel comprimida. Tal vez esta imagen no funcione en sistemas más antiguos. Si no, intente utilizar la opción **zImage** para crear un archivo de kernel llamado **zImage**. Después instale el archivo **zImage** de forma manual de la misma forma que lo haría con **bzImage**. Tenga en cuenta que el soporte para **zImage** se eliminará progresivamente.

También tendrá que hacer una copia del archivo **System.map**, relacionándolo con el vínculo simbólico **System.map**.

```
cp arch/i386/boot/System.map /boot/System.map-2.6.versión
```

Los siguientes comandos muestran una compilación básica de una instalación manual. En primer lugar, todos los archivos binarios anteriores se eliminan con la opción **clean**. Después el kernel se crea al utilizar la opción **bzImage**. Esto crea un programa de kernel llamado **bzImage**, ubicado en el directorio **arch/i386/boot**. Este archivo de kernel se copia en el directorio **/boot** y recibe el nombre **vmlinuz-2.6.versión**. Después se crea un vínculo simbólico llamado **/boot/vmlinuz** para el archivo kernel **vmlinuz-2.6.versión**. Por último, se crean e instalan los módulos:

```
make clean
make
make modules_install
cp arch/i386/boot/bzImage /boot/vmlinuz-2.6.versión
cp System.map /boot/System.map-2.6.versión
```

Discos de arranque del kernel

En lugar de instalar el kernel en su sistema, sólo se coloca en un disco de arranque o CD-ROM y se arranca su sistema desde ese disco. En el caso de un CD-ROM, primero se crea el kernel como **vzImage**, se instala el kernel y después se utiliza **mkbootdisk** para crear un CD-ROM de arranque. Para un disco de arranque, tiene la opción de crear un disco flexible directamente o una imagen de disco flexible.

Si está utilizando una versión configurada reducida del kernel que cabrá en un disco flexible de 1.44 MB, se utilizan las opciones **bzdisk** o **zdisk** para compilar el kernel e instalarlo directamente en un disco flexible. Necesitará colocar un disco en su unidad de disco flexible. Una configuración estándar del kernel 2.6 es muy grande y no cabe en un disco flexible, aunque las versiones de 2.4 sí cabrán.

Para una imagen de disco flexible, tiene la opción de crear una imagen de 1.44 o 2.88 (que almacenará el kernel 2.6). Utilice las opciones **fdimage** para una imagen 1.44 y la **fdimage288** para la imagen 2.88. **fdimage** y **fimage288** crean imágenes de disco flexible correspondientes en el directorio **arch/i386/boot**. Utilizan su propia configuración **mtools.conf** ubicada en el directorio para generar las letras para la imagen de disco flexible, que **mcopy** después utiliza para crear las imágenes. La imagen **fdimage288** a menudo es utilizada por usuarios virtuales.

```
make bzdisk
make fdimage
make fdimage288
```

SUGERENCIA *Si está experimentando con sus configuraciones de kernel, será más seguro poner una nueva versión en un CD-ROM de arranque, en lugar de instalarlo en su sistema. Si algo sale mal, siempre se puede arrancar normalmente con su kernel original que está todavía en su sistema (aunque siempre se configura su cargador de arranque para acceder a las versiones anteriores).*

Configuraciones de cargador de arranque: GRUB

Si está utilizando un cargador de arranque como GRUB o LILO, puede configurar su sistema para permitirle iniciar cualquiera de sus kernels instalados. Como se vio en la sección anterior, “Pasos preventivos para modificar un kernel de la misma versión”, se crea una entrada agregada en el archivo de configuración del cargador de arranque para su viejo kernel. A medida que instale nuevas versiones del kernel, sólo se agregan más entradas, que le permiten utilizar cualquiera de los kernels antiguos. Siempre que arranque, su cargador le presentará una lista de kernels para escoger. Por ejemplo, se instala una versión de desarrollo del kernel, junto con una versión estable actual, mientras mantiene su versión anterior. En la línea de imagen para cada entrada, debe especificar el nombre de archivo del kernel. Se crea otra entrada de cargador de arranque para su kernel anterior.

En el siguiente ejemplo, el archivo de configuración Grub (**/boot/grub/menu.lst**) contiene entradas para dos kernels de Linux, una para el kernel instalado antes, **2.6.20.3**, y otra para el kernel más reciente, **2.6.21.1**. Con GRUB, sólo tiene que agregar una entrada nueva para el nuevo kernel.

```
#
#boot=/dev/hda
default=0
timeout=30
splashimage= (hd0,2)/boot/grub/splash.xpm.gz
title Nuevo Linux (2.6.21.1)
    root (hd0,2)
    kernel /boot/vmlinuz-2.6.21.1 ro root=/dev/hda3
    initrd /boot/initrd-2.6.21.1.img
title Linux Viejo (2.6.20.3)
    root (hd0,2)
    kernel /boot/vmlinuz-2.6.20.3 ro root=/dev/hda3
    initrd /boot/initrd-2.6.20.3.img
title Windows XP
    rootnoverify (hd0,0)
    chainloader +1
```

En caso de que su directorio raíz se instale en una partición LVM lógica, el nombre LVM se utiliza como un volumen Logical01 en un grupo de volúmenes Logical00. Su entrada **kernel** tendrá un aspecto parecido a éste:

```
kernel /vmlinuz-2.6.21.1 ro root=/dev/Logical00/Logical101
```

Discos RAM de módulos

Si su sistema utiliza ciertos dispositivos de bloque sin soporte en el kernel, como dispositivos SCSI, RAID o IDE, necesitará cargar ciertos módulos obligatorios cuando arranque. Estos módulos de dispositivo de bloque se almacenan en un disco RAM al que se accede cuando su sistema arranca (los discos RAM también se utilizan para sistemas sin discos). Por ejemplo, si tiene un disco duro

SCSI o CD-ROM, los controladores SCSI de éstos a menudo se almacenan en módulos que se cargan cuando su sistema inicia. Estos módulos se almacenan en un disco RAM que se lee durante el proceso de inicio. Si crea un nuevo kernel que necesita cargar módulos para iniciar, debe crear un nuevo disco RAM para esos módulos. Sólo necesita crear un nuevo disco RAM si su kernel tiene que cargar módulos en el arranque. Si, por ejemplo, se utiliza un disco duro SCSI pero incorporó un disco duro SCSI y soporte a CD-ROM (al incluir soporte al modelo específico) directamente en su kernel, no necesita configurar un disco RAM (el soporte para casi todos los discos duros IDE y CD-ROM ya está incorporado directamente en el kernel).

Si necesita crear un disco RAM, se utiliza el comando **mkinitrd** para crear un archivo de imagen de disco RAM. El comando **mkinitrd** incorpora todos los módulos IDE, SCSI y RAID que su sistema utiliza, incluidos los que aparecen en su archivo **/etc/modules.conf**. Consulte las páginas Man para **mkinitrd** y la documentación de discos RAM para conocer más detalles. **mkinitrd** toma como argumentos el nombre del archivo de la imagen de disco RAM y el kernel de donde se toman los módulos. En el siguiente ejemplo, una imagen de disco RAM llamada **initrd-2.6.versión.img** se crea en el directorio **/boot**, al utilizar módulos del kernel 2.6.21. El kernel 2.6.21 ya debe estar instalado en su sistema y deben estar creados sus módulos.

```
# mkinitrd /boot/initrd-2.6.versión.img 2.6.21.1
```

Puede seleccionar ciertos módulos para que se carguen antes o después de cualquier módulo SCSI. La opción **--preload** se carga antes de los módulos SCSI, y **--with** se carga después. Por ejemplo, para cargar el soporte RAID5 antes de los módulos SCSI, se utiliza **--preload=raid5**:

```
mkinitrd --preload=raid5 raid-ramdisk 2.6.21.1
```

En el segmento **/etc/grub/menu.lst** del nuevo kernel, coloque una entrada **initrd** que especifica el nuevo disco RAM:

```
initrd /boot/initrd-2.6.21.1.img
```

Virtualización

Ahora existen varios métodos de virtualización disponibles para uso en Linux. Éstos van de la implementación de paravirtualización empleada por Xen a la aceleración de hardware usada por Kernel-Based Virtualization Machine (KVM), en el caso de procesadores Intel y AMD compatibles con virtualización de hardware. Incluso se utiliza emulación de software. Todos éstos se instalan y administran de manera sencilla con Administrador de máquina virtual, una herramienta basada en GNOME que proporciona una interfaz GUI simple para administrar sus máquinas virtuales e instalar nuevas. Linux también proporciona la applet GNOME VM, **gnome-applet-vm**, una applet de panel que monitorea sus máquinas virtuales. Consulte la tabla 32-2 para conocer una lista de recursos de virtualización. Visite **virt.kernelnewbies.org** para conocer vínculos sobre virtualización y una revisión general.

Todos los métodos de virtualización se instalan y administran con el Administrador de máquina virtual. Éste simplifica mucho el proceso de instalar y administrar sistemas operativos virtuales (sistemas operativos invitados). Con unos cuantos pasos puede instalar Windows u otras distribuciones de Linux en su sistema Linux y ejecutarlos como sistemas operativos invitados cuando los necesite. Los hosts virtuales KVM se ejecutarán directamente desde el procesador, y serán tan rápidos y estables como si los hubiera instalado de forma separada con una configuración de arranque dual.

Recurso	Descripción
virt.kernelnewbies.org	Documentación de virtualización de kernel
virt-manager.et.redhat.com/	Administrador de máquina virtual, virt-manager
xensource.com	Sitio Web de paravirtualización de Xen
fabrice.bellard.free.fr/qemu/	Virtualización de software QEMU
kvm.qumranet.com/kvmwiki	Virtualización de hardware KVM
libvirt.org/	Conjunto de herramientas libvirt para acceder a capacidades de virtualización de Linux

TABLA 32-2 Recursos de virtualización

Existen dos métodos principales, completa y para-virtualización. La virtualización completa (KVM o QEMU) ejecuta un sistema operativo invitado independiente, mientras que la para-virtualización (Xen) requiere que primero arranque un kernel de Linux Xen de donde se lanza un sistema operativo invitado para-virtualizado. Esto significa que un sistema operativo virtualizado completo se inicia con Administrador de máquina virtual de un kernel de Linux normal, mientras un sistema operativo para-virtualizado debe arrancar con un kernel Xen.

Administrador de máquina virtual: virt-manager (Red Hat)

El Administrador de máquina virtual es un proyecto desarrollado por Red Hat y que recibe soporte de éste; actualmente está disponible en muchas distribuciones similares. Las máquinas virtuales se administran y configuran de manera sencilla con el Administrador de máquina virtual (**virt-manager**). Asegúrese de que **virt-manager** está instalado. Esto desplegará una ventana que presenta una lista de sus máquinas virtuales. Se desplegarán características como ID, nombre, estado, CPU y uso de memoria de la máquina. Se utiliza el menú Vista para determinar qué características desplegar. Haga clic en la entrada Ayuda, en el menú Ayuda, para mostrar un manual detallado del Administrador de máquina virtual.

Para conocer información detallada acerca de la máquina host, haga clic en Detalles del equipo en el menú Editar. El panel Repaso mostrará información como el nombre de host, el número de CPU que tiene y el tipo de hipervisor que lanza. El panel Redes virtuales muestra sus redes virtuales, que presentan información de conexión IPv4, el nombre de dispositivo y el nombre de red. Ya estará configurada una red virtual predeterminada. Seleccione el invitado y haga clic en el botón Detalles para ver información del invitado.

Para crear una máquina virtual, seleccione Nueva máquina, del menú Archivo. Esto iniciará el asistente **virt-install**. Se le pedirá el nombre, el tipo de virtualización, la ubicación del disco o los archivos de instalación del sistema operativo, el almacenamiento que se utilizará para el sistema operativo invitado y la cantidad de memoria de sistema asignada a éste.

Después de insertar el nombre, seleccione el método de virtualización. Si está ejecutando un kernel estándar, sólo tendrá la opción de utilizar un método virtualizado completo. En sistemas con procesadores Intel VT y AMD SVM también tendrá la opción de habilitar aceleración de hardware. Esto significa utilizar el soporte KVM (Kernel-based Virtualization Machine) que proporcionará virtualización de hardware en el nivel del procesador. En el caso de procesadores sin soporte de virtualización de hardware, se utiliza una emulación de software.

Si, en cambio, está ejecutando el Administrador de máquina virtual del kernel Xen (como Domain 0), se utiliza paravirtualización. En el caso de versiones de sistema operativo invitado modificadas especialmente para que las utilice Xen, ese sistema operativo invitado se ejecuta con

virtualización empleada según se requiera. Además, para Intel VT y AMD SVM, los métodos HVM de Xen se utilizan para emplear virtualización de hardware cuando se necesita aquélla.

SUGERENCIA *En el caso de un sistema con soporte extenso a procesador y memoria, incluso se ejecutan o instalan sistemas operativos invitados simultáneamente al utilizar KVM para Administrador de máquina virtual.*

Después se selecciona la ubicación del medio de instalación del sistema operativo. En el caso de uno totalmente virtualizado, puede ser una imagen de disco o un CD/DVD-ROM, como un disco de instalación de Windows. Entonces se selecciona el tipo de sistema operativo que está instalando, primero al seleccionar la categoría como Linux o Windows, y después una distribución particular como Centos de Linux o XP de Windows. Para un sistema operativo paravirtualizado (Xen), se selecciona una ubicación de red para el medio de instalación.

Luego puede seleccionar un método de almacenamiento. Esto es ya sea una partición existente o un archivo. Si selecciona un archivo, se establece un tamaño fijo (como una partición fija), o se hace que el archivo se expanda cuando se necesite. Si el archivo está en una partición con una gran cantidad de espacio, tal vez no sea un problema. Al principio, el archivo será de 4 GB, aunque tal vez quiera hacerlo más grande para permitirle un uso regular.

Entonces puede seleccionar una red virtual o un dispositivo físico para su conexión de red. Después seleccione la cantidad de memoria de sistema asignada a cada máquina virtual, además del número de CPU virtuales que habrá de utilizarse. Una pantalla final despliega toda su información de configuración para la nueva máquina virtual antes de iniciar la instalación. Todavía se puede cancelar en este punto. Cuando inicia la instalación, se despliega la ventana de instalación del sistema operativo invitado y se instala como se haría normalmente. Un sistema operativo instalado se ejecuta en la ventana de la consola de la máquina virtual. Existen botones para ejecutar, pausar o apagar el sistema operativo.

SUGERENCIA *También se administran sus máquinas virtuales desde la línea de comandos con **virsh**.*

Kernel-based Virtualization Machine (KVM): virtualización de hardware

Con la versión 2.6.21, ahora la virtualización de hardware se soporta directamente en el kernel (las versiones anteriores utilizan un módulo de kernel). Intel y AMD implementan la virtualización de hardware como una capa de abstracción Hardware Virtual Machine. Los procesadores Intel que tienen soporte a virtualización de hardware se etiquetan SVM (Secure Virtual Machine, máquina virtual segura). Un sistema HVM tiene la capacidad de proporcionar virtualización completa, sin necesitar versiones especialmente modificadas de un kernel de sistema operativo, como el que usa el método de paravirtualización de Xen. Incluso se ejecuta Windows XP directamente desde Linux al utilizar la capacidad de HVM. KVM es un proyecto de fuente abierta desarrollado por Qumranet, kvm.qumranet.com/kvmwiki. Asegúrese también de revisar la información de KVM en la documentación de Virtualization para su distribución.

SUGERENCIA *KVM se ejecuta con una versión modificada de QEMU, que tiene un soporte limitado a dispositivo virtual, como el controlador de gráficos (Xen tiene acceso nativo completo a controlador de dispositivos).*

KVM utiliza la virtualización de hardware en un procesador para ejecutar una máquina virtual directamente desde el hardware. No existe traducción de software; mientras que Xen trabajará a través de un kernel de dominio 0, KVM opera directamente con el procesador.

Los requisitos de hardware son los siguientes:

- Un procesador Intel (VT) o AMD (SVM) con virtualización habilitada (como el conector de procesador AM2 de AMD o los procesadores Core2Duo de Intel). Tal vez necesite habilitar el soporte de virtualización en su tarjeta madre. Algunas tarjetas funcionarán mejor que otras. En algunos casos, quizás tenga que deshabilitar el soporte a ACPI en el BIOS de la tarjeta madre para permitir que Windows XP se ejecute.
- Al menos 1 GB de memoria de sistema para tener espacio suficiente para ejecución del sistema operativo virtual. El sistema operativo virtual de hardware requiere su propia memoria.

KVM se lanza como un proceso directamente desde el kernel de Linux, como si arrancara en un nuevo sistema operativo. Se administra como cualquier proceso de Linux. KVM agrega un modo de proceso de búsqueda con su propio modo de usuario y kernel. Esto es una adición a los modos del kernel y usuario de Linux. KVM utiliza su propio controlador de dispositivo para interactuar con el hardware de virtualización del procesador, `/dev/kvm`. KVM utiliza los módulos kernel `kvm-intel` o `kvm-amd` para interactuar con el hardware de virtualización del procesador. Una versión modificada de un emulador de software QEMU se utiliza para ejecutar el sistema operativo invitado. QEMU se creó como un emulador y también está disponible para procesadores que no tienen virtualización de hardware. Visite fabrice.bellard.free.fr/qemu para conocer más información acerca de QEMU. No tiene que instalar Xen para ejecutar KVM; KVM se ejecuta por separado, aunque la instalación de Xen no interferirá con las instalaciones de KVM.

NOTA Puede implementar KVM de forma manual al utilizar QEMU, un emulador de procesador para virtualización completa. En primer lugar se crea un archivo de imagen para el nuevo sistema operativo con `qemu-img`. Después se utiliza `kvm` para iniciar el sistema operativo invitado.

Asegúrese de arrancar en el kernel estándar, no en el de Xen. Inicie Administrador de máquina virtual en su escritorio GNOME. Seleccione Nueva máquina, del menú Archivo. Esto inicia el asistente **virt-install**. Después seleccione el tipo de virtualización que habrá de usarse, seleccione Fully virtualized y asegúrese de que esté seleccionada la aceleración de hardware (Enabled kernel /hardware acceleration). Después se le piden varias características como el nombre, la cantidad de memoria de sistema, si se utiliza una partición determinada o un archivo de imagen junto con el tamaño de archivo, soporte gráfico y dónde se ubica la imagen de instalación (puede ser un CD /DVD-ROM, aunque para Windows se prefiere un archivo de imagen).

Una vez instalado, se utiliza Administrador de máquina virtual para iniciar su sistema operativo invitado en cualquier momento. Éste se ejecuta en una consola de máquina virtual.

NOTA Para acceder a los datos directamente en sus discos o archivos virtuales, se utiliza `lvm` o `lvmtools`.

Kernel de virtualización de Xen

Las distribuciones suelen proporcionar versiones del kernel que incorporan Xen Virtualization. Esta tecnología permite ejecutar diferentes sistemas operativos en un sistema Linux, además de versiones virtuales del kernel para probar nuevas aplicaciones. Xen es un proyecto de fuente abierta desarrollado originalmente por el Laboratorio de Cómputo de la Universidad de Cambridge, en coordinación con Open Source Development Labs y varios distribuidores de Linux. Conocerá más de Xen si visita cl.cam.ac.uk/Research/SRG/netos/xen. Xen.source administra el desarrollo de Xen;

se trata de un servicio comercial que proporciona versiones de fuente abierta gratuitas de Xen e implementaciones comerciales con soporte. Aquí se encuentra documentación detallada acerca de lo más reciente de los lanzamientos de Xen, xensource.com.

NOTA VMware proporciona una versión gratuita de su servidor de virtualización. También puede comprar el escritorio de virtualización para instalar otros sistemas operativos, además del servidor de virtualización ESX, que es mucho más estable y eficiente.

En un solo servidor Xen se ejecutan varias máquinas virtuales con diferentes sistemas operativos al mismo tiempo. VMware proporciona actualmente virtualización comercial, aunque también proporciona una versión gratuita de su servidor de virtualización. Xen es un sistema paravirtualizado, lo que significa que el sistema operativo invitado tiene que modificarse para ejecutarse en Xen. No se ejecuta sin modificación, como lo haría en un sistema totalmente virtualizado como VMware. Xen utiliza un método de paravirtualización para incrementar la eficiencia, al darles a sus máquinas virtuales casi el mismo nivel de eficiencia que el kernel nativo. Esto hace que la virtualización sea práctica para sistemas a nivel empresarial. Entre algunas de las ventajas que se citan para Xen se encuentra la configuración de sistemas de prueba separados, que aíslan servidores en máquinas virtuales en el mismo sistema, permitiendo a la máquina virtual acceder al soporte de hardware proporcionado por el kernel nativo. Para que un sistema operativo trabaje en Xen, se debe configurar para que acceda a la interfaz de Xen. Actualmente sólo los sistemas operativos Linux y Unix se configuran para ser compatibles con Xen, aunque se está trabajando para llevarla a Windows.

Para utilizar el kernel de Xen, primero tiene que instalar el paquete de kernel de Xen, además del servidor, las herramientas y la documentación de Xen. Existe un paquete de Xen, que incorpora soporte para ejecutar Xen en un dominio 0 (`xen0`), como servidor, y para acceso de usuario sin privilegios (`xenU`). La documentación detallada estará en `/usr/share/doc/xen-versión`. Los archivos de configuración se colocarán en el directorio `/etc/xen`, y los kernels correspondientes en los directorios `/boot` y `/lib/modules`. En el directorio `/etc/xen` encontrará el archivo `xend-config` para configurar el servidor Xen `xend`, además de un ejemplo de archivos de configuración de Xen.

Una vez que el paquete está instalado, se reinicia y se selecciona el kernel en la pantalla de GRUB. Su kernel original estándar también estará en la lista, que se selecciona para regresar al kernel normal. Al seleccionar Xen se iniciará el kernel de Xen con la habilidad de crear sus sistemas virtuales basados en Xen. Todo se verá exactamente igual. Si tiene la applet de monitor VM de GNOME en ejecución, ahora detectará un dominio 0.

Xen configura máquinas virtuales separadas llamadas dominios. Cuando el kernel de Xen inicia, crea un dominio principal, `domain0`, que administra su sistema y configura máquinas virtuales para otros sistemas operativos. El servidor `xend` administra sus máquinas virtuales. Su kernel nativo se instala en `domain0`, que manejará casi todos los dispositivos de hardware para todas las demás máquinas virtuales.

El servidor `xend` controla los dominios. Los mensajes `xend` se colocan en el archivo `/var/log/xend.log`. El servidor `xend` debe iniciarse automáticamente cuando inicia con el kernel de Xen.

NOTA Xen también proporciona soporte a Hardware Virtual Machine (HVM), la capa de abstracción HVM que Intel está implementando en sus nuevos procesadores como VT-x de Intel. AMD implementará HVM como SVM. El archivo de configuración de ejemplo para HVM en el directorio `/etc/xen` tiene la extensión `.hvm`. En este archivo, las opciones se configuran para detectar y utilizar HVM. La secuencia de comandos `virt-install` también revisa HVM.

XenMan

En casi todas las distribuciones, incluidas Debian, Fedora, SUSE y Ubuntu, se utiliza XenMan para administrar sus máquinas virtuales de Xen (xenman.sourceforge.net). La herramienta XenMan proporciona una interfaz de escritorio en que se administran sus dominios Xen, al agregar nuevos o eliminar antiguos. El panel de instrumentos le permite revisar estadísticas como el uso de la CPU y memoria. Puede manejar cada inicio, detención y reinicio de la máquina virtual. Incluso se guarda una instantánea de una máquina y se restaura en ese puerto. Para iniciar XenMan, inserte el comando **xenman** en una ventana de terminal. El archivo de configuración de XenMan es **xenman.conf**.

Los usuarios tienen su propio directorio **.xenman**. Las definiciones globales se almacenan en el archivo **/etc/xenman/xenman.conf**. El archivo de configuración define la ruta a sus dispositivos de bloque virtuales, la ubicación de las instantáneas y dónde se ubican los archivos de configuración de Xen. También proporciona entorno (dominio actual), cualquier configuración de aplicación y configuración de cliente como soporte de GNOME. XenMan también soporta el uso de imágenes, que se coleccionan en un almacén de imágenes. Una imagen es una máquina virtual predefinida en que se pueden generar otras varias máquinas virtuales. Se pueden configurar imágenes de Fedora y Ubuntu, y después generar varias máquinas virtuales de Fedora o Ubuntu que utilizan esas imágenes.

virt-install (Red Hat)

En Red Hat, Fedora y distribuciones similares, en vez de configurar un archivo directamente o utilizar Administrador de máquina virtual, se utiliza la secuencia de comandos **virt-install**. Actualmente sólo puede instalarse desde una ubicación de red remota al utilizar un prefijo **http://**, **nfs://** o **ftp://**. Esta secuencia de comandos no permitirá el uso de menos de 256 MB para cada máquina virtual. Si quiere utilizar menos memoria, digamos para una versión más pequeña de Linux, tendrá que utilizar los archivos de configuración directamente como se describió en la sección anterior.

Si tiene una cantidad limitada de memoria RAM, tal vez necesite limitar la cantidad que utiliza la máquina virtual **domain 0**. Se reduce al tamaño recomendado de 256 MB con el siguiente comando:

```
xm mem-set 0 256
```

Para instalar la secuencia de comandos, **virt-install**, abra una ventana terminal e inserte un nombre de secuencia de comandos.

```
virt-install
```

Se le pedirá que configure parámetros, y después se generará automáticamente un archivo de configuración. Primero se le pedirá el nombre de la máquina virtual. Éste es su nombre de host. Después se le pregunta cuánta memoria RAM asignar. Se requiere un mínimo de 256 MB. Esto significa que debe tener al menos 500 MB de RAM para una máquina virtual, 256 MB para el servidor Xen0 (dominio 0) y 256 MB para la máquina del invitado/usuario. Más máquinas virtuales usarían de manera correspondiente más RAM.

Luego se le pide la ruta de disco al archivo de imagen de máquina virtual. Inserte la ruta con el nombre del archivo de imagen. Después se le pide el tamaño del archivo de imagen en gigabytes. Las máquinas virtuales utilizan un archivo de imagen donde se almacena todo su sistema. Por último, se le pide una ubicación para la instalación de los archivos del sistema operativo que quiere instalar. Aquí se inserta un sitio FTP, Web o NFS. Tenga en cuenta que la secuencia de comandos busca las imágenes de kernel compatibles con Xen en el directorio **images/Xen** de la distribución. Su ubicación de descarga tiene que incluirse antes que este directorio. En el siguiente ejemplo se muestra el directorio en línea de Fedora 7:

```
# virt-install
What is the name of your virtual machine? mi-nuevovml
How much RAM should be allocated (in megabytes)? 256
What would you like to use as the disk (path)? /home/mi-nuevovml
How large would you like the disk to be (in gigabytes)? 8
Would you like to enable graphics support? (yes or no) yes
What is the install location?
http://download.fedora.redhat.com/pub/fedora/linux/7/i386/os/
```

Una vez que se han descargado los archivos, inicia la interfaz de instalación basada en texto, preguntándole el teclado y el idioma. Si habilita el soporte de gráficos, iniciará la instalación gráfica estándar.

Creación de máquinas virtuales de Xen con archivos de configuración

Los usuarios avanzados crean una máquina virtual para configurar y editar directamente el archivo de configuración de la máquina Xen. En lugar de utilizar Administrador de máquina virtual o **virt-install** para generar su configuración e instalar su sistema operativo, se establecen opciones más refinadas. Primero tiene que crear el archivo de configuración de la máquina en el directorio **/etc/xen**. En este directorio encontrará archivos de configuración de ejemplo que se utilizan como plantilla. Existen ejemplos de configuración de Xen estándar, además de configuración para implementaciones de HVM de Xen. Aquí están las configuraciones que tal vez quiera cambiar:

- **kernel** La ruta a la imagen de kernel utilizada por la máquina virtual.
- **root** El dispositivo raíz para el dominio.
- **memory** La cantidad de memoria que podrá utilizar el dominio.
- **disk** Los dispositivos de bloque (particiones) que quiere que el dominio use.
- **dhcp** Hace que el dominio utilice DHCP para establecer redes; para configuración manual, se establecen parámetros de máscara de red y puerta de enlace.
- **hostname** El nombre de host de la máquina virtual.
- **vif** La dirección MAC que se utiliza (se generarán de manera aleatoria, si no se especifica una).
- **extra** Parámetros de arranque adicionales.
- **restart** Opciones de reinicio automático: always, never, onreboot.

Tendrá que configurar el kernel. Ésta es la ubicación del kernel xenU. La entrada root especifica la partición donde está almacenada la imagen de arranque. La entrada disk le permite especificar los discos que utilizarán los nuevos dominios. Éstos son volúmenes lógicos o archivos de imagen de disco. No deben estar montados por el dominio principal. Observe que **xenguest-install**, que se describe más adelante, configurará una imagen de disco que será utilizada por la máquina virtual, en lugar de un volumen lógico. En el siguiente ejemplo se utiliza una imagen de disco **/home/minuevovml**, que aparecerá para la máquina virtual como **xvda** y tendrá acceso de lectura/escritura:

```
disk = [ 'phy:/home/minuevovml, xvda,w' ]
```

Además, tal vez quiera habilitar **dhcp** si su red la utiliza para configurar una conexión de red. Por ejemplo, la configuración del kernel se vería así:

```
kernel = "/boot/vmlinuz-2.6.21.1
```

Se utiliza el comando **xm** con la opción **create** seguida por el archivo de configuración. En este ejemplo hay un archivo de configuración `/etc/xen` llamado **mi-nuevovm1**, que es el mismo nombre de la máquina virtual. Revise la página Man para **xmdomain.cfg** para conocer opciones y ejemplos de configuración detallados.

```
xm create mi-nuevovm1
```

Entonces se conecta con la opción **console**.

```
xm console mi-nuevovm1
```

Se combinan los dos comandos **xm** al utilizar la opción **-c** para la conexión de consola.

```
xm create -c mi-nuevovm1
```

Todavía tendrá que instalar el sistema operativo que quiere usar. Se hace esto en una partición o un volumen lógico designado, o se utiliza **virt-install**. Se instala con un sistema basado en texto o al utilizar VNC para una instalación virtual.

Administración de máquinas virtuales de Xen con xm

Después de que ha instalado el sistema, se crea una conexión con el comando **xm**. El comando **xm** está disponible para todas las distribuciones. Para acceder a un dominio particular, se utiliza la opción **console** y el nombre de dominio.

```
xm console mi-nuevovm1
```

Si el dominio ya no existe, también tiene que crearlo y después conectarlo.

```
xm create -c mi-nuevovm1
```

Revise las opciones **xm** en la página Man **xm** para otras operaciones que se realizan con sus máquinas virtuales.

Para acceder a dominios, se utiliza el comando **xm**. La opción **list** presenta una lista de sus dominios. La lista incluirá información detallada como su ID de dominio, el tiempo de uso del CPU, la memoria utilizada y el estado del dominio. Lo siguiente presenta una lista de sus dominios:

```
xm list
```

Las opciones **xm save** y **restore** se utilizan para suspender y reiniciar un dominio.

Los dispositivos de bloques como particiones y CD-ROMs se exportan del dominio principal a dominios virtuales. Esto permite a un dominio determinado utilizar una partición particular. Incluso se comparten dispositivos de bloque entre dominios, aunque tales dispositivos compartidos deben ser de sólo lectura.

Para iniciar y detener sus dominios, se utiliza la secuencia de comandos de servicio **xendomains**. La secuencia de comandos **xendomains** utilizará **xm** con la opción **create** para crear un dominio configurado en el directorio `/etc/xen/auto`. Coloque los archivos de configuración de dominio de Xen en este directorio para iniciar **xendomains**. Estos dominios se iniciarán automáticamente cuando se inicie el kernel para domain0. El siguiente comando inicia de forma manual sus dominios:

```
service xendomains start
```

NOTA Para utilizar la memoria de bloque de forma más eficiente (particiones de disco duro), se implementa de forma dinámica el espacio asignado al utilizar VDB (dispositivos de bloque virtuales implementados como archivos) o VBD de LVM.

Administración de copia de seguridad

Las operaciones de copia de seguridad se han vuelto una parte importante de las tareas administrativas. En sistemas Linux se proporcionan varias herramientas de copia de seguridad, incluidas Anaconda y las herramientas `dump`/`restore` tradicionales, además del comando `rsync` para hacer copias individuales. Anaconda proporciona copias de seguridad basadas en servidores, permitiendo que diferentes sistemas de una red creen copias de seguridad de un servidor central. BackupPC proporciona copias de seguridad locales y de red al utilizar herramientas `rsync` y `tar`. Las herramientas `dump` le permiten refinar el proceso de copia de seguridad, al detectar datos que se cambian desde la última copia. En la tabla 33-1 se presenta una lista de los sitios Web de herramientas de copia de seguridad de Linux.

Copias de seguridad individuales: archivadores y `rsync`

Se crean copias de seguridad y se restauran archivos y directorios particulares con las herramientas de archivado como `tar`, que restaura los archivos después. En el caso de copias de seguridad, `tar` suele utilizarse con un dispositivo de cinta. Para programar copias de seguridad automáticamente, se utilizan los comandos `tar` apropiados con la utilería `cron`. Los archivadores también se comprimen para ahorrar espacio de almacenamiento. Después se copian los archiveros comprimidos en cualquier medio, como un DVD, un disco flexible o una cinta. En GNOME se utiliza File Roller para crear archiveros de forma sencilla (Gestor de archivadores, en Herramientas del sistema). La herramienta KDAT en KDE creará copias de seguridad en cintas, un client para `tar`. Consulte el capítulo 10 para conocer un análisis acerca de archiveros comprimidos.

Si quiere copiar de forma remota un directorio o varios archivos de un host a otro, al hacer un copia de seguridad particular, se utiliza `rsync`, que está diseñado para copias de seguridad de red de directorios o archivos particulares, que copia de forma inteligente sólo los archivos que han sido cambiados, en lugar del contenido completo de un directorio. En el modo `archive`, se preservan los permisos y propietarios originales, lo que permite que los usuarios correspondientes existan en el sistema host. El siguiente ejemplo copia el directorio `/home/jorge/miproyecto` en el directorio `/copiadeseuridad` del host `conejo`, al crear un subdirectorio `miproyecto` correspondiente. `-t` especifica que esto es una transferencia. Se alude al host remoto con dos puntos adjuntos, `conejo:`.

Sitio Web	Herramientas
rsync.samba.org	Copia de seguridad remota rsync
amanda.org	Copia de seguridad de red Amanda
dump.sourceforge.net	Herramientas dump y restore
backuppc.sourceforge.net	Copia de seguridad de red o local de BackupPC al utilizar herramientas rsync y tar configuradas.

TABLA 33-1 Recursos de copia de seguridad

```
rsync -t /home/jorge/miproyecto conejo:/copiadeseguridad
```

Si, en cambio, quiere preservar los permisos y propietarios de los archivos, se utiliza la opción **-a** (archivero). Al agregar una opción **-z** se comprimirá el archivo. La opción **-v** proporciona un modo de texto extenso.

```
rsync -avz /home/jorge/miproyecto conejo:/copiadeseguridad
```

Una diagonal al final de la fuente copiará los contenidos del directorio, en lugar de generar un subdirectorio con ese nombre. Aquí el contenido de **miproyecto** se copia en el directorio **jorge-proyecto**.

```
rsync -avz /home/jorge/miproyecto/ conejo:/copiadeseguridad/jorge-proyecto
```

El comando **rsync** se configura para usar la shell remota SSH como opción predeterminada. Se especifica la shell remota SSH u otra alterna con la opción **-e**. Para transmisiones seguras, se cifra la operación de copia con ssh. Use la opción **-e ssh** o establezca la variable **RSYNC_RSH** en ssh.

```
rsync -avz -e ssh /home/jorge/miproyecto conejo:/copiadeseguridad/miproyecto
```

Como cuando se utiliza **rcp**, se copia de un host remoto a otro en el que está.

```
rsync -avz lagarto:/home/marco/misimagenes/ /archivero-imagenes/imagenesmarco
```

También se ejecuta un daemon de servidor. Esto permitirá a los usuarios remotos sincronizar las copias de archivos en su sistema con otras versiones propias, al transferir sólo los archivos cambiados, en lugar de todos sus directorios. Muchos sitios de espejo y FTP de software operan como servidores rsync, que le permiten actualizar archivos sin tener que descargar nuevamente las versiones completas. La información de configuración para rsync como servidor se mantiene en el archivo **/etc/rsyncd.conf**. En Linux, rsync como servidor se administra mediante **xinetd**, al utilizar el archivo **/etc/xinetd.d/rsync**, que inicia rsync con la opción **--daemon**. En el archivo **/etc/services**, se define que se ejecute en el puerto 873. Está deshabilitada como opción predeterminada, pero se habilita con una herramienta como **chkconfig** (Fedora y SUSE) o **sysv-rc-conf** (Debian o Ubuntu).

***SUGERENCIA** Aunque está diseñado para copiar entre hosts, también se utiliza rsync para hacer copias dentro de su propio sistema, por lo general en un directorio de otra partición o disco duro. En realidad, existen ocho formas diferentes de usar rsync. Revise la página Man de rsync para conocer descripciones detalladas de cada una.*

BackupPC

BackupPC proporciona una copia de seguridad de red o local de su sistema o host que se administra de manera sencilla en un sistema al utilizar las herramientas configuradas `rsync` o `tar`. No hay que instalar una aplicación de cliente, sólo archivos de configuración. BackupPC respalda hosts en una red, incluidos los servidores, o sólo un sistema. Se crean copias de seguridad de los datos en un disco duro local o de almacenamiento de red como particiones compartidas o servidores de almacenamiento. Se configura BackupPC al utilizar su interfaz de configuración de página Web. Ésta es el nombre host de su computadora con el nombre `/backuppc` adjunto, como <http://conejo.tortuga.com/bakcuppc>. La documentación detallada se instala en `/usr/share/doc/BackupPC`. Encontrará más información acerca de BackupPC en backuppc.sourceforge.net.

BackupPC utiliza compresión y detección de archivos idénticos para reducir de manera importante el tamaño de la copia de seguridad, permitiendo que varios hosts se copien en un espacio limitado. Una vez que se realiza una copia de seguridad inicial, BackupPC sólo copiará los archivos cambiados, lo que reduce significativamente el tiempo de la operación.

BackupPC tiene su propia secuencia de comandos para iniciar el servicio BackupPC, `/etc/init.d/backuppc`. Los archivos de configuración se ubican en `/etc/BackupPC`. El archivo `config.pl` almacena opciones de configuración de BackupPC y el archivo de host incluye los hosts que se copiarán.

Amanda

Para crear copias de seguridad de los hosts conectados a una red, se utiliza Advanced Maryland Automatic Network Disk Archiver (Amanda, archivador avanzado y automático de disco de red de Maryland) para archivar hosts. Amanda utiliza herramientas `tar` para copiar todos los hosts en uno solo que opera como un servidor de copias de seguridad. Cada host envía los datos de la copia al host que opera como servidor Amanda, donde se graban en un medio, como una cinta. Con un servidor Amanda, las operaciones de copia de seguridad de todos los hosts se centralizan en un servidor, en lugar de que cada host tenga que realizar su propia copia. Cualquier host que necesite restaurar datos sólo tendrá que pedirlos al servidor Amanda, especificando el sistema de archivos, la fecha y los nombres de los archivos. Los datos de copia de seguridad se copian en el disco de almacenamiento del servidor y de ahí en las cintas. La documentación detallada y las actualizaciones las proporciona amanda.org. Para el servidor, asegúrese de instalar un paquete de servidor de amanda, y para clientes use el paquete de amanda para clientes.

Amanda está diseñado para copias de seguridad automáticas de host que tienen configuraciones muy diferentes, además de sistemas operativos. Se copia cualquier host compatible con herramientas GNU, incluidos los sistemas Mac OS X y Windows conectados mediante Samba.

Comandos de Amanda

Amanda tiene sus propios comandos que corresponden a las tareas de copia de seguridad comunes, que comienzan con “am”, como `amdump`, `amrestore` y `amrecover`. Los comandos se muestran en la tabla 33-2. El comando `amdump` es la operación de copia de seguridad principal.

El comando `amdump` realiza copias de seguridad solicitadas; no está diseñado para uso interactivo. Si desea una copia interactiva, se utiliza directamente una herramienta de archivador como `tar`. El comando `amdump` se coloca con una instrucción `cron` para ejecutarse en un momento específico. Si, por alguna razón, `amdump` no guarda todos sus datos en el medio de copia de seguridad (cinta o disco), mantendrá los datos en el disco de almacenamiento. Después, los datos se graban directamente con el comando `amflush`.

Comando	Descripción
amdump	Realiza copias de seguridad automáticas de los sistemas incluidos en el archivo de configuración disklist .
amflush	Crea copia de seguridad de directamente de los datos de un disco de almacenamiento a una cinta.
amcleanup	Realiza una limpieza si hay una falla de sistema en el servidor.
amrecover	Selecciona las copias de seguridad que se restaurarán al utilizar una shell interactiva.
amrestore	Restaura copias de seguridad, ya sea archivos o sistemas completos.
amlabel	Etiqueta el medio de copia de seguridad para Amanda.
amcheck	Revisa los sistemas de copia de seguridad y archivos, además de las cintas, antes de las operaciones de copia.
amadmin	Respalda tareas administrativas.
amtape	Administra cintas de copia de seguridad, al cargarlas y eliminarlas.
amverify	Revisa el formato de las cintas.
amverifyrun	Revisa las cintas de ejecuciones previas, especifica el directorio de configuración para la copia de seguridad.
amrmntape	Elimina una cinta de la base de datos de Amanda; se utiliza para cintas dañadas.
amstatus	Muestra el estado de la operación de copia de seguridad actual de Amanda.

TABLE 33-2 Comandos de Amanda

Puede restaurar archivos particulares, además de sistemas completos con el comando **amrestore**. Con la herramienta **amrecover**, puede seleccionar de una lista de copias de seguridad.

Configuración de Amanda

Los archivos de configuración se colocan en **/etc/amanda** y los de registro y base de datos en **/var/lib/amanda**. Éstos se crean cuando se instala Amanda. También necesita crear un directorio que se utilizará como un disco de almacenamiento donde las copias de seguridad se mantienen antes de grabarse en cinta.

Éste debe ser un sistema de archivos con espacio disponible muy grande, suficiente para mantener copias de seguridad completas de su host más grande.

/etc/amanda

Dentro del directorio **/etc/amanda** se encuentran subdirectorios para los diferentes tipos de copias de seguridad que quiera realizar. Cada directorio contendrá sus propios archivos **amanda.conf** y **disklist**. Como opción predeterminada, se crea un diario un directorio de copia de seguridad llamado **DailySet1**, con un archivo predeterminado **amanda.conf** y uno de ejemplo **disklist**. Para usarlos, tendrá que editarlos insertando sus propias configuraciones del sistema. Para configurar una copia de seguridad diferente, puede crear un nuevo directorio y una copia de los archivos **DailySet1 amanda.conf** y **disklist**, editándoles de acuerdo con las necesidades. Cuando se envían comandos de Amanda como **amdump** para realizar copias de seguridad, se utilizará el nombre del subdirectorio **/etc/amanda** para indicar el tipo de copia que quiere realizar.

```
amdump DailySet1
```


El directorio `/etc/amanda` también contiene un ejemplo de archivo `cron`, `crontab.sample`, que muestra cómo debe verse una entrada `cron`.

amanda.conf

El archivo **amanda.conf** contiene parámetros de configuración básicos como el tipo de cinta y archivo de registro, además de almacenar las ubicaciones de archivo. Casi siempre se utilizan las opciones predeterminadas, como aparecen en el archivo **DailySet1/amanda.conf**. El archivo incluye comentarios detallados, al indicarle qué entradas tendrá que cambiar. Necesitará configurar las entradas `tapedev` de acuerdo con el dispositivo de cinta que utiliza, y la entrada de tipo de cinta correspondiente. En el segmento de disco de almacenamiento, necesitará especificar la partición y el directorio para el disco de almacenamiento que quiere utilizar. Consulte las páginas Man y de documentación de Amanda para conocer información detallada acerca de varias opciones.

disklist

Es en el archivo **disklist** donde se especifican los sistemas de archivos y particiones que se respaldan. Una entrada incluye los hosts, la partición y el tipo de volcado. Los tipos de volcado posibles se definen en **amanda.conf**. Establecen ciertos parámetros como la prioridad de la copia de seguridad y si utiliza compresión o no. El tipo `comp-root` respaldará las particiones raíz con compresión y prioridad baja, mientras que el tipo `always-full` respaldará toda la partición sin compresión y la prioridad más alta. Se definen otros tipos de volcado en **amanda.conf** y se utilizan para particiones diferentes.

Las copias de seguridad se realizarán en el orden en que aparecen; asegúrese de incluir primero los más importantes. El archivo **disklist** en **DailySet1** proporciona ejemplos detallados.

Habilitación de Amanda en la red

Para utilizar Amanda en la red, necesita ejecutar dos servidores en el servidor Amanda, además de un cliente Amanda en cada host de red. El acceso debe habilitarse para los clientes y el servidor.

Servidor Amanda

El servidor Amanda se ejecuta mediante **xinetd**, al utilizar los archivos de servicio **xinetd** ubicados en `/etc/xinetd.d`. Los tres archivos de servicio son **amanda**, **amidxtape** y **amandaidx**. Entonces reinician el daemon **xinetd** para que se aplique de inmediato.

Para que los clientes puedan recuperar copias de seguridad del servidor, los nombres de host de los clientes deben colocarse en el archivo **.amandahosts**, en el directorio de los usuarios de Amanda del servidor, `/var/lib/amanda`. En el servidor, `/var/lib/amanda/.amandahosts` presentará una lista de todos los hosts copiados por Amanda.

Hosts de Amanda

Cada host debe permitir el acceso al servidor Amanda. Para esto, se coloca el nombre de host del servidor Amanda en cada archivo con punto **.amandahosts** del cliente. Este archivo se ubica en el directorio de inicio del usuario de Amanda en el cliente, `/var/lib/amanda`.

Cada host debe ejecutar el daemon de cliente, **amanda**, que también se ejecuta bajo **xinetd**. En Debian, Ubuntu y distribuciones similares, se utiliza el archivo de configuración `/etc/xinetd.d/amanda` para controlar la habilitación de Amanda. En Red Hat, SUSE y distribuciones similares, se utiliza **chkconfig** para activarlo.

```
chkconfig amanda on
```

SUGERENCIA Si su servidor y hosts tienen firewalls, debe permitir el acceso a través de los puertos que utiliza Amanda, por lo general 10080, 10082 y 10083.

Uso de Amanda

Las copias de seguridad se realizan con el comando **amdump**.

```
amdump DailySet1
```

Se coloca un comando **amdump** para cada copia de seguridad en el archivo **crontab** de Amanda. Es útil ejecutar una operación **amcheck** para asegurarse de que la cinta está lista.

```
0 16 * * 1-5 /usr/sbin/amcheck -m DailySet1
45 0 ** 2-6 /usr/sbin/amdump DailySet1
```

Antes de que se utilice una cinta, tendrá que etiquetarla con **amlabel**. Amanda utiliza la etiqueta para determinar qué cinta debe utilizar para una copia de seguridad. Inicie sesión como usuario de Amanda (no raíz) y etiquete la cinta para que pueda utilizarse.

```
amlabel DailySet DailySet1
```

Un cliente recupera una copia de seguridad al utilizar **amrecover**. Debe ejecutarse como usuario root, no como uno de Amanda. El comando **amrecover** trabaja con una shell interactiva casi igual a **ftp**, que le permite mostrar los archivos disponibles y seleccionarlos para restauración. Dentro de la shell **amrecover**, el comando **ls** mostrará los copias de seguridad disponibles, el comando **add** seleccionará una, y la operación de extracción lo restaurará. El comando **lcd** le permite configurar el directorio de cliente; **amrecover** utilizará **DailySet1** como opción predeterminada, pero para otras configuraciones necesitará especificar su directorio de configuración con la opción **-C**. Si tiene más de un servidor Amanda, se muestra sólo el que quiere con la opción **-t**.

```
amrecover -C DailySet1
```

Para restaurar las copias de seguridad de todo el sistema, se utiliza el comando **amrestore**, con especificación del dispositivo de cinta y el nombre de host.

```
amrestore /dev/rmt1 conejo
```

Para seleccionar ciertos archivos, se canaliza la salida a un comando de recuperación como **restore** (que se analiza en la siguiente sección).

```
amrestore -p /dev/rmt1 conejo midir | restore -ibvf 2 -
```

Copias de seguridad con dump y restore

Se crean copias de seguridad y se restaura su sistema con las utilerías **dump** y **restore**. **dump** copia su sistema completo o realiza copias incrementales, al guardar sólo los archivos que han sido cambiados desde la última copia. **dump** permite varias opciones para administrar la operación de copia de seguridad, como especificar el tamaño y la cantidad del medio de almacenamiento (consulte la tabla 33-3).

NOTA Existen varias herramientas de volcado de discos. El comando **diskdumpfmt** se usa para dar formato a cintas que se utilizarán para volcado. **diskdumpctl** registra una partición de volcado con el sistema. **savecore** guarda un archivo **vmcore** de los datos en la partición de volcado. La herramienta **crash** puede leer las tareas de volcado. Revise la página Man de **crash** para conocer más detalles.

Opción	Descripción
-0 a -9	Especifica el nivel de volcado. Un nivel 0 es una copia de seguridad de todo el sistema de archivos (revise también la opción -h). Los números de nivel de volcado superior de 0 realizan copias de seguridad incrementales, que copian todos los archivos nuevos o modificados en el sistema de archivos desde la última copia en un nivel bajo. La opción predeterminada es 9.
-B registros	Permite especificar el número de bloques en un volumen, al invalidar la detección de fin del medio o las cantidades de tamaño y densidad que dump normalmente utiliza para volcados en varios volúmenes.
-a	Permite a dump omitir cualquier cálculo de tamaño de cinta y grabar hasta que se detecte una indicación de final de medio. Se recomienda para casi todas las unidades de cinta modernas y es la opción predeterminada.
-b tamaño del bloque	Permite especificar el número de kbytes por registro de volcado. Con esta opción, se crean bloques más largos, al acelerar las copias de seguridad.
-d densidad	Especifica la densidad de una cinta en bits por pulgada (la opción predeterminada es 1600BPI).
-h nivel	Los archivos que se etiquetan con una marca nodump del usuario no se respaldarán en este nivel especificado o uno superior. El nivel predeterminado es 1, que no copiará los archivos etiquetados en copias de seguridad incrementales.
-f archivo/dispositivo	Crea una copia de seguridad del sistema de archivos para especificar el archivo o dispositivo. Puede ser un archivo o una unidad de cinta. Se especifican varios nombres de archivos, separados por comas. Se hace referencia a un dispositivo remoto o archivo al antecederlo con el nombre de <i>nombrehost:archivo</i> .
-k	Utiliza la autenticación Kerberos para comunicarse con servidores de cinta remotos.
-M archivo/dispositivo	Implementa una copia de seguridad de varios volúmenes, donde el archivo en que se escribirá se trata como un prefijo, y el sufijo, que consta de una secuencia numerada a partir de 001 se utiliza para cada archivo siguiente, <i>archivo001</i> , <i>archivo002</i> , etc. Es útil cuando los archivos de copia de seguridad necesitan ser mayores que el límite de tamaño de archivo de 2 GB ext3 de Linux.
-n	Notifica a los operadores si una copia de seguridad necesita su atención.
-s pies	Especifica el tamaño de una cinta en pies. dump pedirá una nueva cinta cuando se llegue a este tamaño.
-S	Estima la cantidad de espacio necesaria para realizar una copia de seguridad.
-T date	Le permite especificar su propia fecha, en lugar de utilizar el archivo /etc/dumpdates .
-u	Escribe una entrada para una actualización correcta en el archivo /etc/dumpdates .
-W	Detecta y despliega los sistemas de archivos que deben respaldarse. Esta información se toma de los archivos /etc/dumpdates y /etc/fstab .
-w	Detecta y despliega los sistemas de archivos que deben respaldarse, al incluir sólo información en /etc/fstab .

TABLA 33-3 Opciones de dump

Niveles de volcado

La utilidad `dump` utiliza *niveles de volcado* para determinar a qué grado quiere que se creen copias de seguridad de su sistema. Un nivel de volcado 0 copiará sistemas de archivos en su totalidad. Los niveles restantes realizan copias incrementales, al copiar sólo los archivos y directorios que se han creado o modificado desde la última copia de seguridad de nivel más bajo. Un nivel de volcado 1 copiará sólo los archivos que han cambiado desde la última copia de nivel 0. El nivel de volcado 2, en cambio, sólo copia los archivos que han cambiado desde la última copia de seguridad de nivel 1 (o 0 si no hay nivel 1), y así hasta el nivel 9. Se ejecuta una copia completa inicial en un nivel de volcado 0 para crear copias de seguridad de todo su sistema y ejecutar copias incrementales en ciertas fechas posteriores, copiando sólo los cambios hechos desde la última copia completa.

Al utilizar niveles de volcado, se diseñan varias estrategias para respaldar un sistema de archivos. Es importante tener en cuenta que una copia de seguridad incremental se realiza a partir de los cambios tras la última copia de nivel inferior. Por ejemplo, si la última copia fue 6 y la siguiente fue 8, entonces el nivel 8 copiará todo desde el nivel de copia de seguridad 6. La secuencia es importante. Si hubiera tres copias de seguridad con niveles 3, 6 y 5, el nivel de copia de seguridad 5 tomaría todo desde el nivel 3, sin detenerse en el nivel 6. El nivel 3 es el nivel *inferior* siguiente al nivel 5, en este caso. Esto crea estrategias para algunas copias de seguridad incrementales complejas. Por ejemplo, si quiere que cada copia siguiente incluya todos los cambios de las copias incrementales anteriores, puede ejecutar las copias en orden de nivel de volcado descendente. Dada una secuencia de copias de seguridad de 7, 6 y 5, con 0 como nivel inicial de copia de seguridad completo, 6 incluirá todos los cambios a 7, porque su nivel inferior siguiente es 0. Después 5 incluirá todos los cambios a 7 y 6, también porque su nivel inferior siguiente es 0, incluyendo todos los cambios desde la copia de seguridad completa de nivel 0. Una forma más simple de implementar esto es hacer que todos los niveles incrementales sean iguales. Al dar un nivel inicial 0 y después dos copias de seguridad para nivel 1, el último nivel 1 incluirá todos los cambios de la copia de seguridad con nivel 0, ya que el nivel 0 es el nivel *inferior* siguiente (no el nivel 1 de copia de seguridad anterior).

Grabación de copias de seguridad

Las copias de seguridad se registran en el archivo `/etc/dumpdates`. Este archivo mostrará todos los copias de seguridad anteriores, que especifican los sistemas de archivos donde se realizaron, las fechas en que se realizaron y el nivel de volcado que se utilizó. Se usa esta información para restaurar archivos de una copia de seguridad específica. Recuerde que el archivo `/etc/fstab` registra los niveles de volcado, además de la frecuencia de copia de seguridad recomendada para cada sistema de archivos. Con la opción `-w`, `dump` analizará los archivos `/etc/dumpdates` y `/etc/fstab` para determinar qué sistemas de archivos necesita copiar. El comando `dump` con la opción `-w` sólo utiliza `/etc/fstab` para reportar los sistemas de archivos listos para copia de seguridad.

Operaciones con dump

El comando `dump` toma como argumentos el nivel de volcado, el dispositivo donde se está almacenando la copia de seguridad y el nombre de dispositivo del sistema de archivos que se está copiando. Si el medio de almacenamiento (como la cinta) es muy pequeño para acomodar la copia de seguridad, `dump` hará una pausa y le permitirá insertar otro. `dump` permite copias de seguridad en varios volúmenes. La opción `u` registrará la copia de seguridad en el archivo `/etc/dumpdates`. En el siguiente ejemplo, una copia de seguridad completa (nivel de volcado 0) se realiza en el sistema de archivos en la partición de disco duro `/dev/hda3`. La copia de seguridad se almacena en un dispositivo de cinta, `/dev/tape`.

```
dump -0u -f /dev/tape /dev/hda3
```

NOTA Puede utilizar el comando **mt** para controlar su dispositivo de cinta; **mt** tiene opciones para regresar, borrar y colocar la cinta. El comando **rmt** controla un dispositivo de cinta remoto.

El dispositivo de almacenamiento puede ser otra partición de disco duro, pero suele ser un dispositivo de cinta. Cuando instaló su sistema, lo más probable es que haya detectado el dispositivo y haya configurado **/dev/tap** como un vínculo a éste (como lo hizo con sus CD-ROMs). Si el vínculo no se configuró, tiene que crearlo o utilizar el nombre de dispositivo directamente. Los dispositivos de cinta tienen diferentes nombres, dependiendo del modelo o la interfaz. Los dispositivos de cinta SCSI se etiquetan con el prefijo **st**, con un número adjunto al dispositivo particular: **st0** es el primer dispositivo de cinta SCSI. Para utilizarlo con el comando **dump**, sólo especifique su nombre.

```
dump -0u -f /dev/st0 /dev/hda5
```

Si necesita crear una copia de seguridad de un dispositivo ubicado en otro sistema en su red, tiene que especificar el nombre de host del sistema en el nombre de su dispositivo. El primero se inserta antes del segundo y se delimita con dos puntos. En el siguiente ejemplo, el usuario crea una copia de seguridad del sistema de archivos **/dev/hda5** al dispositivo de cinta SCSI con el nombre **/dev/st1** en el sistema **conejo.mipista.com**:

```
dump -0u -f conejo.mipista.com:/dev/st0 /dev/hda5
```

El comando **dump** trabaja en un sistema de archivos a la vez. Si su sistema tiene más de un sistema de archivos, necesitará usar un comando **dump** separado para cada uno.

SUGERENCIA Puede utilizar la utilidad de sistema **cron** para programar copias de seguridad al utilizar **dump** en momentos específicos.

Recuperación de copias de seguridad

Se utiliza el comando **restore** para restaurar un sistema de archivos completo o sólo para recuperar archivos particulares. **restore** extraerá los archivos o directorios de un archivero de copia de seguridad y los copiará al directorio de trabajo actual. Asegúrese de que está en el directorio donde quiere que los archivos se restauren cuando ejecute el comando **restore**. Éste también generará cualquier subdirectorio necesario y tiene varias opciones para administrar la operación de restauración (consulte la tabla 33-4).

Para recuperar archivos y directorios individuales, se ejecuta **restore** en un modo interactivo, al utilizar la opción **-i**. Esto generará una shell con todos los directorios y archivos en la cinta, permitiéndole seleccionar los que quiere restaurar. Cuando haya terminado, **restore** sólo recuperará de una copia de seguridad los archivos seleccionados. Esta shell tiene su propio conjunto de comandos que se utilizan para seleccionar y extraer archivos y directorios (consulte la tabla 33-5). El siguiente comando generará una interfaz interactiva que muestra todos los directorios y archivos respaldados en la cinta en el dispositivo **/dev/tape**:

```
restore -ivf /dev/tape
```

Este comando generará una shell que abarca toda la estructura de directorios de la copia de seguridad. Se le presenta un indicador de shell y se utiliza el comando **cd** para moverse a diferentes directorios y el comando **ls** para mostrar los archivos y subdirectorios. Se utiliza el comando **add** para etiquetar un archivo o directorio para la extracción. Si después decide no extraerlos, se utiliza el comando **delete** para eliminarlos de la lista etiquetada. Una vez que haya seleccionado todos

Operación	Descripción
-C	Le permite revisar una copia de seguridad al comparar los archivos de un sistema de archivos con los de la copia de seguridad.
-i	El modo interactivo para restaurar archivos y directorios en una copia de seguridad. Se genera una interfaz de shell donde el usuario utiliza comandos para especificar los archivos y directorios que habrán de restaurarse (consulte la tabla 33-5).
-R	Instruye a restore para que pida una cinta que es parte de un copia de seguridad de varios volúmenes, con lo que se continúa la operación de restauración. Es útil cuando se interrumpen las operaciones de restauración de varios volúmenes.
-r	Restaura un sistema de archivos. Asegúrese de que se ha montado una nueva partición formateada y que ha cambiado a su directorio superior.
-t	Presenta una lista del contenido de una copia de seguridad o archivos específicos en éste.
-x	Extrae archivos o directorios específicos de una copia de seguridad. Un directorio se restaura con todos sus subdirectorios. Si no se especifica un archivo o directorio, se restaura todo el sistema de archivos.
Opciones adicionales	Descripción
-b <i>tamañodebloque</i>	Especifica un tamaño de bloque; de otra forma, restore lo determinará de forma dinámica a partir del dispositivo de bloque.
-f <i>archivo/dispositivo</i>	Restaura la copia de seguridad en el archivo o sistema especificado. Especifica un nombre de host para los dispositivos remotos.
-F <i>secuenciadecomandos</i>	Ejecuta una secuencia de comandos al principio de la restauración.
-k	Utiliza la autenticación Kerberos para dispositivos remotos
-h	Extrae sólo los directorios especificados, sin sus subdirectorios.
-M <i>archivo/dispositivo</i>	Restaura a partir de copias de seguridad de varios volúmenes, donde <i>archivo</i> se trata como un prefijo y el sufijo es una secuencia numerada, <i>archivo001</i> , <i>archivo002</i> .
-N	Despliega los nombres de los archivos y directorios, no los extrae.
-T <i>directorio</i>	Especifica un directorio que se utilizará para el almacenamiento de archivos temporales. El valor predeterminado es /tmp .
-v	El modo textual extenso, donde se despliega cada archivo y su tipo de archivo en que opera restore .
-y	Como opción predeterminada, restore consultará al operador para que continúe si ocurre un error, como bloques malos. Esta opción suprime esa consulta, al permitir que restore continúe automáticamente.

TABLA 33-4 Operaciones y opciones para **restore**

los elementos que quiere, inserte el comando **extract** para recuperarlos del archivero de copia de seguridad. Para salir de la shell de restauración, inserte **quit**. El comando **help** muestra una lista de los comandos shell restaurados.

Comando	Descripción
add [arg]	Agrega archivos o directorios a la lista de archivos para extracción. Tales archivos etiquetados despliegan un * antes de sus nombres cuando se despliegan con ls . También se extraen todos los subdirectorios de un directorio etiquetado
cd arg	Cambia al directorio de trabajo actual.
delete [arg]	Elimina un archivo o directorio de la lista de extracción. También se moverán todos los subdirectorios de los directorios eliminados.
extract	Extrae archivos y directorios de la lista de extracción.
help	Despliega una lista de comandos disponibles.
ls [arg]	Presenta una lista de los contenidos del directorio de trabajo actual o un directorio especificado.
pwd	Despliega el nombre de ruta completo del directorio de trabajo actual.
quit	Sale de la shell de modo interactivo de restauración. El comando quit no realiza ninguna extracción, aunque la lista de extracción todavía tenga elementos.
setmodes	Establece el propietario, los modos y las horas de los archivos y directorios en la lista de extracción. Se utiliza para limpiar una restauración interrumpida.
verbose	En el modo verbose, todos los archivos se muestran a medida que se extraen. Además, el comando ls muestra los números inodo para archivos y directorios.

TABLA 33-5 Comandos de shell del modo interactivo para **restore**

Si necesita restaurar un sistema de archivos completo, utilice **restore** con la opción **-r**. Se restauran los sistemas de archivos a cualquier partición de disco duro en blanco con formato del tamaño adecuado, incluida la partición original del sistema de archivos. Puede ser recomendable, si es posible, restaurar el sistema de archivos en otra partición y ver los resultados.

Para restaurar un sistema de archivos completo es necesario configurar una partición formateada, montarla en su sistema y después cambiar al directorio superior para ejecutar el comando **restore**. Primero debe utilizar **mkfs** para dar formato a la partición donde está almacenando el sistema de archivos, y después montarlo en su sistema. Después se utiliza **restore** con las opciones **-r** y **-f** para especificar el dispositivo que almacena la copia de seguridad del sistema de archivos. En el siguiente ejemplo, el usuario da formato y monta la partición **/dev/hda5** y después restaura la partición copiada por el sistema de archivos, actualmente una cinta en el dispositivo **/dev/tape**.

```
mkfs /dev/hda5
mount /dev/hda5 /mystuff
cd /mystuff
restore -rf /dev/tape
```

Para restaurar desde un dispositivo de copia de seguridad ubicado en otro sistema de su red, tiene que especificar el nombre de host del sistema y el nombre de su dispositivo. El nombre de host se inserta antes del nombre de dispositivo y se delimita con dos puntos. En el siguiente ejemplo, el usuario restaura el sistema de archivos de copia de seguridad en el dispositivo de cinta con el nombre **/dev/tape** en el sistema **conejo.mipista.com**:

```
restore -rf conejo.mipista.com:/dev/tape
```


VIII

PARTE

Servicios de administración de red

CAPÍTULO 34

Administración de redes
TCP/IP

CAPÍTULO 35

Configuración automática de
red con IPv6, DHCPv6
y DHCP

CAPÍTULO 36

NFS y NIS

CAPÍTULO 37

Sistemas de archivos de red
distribuidos

Administración de redes TCP/IP

Los sistemas Linux se configuran para conectarse en redes que utilizan protocolos TCP/IP. Son los mismos protocolos que utiliza Internet, al igual que muchas redes de área local (LAN, Local Area Network). TCP/IP es un conjunto robusto de protocolos diseñado para proporcionar comunicaciones entre sistemas con diferente hardware y sistemas operativos. Los protocolos TCP/IP fueron desarrollados en la década de 1970 como un proyecto especial denominado DARPA para mejorar las comunicaciones entre universidades y centros de investigación. Estos protocolos se desarrollaron originalmente en sistemas Unix, y gran parte de la investigación se llevó a cabo en la Universidad de California en Berkeley. Linux, como una versión de Unix, se beneficia de gran parte del método original en Unix. Actualmente, el desarrollo de protocolo TCP/IP lo administra la Internet Engineering Task Force (IETF), que, a su vez, es supervisada por la Internet Society (ISOC). ISOC supervisa varios grupos responsables de diferentes áreas de desarrollo de Internet, como la Internet Assigned Numbers Authority (IANA), que es responsable de las direcciones de Internet (consulte la tabla 34-1). A través de los años, los protocolos estándar y la documentación de TCP/IP se han publicado en forma de documentos de solicitud de comentarios (RFC, Request for Comments). Revise los más recientes para conocer desarrollos actuales en el sitio Web de IETF en ietf.org.

Suite de protocolo TCP/IP

El conjunto de protocolos TCP/IP realmente está integrado por protocolos diferentes, cada uno diseñado para una tarea específica en una red TCP/IP. Los tres protocolos básicos son protocolo de control de transmisión (TCP, Transmission Control Protocol), que maneja la recepción y el envío de comunicaciones; protocolo de Internet (IP, Internet Protocol), que administra las transacciones reales; y protocolo de datagrama de usuario (UDP, User Datagram Protocol), que también administra la recepción y el envío de paquetes. El protocolo IP, que es la base que todos los demás utilizan, administra las transmisiones reales, los paquetes de datos con información de emisor y receptor en cada uno. El protocolo TCP está diseñado para trabajar con mensajes o datos cohesivos. Este protocolo revisa los paquetes recibidos y los ordena de la manera que se determine, formando

Grupo	Título	Descripción
ISOC	Internet Society	Internet Society Organización de membresía profesional de expertos de Internet que supervisa comités y grupos operativos que tratan con problemas de directivas de red isoc.org
IESG	The Internet Engineering Steering Group	Responsable de la administración técnica de actividades de IETF y el proceso estándar de Internet ietf.org/iesg.html
IANA	Internet Assigned Numbers Authority	Responsable de las direcciones de protocolo de Internet (IP) iana.org
IAB	Internet Architecture Board	Define la arquitectura general de Internet, que proporciona instrucciones y una dirección extensa a IETF iab.org
IETF	Internet Engineering Task Force	Rama de ingeniería y desarrollo de protocolos de Internet ietf.org

TABLA 34-1 Grupos de desarrollo de protocoloTCP/IP

el mensaje original. En el caso de datos enviados, el protocolo TCP los divide en paquetes separados, designándoles un orden. El protocolo UDP, que funciona en un nivel mucho más básico, también divide datos en paquetes pero no revisa su orden. El protocolo TCP/IP está diseñado para proporcionar conexiones estables y confiables que aseguran que todos los datos se reciben y reorganizan en su orden original. UDP, por otro lado, está diseñado para enviar simplemente la mayor cantidad de datos posibles, sin garantizar que los paquetes se recibirán o colocarán en el orden apropiado. UDP a menudo se utiliza para transmitir grandes cantidades de datos del tipo que sobrevive a la pérdida de unos cuantos paquetes (por ejemplo, imágenes temporales, videos y anuncios desplegados en Internet).

Otros protocolos proporcionan varios servicios de red y usuario. El servicio de nombres de dominio (DNS, Domain Name Service) proporciona resolución de direcciones. El protocolo de transferencia de archivos (FTP, File Transfer Protocol) proporciona transmisión de archivos, y sistema de archivos de red (NFS, Network File System) proporciona acceso a sistemas de archivos remotos. En la tabla 34-2 se presenta una lista de los diferentes protocolos en la suite TCP/IP. Estos protocolos usan TCP o UDP para enviar y recibir paquetes, que, a su vez, utilizan el protocolo IP para transmitir realmente los paquetes.

En una red TCP/IP, los mensajes se dividen en pequeños componentes, llamados *datagramas*, que después se transmiten a través de varias rutas entrelazadas y entregan el paquete a sus equipos de destino. Una vez recibidos, los datagramas se reensamblan para formar el mensaje original. Los datagramas por sí solos se dividen en pequeños paquetes. El *paquete* es la unidad de mensaje físico que se transmite realmente entre redes. Se ha probado que enviar mensajes como pequeños componentes es mucho más confiable y rápido que enviarlos como una sola transmisión grande. Con componentes pequeños, si uno se pierde o se daña, sólo se tiene que enviar ese componente, mientras que si una parte de la transmisión grande se corrompe o pierde, todo el mensaje tiene que enviarse nuevamente.

La configuración de una red TCP/IP en su sistema Linux se implementa al utilizar un conjunto de archivos de configuración de red (en la tabla 34-6, en páginas posteriores de este capítulo, se proporciona una lista). Muchos de éstos se administran al utilizar programa administrativos

proporcionados por su distribución en su escritorio de usuario raíz. También se utilizan programas más especializados, como netstat, ifconfig, Wireshark y route. Algunos archivos de configuración se modifican de manera mucho más sencilla con el uso de un editor de texto.

Transporte	Descripción
TCP	Protocolo de control de transmisión; coloca sistemas en comunicación directa
UDP	Protocolo de datagrama de usuario
IP	IProtocolo de Internet; transmite datos
ICMP	Protocolo de control de mensajes de Internet; mensajes de estado para IP
Enrutamiento	Descripción
RIP	Protocolo de información de enrutamiento; determina el enrutamiento
OSPF	Abrir primero la ruta más corta; determina el enrutamiento
Dirección de red	Descripción
ARP	Protocolo de resolución de direcciones; determina direcciones IP únicas de sistema
DNS	Servicio de nombres de dominio; traduce nombres de host en direcciones IP
RARP	Protocolo de resolución de dirección inversa; determina direcciones de sistemas
Servicio de usuario	Descripción
FTP	Protocolo de transferencia de archivos; transmite archivos de un sistema a otro utilizando TCP
TFTP	Protocolo trivial de transferencia de archivos; transfiere archivos utilizando UDP
Telnet	Inicio de sesión remoto a otros sistemas en la red
SMTP	Protocolo simple de transferencia de correo; transfiere correos electrónicos entre sistemas
RPC	Llamada a procedimiento remoto; permite que programas en sistemas remotos se comuniquen
Puerta de enlace	Descripción
EGP	Protocolo de puerta de enlace exterior; proporciona enrutamiento para redes externas
GGP	Protocolo de puerta de enlace a puerta de enlace; proporciona enrutamiento entre puertas de enlace de Internet
IGP	Protocolo de puerta de enlace interior; proporciona enrutamiento para redes internas
Servicio de red	Descripción
NFS	Sistema de archivos de red; permite el montaje de sistemas de archivos en máquinas remotas
NIS	Servicio de información de red; mantiene cuentas de usuario a través de una red
BOOTP	Protocolo de arranque; inicia el sistema al utilizar información boot en el servidor para la red
SNMP	Protocolo de administración simple de red; proporciona mensajes de estado en configuración TCP/IP
DHCP	Protocolo de configuración dinámica de host; proporciona automáticamente información de configuración de red para sistemas host

TABLA 34-2 Conjunto de protocolos TCP/IP

TCP/IP se configura y administra con un conjunto de utilerías: `ifconfig`, `route` y `netstat`. La utilería `ifconfig` opera desde su escritorio de usuario raíz y le permite configurar sus interfaces de red en su totalidad, al agregar nuevas o modificar otras. Las utilerías `ifconfig` y `route` son programas de nivel bajo que requieren conocimientos más específicos de su red para utilizarse de forma eficaz. La utilería `netstat` le proporciona información acerca del estado de sus conexiones de red. Wireshark es un analizador de protocolo de red que le permite capturar paquetes a medida que se transmiten a través de su red, seleccionando los que quiere revisar.

Configuración de redes en GNOME y KDE

GNOME y KDE proporcionan herramientas para configurar sus conexiones de red de manera sencilla. En GNOME, la herramienta **network-admin** despliega una ventana Configuración de red, con paneles para Conexiones, General, Anfitriones y DNS. La entrada Conexión presenta una lista de las posibles conexiones como Inalámbrica, Cableada y Modem. Las conexiones activas tendrán marcas de verificación en sus casillas. Se desactiva una conexión al quitar la marca de la casilla de verificación. Para configurar una conexión, selecciónela y haga clic en Propiedades. Esto abre una ventana Configuraciones para su interfaz conectada. Aquí se selecciona el tipo de configuración que quiere, como DHCP, IP estática o zeroconf. Si selecciona IP estática se inserta la dirección IP, la máscara de red y la dirección de puerta de enlace.

El panel General le permite establecer el nombre de host y de dominio. En el panel DNS se especifican servidores DNS y dominios de búsqueda (en una configuración DHCP se especificarán de manera automática). El panel Anfitriones muestra todos los nombres de host especificados manualmente en su red. Se agregan o eliminan entradas.

Si ha hecho cambios, se guardan al hacer clic en el botón de disco en la ventana principal. Se le pedirá que especifique una ubicación o agregue una nueva (la actual se mostrará como opción predeterminada). Se establecen diferentes configuraciones para varias ubicaciones como oficina o casa. Después se carga la ubicación que quiere del menú desplegable Ubicaciones.

En KDE, se selecciona Parámetros de red, en el menú Internet y red, del Centro de control. Haga clic en el botón Modo administrador, para que se le permita cambiar entradas. Se despliegan cuatro paneles para Interfaces de red, Rutas, Sistema de nombres de dominio y Perfiles de red. En el panel Interfaces de red, se muestran sus interfaces de red conectadas. Existen botones para Habilitar o Deshabilitar una interfaz. Para configurar un dispositivo, haga clic en su entrada y luego en el botón Configurar interfaz. Esto abre una ventana donde se seleccionan configuraciones Automática o Manual. En el primer caso se utiliza DHCP (la opción predeterminada). La configuración manual le permite insertar una dirección IP y máscara de red. En el panel Rutas se inserta la dirección de Puerta de enlace (en configuraciones DHCP, ya estará insertada). El panel de sistema de nombres de dominio le permite especificar el nombre de host de su equipo, su nombre de dominio y cualquier servidor de nombres de dominio en su red. También se configuran hosts estáticos en su red. El panel Perfiles de red le permite guardar o crear un perfil de red, como uno para la casa u oficina.

Zero Configuration Networking (zeroconf): Avahi y direccionamiento de vínculo local

Zero Configuration Networking (zeroconf) permite la configuración de redes privadas que no se enrutan sin la necesidad de un servidor DHCP o una dirección IP estática. Una configuración zeroconf permite a los usuarios conectarse a una red automáticamente y acceder a todos los recursos de red como impresoras, sin tener que realizar ninguna configuración. En Linux, las redes zeroconf se implementan con Avahi (avahi.org). Avahi incluye soportes DNS de multidifusión (mDNS,

Multicast DNS) y descubrimiento de servicios DNS (DNS-SD, DNS Service Discovery) que detecta automáticamente servicios en una red. Las direcciones IP se determinan al utilizar direccionamiento IPv6 o IPv4 Link Local (IPv4LL). Las direcciones IPv4 Link Local se asignan del almacén de red 168.254.0.0. Derivado de la implementación de zeroconf Bonjour de Apple, existe una versión gratuita y de fuente abierta utilizada actualmente por herramientas de escritorio como el sistema de archivos virtual de GNOME. Algunas distribuciones implementan soporte de red completo a zeroconf (Debian y Ubuntu), mientras otras hacen uso del descubrimiento de servicio DNS (Red Hat y Fedora). Muchas distribuciones incluyen la solución zeroconf de KDE mediante Avahi (kdnssd). Ésta se ubica en los paquetes kdnssd-avahi. Utilice el panel Buscador de servicios, del Centro de control de KDE (Internet y red) para especificar su dominio. Inserte **zeroconf:/** en una ventana de administrador de archivos de KDE.

IPv4 e IPv6

Tradicionalmente, una dirección TCP/IP se organiza en cuatro segmentos, que constan de números separados por puntos. A esto se le llama *dirección IP*, que realmente representa un número entero de 32-bits cuyos valores binarios identifican la red y el host. Esta forma de direccionamiento IP se adhiere al protocolo de Internet, versión 4, también conocida como IPv4. Éste, que es el tipo de direccionamiento IP descrito aquí, todavía se utiliza ampliamente.

En la actualidad, una nueva versión de protocolo IP llamada protocolo de Internet, versión 6 (IPv6), está reemplazando gradualmente la versión IPv4 más antigua. IPv6 expande el número posible de direcciones IP al utilizar 128 bits. Es totalmente compatible con sistemas que todavía utilizan IPv4. Las direcciones IPv6 se representan de forma diferente, al utilizar un conjunto de ocho segmentos de 16-bits, cada uno separado del siguiente por dos puntos. Cada segmento se representa por números hexadecimales. Un ejemplo de dirección es

FEC0:0:0:0:800:BA98:7654:3210

Entre las ventajas de IPv6 se incluyen las siguientes:

- Presenta encabezados simplificados que permiten un procesamiento más rápido.
- Proporciona soporte a cifrado y autenticación, junto con redes privadas virtuales (VPN, Virtual Private Network) al utilizar el protocolo IPsec integrado.
- Una de las ventajas más significativas recae en la extensión del espacio de direcciones para cubrir 2 a la 128 potencia de hosts posibles (miles de miles de millones). Esto rebasa en mucho a los 4 200 millones soportados por IPv4.
- Soporta configuración automática y sin estado de direcciones para host, al evitar la necesidad de que DHCP configure estas direcciones. Una dirección se genera directamente al utilizar la dirección de hardware de una interfaz MAC (Media Access Control, control de acceso a medios).
- Es compatible con operaciones de calidad de servicio (QoS, Quality of Service), que proporcionan tiempos de respuesta suficientes para servicios como tareas multimedia y telecom.
- Las capacidades de multidifusión están integradas en el protocolo, que proporciona soporte directo a tareas multimedia. El direccionamiento de multidifusión también proporciona la misma función que el direccionamiento de transmisión de IPv4.

- Transmisiones más robustas se aseguran con direccionamiento anycast, donde los paquetes se dirigen a un grupo de sistemas anycast; sólo es necesario que uno de ellos los reciba. Varios servidores DNS que dan soporte a una red determinada se designan como un grupo anycast; de ellos sólo un servidor DNS necesita recibir la transmisión, al proporcionar una mayor probabilidad de que la transmisión tenga éxito.
- Provee un mejor acceso para nodos móviles como PDA, equipos portátiles y teléfonos celulares.

Direcciones de red TCP/IP

Como se observó antes, la dirección TCP/IP IPv4 tradicional se organiza en cuatro segmentos, que son números separados por puntos. Este tipo de direcciones todavía es de uso común y es lo que la gente conoce como *dirección IP*. Parte de una dirección IP se utiliza para la dirección de red, y la otra parte se utiliza para identificar una interfaz particular en un host de la red. Debe observar que las direcciones IP se asignan a interfaces (como tarjetas Ethernet o módems) y no al equipo host. Por lo general, un equipo tiene sólo una interfaz y se accede a él al utilizar sólo la dirección IP de la interfaz. En ese aspecto, una dirección IP se considera cómo identificar un sistema de host particular en una red y por eso a la dirección IP suele conocerse como *dirección de host*.

Sin embargo, en realidad un sistema host tiene varias interfaces, cada una con su propia dirección IP. Así sucede con los equipos que operan como puertas de enlace y firewalls de la red local a Internet. Una interfaz suele conectarse a la LAN y otra a Internet, como en el caso de dos tarjetas Ethernet. Cada interfaz (como una tarjeta Ethernet) tiene su propia dirección IP. Si utiliza un módem para conectarse a un ISP, debe configurar una interfaz PPP que también tiene su propia dirección IP (por lo general, el ISP la asigna de forma dinámica). Es importante recordar esta distinción si planea utilizar Linux para configurar una red local o casera, al utilizar Linux como su máquina de puerta de enlace a Internet (consulte la sección “Enmascaramiento de IP” en el capítulo 20).

Direcciones de red IPv4

La dirección IP se divide en dos partes: una identifica la red, y la otra, un host particular. La dirección de red identifica la red a la que pertenece una interfaz particular en un host. Existen dos métodos para implementar las partes de red y host de una dirección IP: la dirección IP original basada en clase y el direccionamiento actual de enrutamiento entre dominios y sin clase (CIDR, Classless Interdomain Routing). El direccionamiento IP basado en clase designa de forma oficial partes predeterminadas de direcciones para las direcciones de red y host, mientras que el direccionamiento CIDR permite que partes se determinen de forma dinámica al utilizar una máscara de red.

Direccionamiento IP basado en clase

Originalmente, las direcciones IP se organizaban de acuerdo con clases. En Internet, las redes se organizan en tres clases, dependiendo de su tamaño (clases A, B y C). Una clase A de red sólo utiliza el primer segmento de una dirección de red y los tres restantes para el host, al permitir que una gran cantidad de computadores se conecten a la misma red. Casi todas las direcciones IP hacen referencia a redes clase C, más pequeñas. En el caso de una red de clase C, los primeros tres segmentos se utilizan para identificar la red, y sólo el último segmento identifica al host. En conjunto, esto forma una dirección única con la cual se identifica cualquier interfaz de red en equipos en una red TCP/IP. Por ejemplo, en la dirección IP 192.168.1.72, la red es parte de 192.168.1 y la parte interfaz/host es 72. La interfaz/host es parte de una red cuya dirección propia es 192.168.1.0.

En una clase de red C, los primeros tres números identifican la parte de red de la dirección IP. Esta parte se divide en tres números; cada uno identifica una subred. Las redes en Internet se organizan en subredes, que comienzan con la mayor y llegan a las subredes más pequeñas. El último número se utiliza para identificar un equipo particular, conocido como *host*. Internet se considera como una serie de redes con subredes; estas subredes tienen sus propias subredes. El que se encuentra en el extremo derecho identifica el equipo host, y el número anterior identifica la subred a la que pertenece. El número de la izquierda identifica la red a la que pertenece la subred, etc. La dirección de Internet 192.168.187.4 hace referencia al cuarto equipo conectado en la red identificado por el número 187. La red 187 es una subred de una red más grande identificada como 168. Esta red más grande es una subred de la red, identificada como 192. Aquí se muestra cómo se divide:

192.168.187.4	Dirección IPv4
192.168.187	Identificación de red
4	Identificación de host

Máscara de red

Los sistemas derivan las direcciones de red de la dirección de host mediante la máscara de subred. Una dirección IP se considera como una serie de 32 bits binarios; algunos se utilizan para la red y el resto para el host. La *máscara de subred* tiene el conjunto de bits de red establecido con 1, y casi todos los bits de host tienen asignados 0 (véase la figura 34-1). En la máscara de red de una dirección IP basada en clase estándar, todos los números en la parte de la red de una dirección host están establecidos en 255, y la parte de host en 0. Esto tiene el efecto de establecer en 1 todos los bits binarios que integran la dirección de red en las máscaras de red. Así, ésta es su máscara de red. De modo que la máscara de red para la dirección de host clase C 192.168.1.72 es 255.255.255.0. La parte de la red, 192.168.1, se define como 255.255.255, y la parte de host, 72, como 0. Entonces, los sistemas utilizan su máscara de subred para derivar su dirección de red a partir de su dirección de host. Pueden determinar qué parte de su dirección de host integra su dirección de red y qué números son.

En el caso de quienes están familiarizados con programación de cómputo, una operación simple AND en el nivel de bits en la máscara de subred y la dirección de red da como resultado que la parte del host sea 0, dejándole con la parte de red de la dirección de host. Puede considerar que una dirección se implementa como un número entero de cuatro bytes, y cada byte corresponde a un segmento de red. En una dirección clase C, los tres segmentos de red corresponden a los primeros tres bytes y el segmento huésped corresponde al cuarto byte. Una máscara de red es diseñada para enmascarar la parte de host de la dirección, dejando los segmentos de red solos. En la máscara de red para una clase C de red estándar, los primeros tres bytes son 1 y el último byte consta de 0. Los 0 del último byte enmascaran la parte de host de la dirección, y los primeros tres bytes 1 dejan sola la parte de red de la dirección. En la figura 34-1 se muestra una operación que trabaja en el nivel de bits de la máscara de red en la dirección 192.168.1.4. Cuando se aplica a esta dirección, se conserva la dirección de red (192.168.1) y se enmascara la dirección de host (4), al darle 192.168.1.0 como dirección de red.

La máscara de red, tal como se utiliza en el enrutamiento entre dominios y sin clase (CIDR), es mucho más flexible. En lugar de que la clase de red determine el tamaño de la dirección de red y su máscara, lo determina un número adjunto al final de la dirección IP. Este número simplemente especifica el tamaño de la dirección de red, cuántos bits ocupa en la dirección. Por ejemplo, en una dirección IP cuya parte de la red ocupa los primeros tres bytes (segmentos), se ocupan 24 bytes: ocho bits por cada byte (segmento). En lugar de utilizar una máscara de red para determinar la

Direccionamiento basado en clase				
Dirección IP	192.168.1.4			
	Red			Host
binario	11000000	10101000	00000001	00000100
numérico	192	168	1	4
Máscara de red	255.255.255.0			
binario	11111111	11111111	11111111	00000000
numérico	255	255	255	000
Dirección de red	192.168.1.0			
binario	11000000	10101000	00000001	00000000
numérico	192	168	1	0
Operación de máscara de red				
Dirección IP	11000000	10101000	00000001	00000100
Máscara de red	11111111	11111111	11111111	00000000
Dirección de red	11000000	10101000	00000001	00000000

FIGURA 34-1 Operaciones de máscara de red basadas en clase

dirección de red, el número del tamaño de red se adjunta al final de la dirección con una diagonal, como se muestra aquí:

192.168.1.72/24

CIDR le da la ventaja de especificar redes con cualquier tamaño en bits, en lugar de sólo tres segmentos posibles. Puede tener una red cuyas direcciones toman 14, 22 o incluso 25 bits. La dirección de host utiliza los bits sobrantes. Una dirección IP con 21 bits para la red cubre direcciones de host que utilizan los 11 bits restantes, de 0 a 2047.

Enrutamiento entre dominios y sin clase (CIDR)

Actualmente, la organización basada en clase de las direcciones IP se está remplazando con el formato CIDR. Éste se diseñó para redes de tamaño mediano, como las que se encuentran entre una clase C y las clases con números de host mayores que 256 y menores que 65,534. Una dirección IP basada en red de clase C sólo utiliza un segmento de host, un número entero de 8 bits con un valor máximo de 256, para hosts. Una dirección IP basada en red de clase B utiliza dos segmentos, que hacen números enteros de 16 bits cuyo máximo valor es 65 534. Se considera que una dirección es como un número entero de 32 bits que ocupa cuatro bytes (cada byte son 8 bits). Cada segmento integra uno de los cuatro bytes. Una red de clase C utiliza tres segmentos, o 24 bits, para integrar su dirección de red. Una red de clase B, en cambio, utiliza dos segmentos, o 16 bits, para su dirección. Con el esquema, los hosts y las direcciones de red permitidas se cambian de byte entero en byte, segmento por segmento. Con direccionamiento CIDR, se define un host y una dirección de red por bits, en lugar de todo un segmento. Por ejemplo, se utiliza un direccionamiento CIDR para expandir el segmento de host de 8 bits a 9, en lugar de tener que saltar a una clase B de 16 bits (dos segmentos).

La notación de direccionamiento CIDR logra esto al incorporar información de máscara de red en la dirección IP (la máscara de red se aplica a una dirección IP para determinar la parte de red de la dirección). En una notación CIDR, el número de bits que integran la dirección de red se coloca

FIGURA 34-2
Direccionamiento CIDR

Direccionamiento CIDR				
Dirección IP 192.168.4.6/22				
	Red		Host	
binario	111000000	10101000	0000011	00 00000110
numérico	192	168	4	6
Máscara de red	255.255.252.0		22 bits	
binario	11111111	11111111	111111	00 00000000
numérico	255	255	252	000

después de la dirección IP, tras una diagonal. Por ejemplo, la forma CIDR de la dirección IP 192.168.187.4 clase C es

192.168.187.4/24

En la figura 34-2 se muestra un ejemplo de dirección CIDR y su máscara de red. La dirección IP es 192.168.1.6 con una máscara de red de 22 bits, 192.168.1.6/22. La dirección de red toma los primeros 22 bits de la dirección IP, y los 10 bits restantes se utilizan para la dirección de host. La dirección de host toma el equivalente al cuarto segmento de una dirección IP basada en clase (8 bits) y 2 bits del tercer segmento.

En la tabla 34-3 se presenta una lista de las diferentes máscaras de red CIDR IPv4 disponibles junto con el número máximo de hosts. Se incluyen las formas cortas y las completas de la máscara de red.

Direccionamiento CIDR IPv4

La dirección de red para cualquier dirección IPv4 clase C estándar ocupa los primeros tres segmentos, 24 bits. Si quiere crear una red con un máximo de 512 hosts, se les dan direcciones IP donde la dirección de red es de 23 bits y la de host ocupa 9 bits (0-511). Sin embargo, la notación de

Forma corta	Forma completa	Número máximo de hosts
/8	/255.0.0.0	16 777,215 (Clase A)
/16	/255.255.0.0	65 534 (Clase B)
/17	/255.255.128.0	32 767
/18	/255.255.192.0	16 383
/19	/255.255.224.0	8 191
/20	/255.255.240.0	4 095
/21	/255.255.248.0	2 047
/22	/255.255.252.0	1 023
/23	/255.255.254.0	511
/24	/255.255.255.0	255 (Clase C)
/25	/255.255.255.128	127
/26	/255.255.255.192	63
/27	/255.255.255.224	31
/28	/255.255.255.240	15
/29	/255.255.255.248	7
/30	/255.255.255.252	3

TABLA 34-3 Máscaras de red IPv4 de CIDR

dirección IP permanece igual al utilizar los cuatro segmentos de 8 bits. Esto significa que un número del segmento determinado se puede utilizar para una dirección de red y una de host. Los segmentos ya no son parte integral de la dirección de host o de red. Asignar una dirección de red de 23 bits y una dirección host de 9 bits significa que el número que se encuentra en el tercer segmento es parte de las direcciones de red y de host, los primeros 7 bits para la red y el último bit para el host. En el siguiente ejemplo, el tercer número, 145, se utiliza como final de la dirección de red y como el principio de la dirección de host:

192.168.145.67/23

Esta situación complica el direccionamiento CIDR, y en algunos casos la única forma de representar la dirección es especificar dos o más direcciones de red. Revise el RFC 1520, en ietf.org, para conocer más detalles.

NOTA Una forma simple de calcular el número de hosts que se puede direccionar en una red consiste en tomar el número de bits en el segmento de host como un potencia de 2, y después restar 2 (es decir, 2 al número de bits de host, menos 2). Por ejemplo, un segmento de host de 8 bits podría ser 2^8 , lo que es igual a 256. Se restan 2 (1 para la dirección de transmisión, 255, y 1 para el valor cero, 000) para dejarle con 254 hosts posibles.

CIDR también permite que un administrador de red tome lo que es oficialmente la parte del host de una dirección IP y la divida en subredes con menos hosts. A esto se le conoce como *división en subredes*. Una red tendrá su dirección de red IP oficial reconocida en Internet o por una red más grande. El administrador de red tiene la opción de crear, a cambio, varias redes más pequeñas dentro de ésta al utilizar enmascaramiento de red CIDR. Un ejemplo clásico consiste en tomar una red de clase C estándar con 254 hosts y dividirla en dos redes más pequeñas, cada una con 64 hosts. Esto se hace al utilizar una máscara de red CIDR para tomar un bit de la parte de host de la dirección IP y utilizarla para subredes. Los números que se encuentran dentro del rango de las 254 direcciones originales cuyo primer bit se establecería a 1 representarían a una subred; las otras, cuyo primer bit se establecería en 0, constituirán la red restante. En la red cuya dirección de red es 192.168.187.0, donde los últimos dos segmentos se utilizan para los nombres de host, los últimos dos segmentos pueden dividirse en dos subredes, cada una con sus propios hosts. Para dos subredes, se utilizaría el primer bit en el último segmento de 8 bits para la red. Los 7 bits restantes se utilizan entonces para las direcciones de hosts, al darle un rango de 127 hosts por red. La subred cuyo bit se establece en 0 tendría un rango de 1 a 127, con una máscara de red CIDR de 25. El segmento de 8 bits para el primer host sería 00000001. Así que el host con la dirección de 1 en la red tendría esta dirección IP:

192.168.187.1/25

Para la subred donde el primer bit es 1, el primer host tendría una dirección de 129, con la máscara de red CIDR 25, como se muestra aquí. La secuencia de 8 bits del primer host sería 10000001.

192.168.187.129/25

Cada subred tendría un conjunto de 126 direcciones, la primera de 1 a 126, y la segunda de 129 a 254; 127 es la dirección de transmisión para la primera subred, y 128 es la dirección de red

para la segunda subred. Aquí se muestran las subredes posibles y las máscaras que pueden utilizarse:

Subred	Dirección CIDR	Máscara binaria
Primera dirección de red de la subred	.0/25	00000000
Segunda dirección de red de la subred	.128/25	10000000
Primera dirección de transmisión de la subred	.127/25	01111111
Segunda dirección de transmisión de la subred	.255/25	11111111
Primera dirección en la primera subred	.1/25	00000001
Primera dirección en la segunda subred	.129/25	10000001
Última dirección de red en la primera subred	.126/25	01111110
Última dirección de red en la segunda subred	.254/25	11111110

Direccionamiento CIDR IPv6

El direccionamiento CIDR IPv6 trabaja casi de la misma forma que con el método IPv4. El número de bits utilizados para la información de red se indica con un número seguido de la dirección. Una dirección de host (interfaz) puede ocupar mucho más que los 64 bits de lo que suele hacer en una dirección IPv6, al hacer la sección del prefijo de red (dirección) menor de 64 bits. El siguiente número indica cuántos bits utiliza ese prefijo de red. En el siguiente ejemplo el prefijo de red (dirección) sólo utiliza los primeros 48 bits de la dirección IPv6, y la dirección de host utiliza los 80 bits restantes:

FEC0:0000:0000:0000:FEDC:BA98:7654:3210/48

También se utiliza una notación de dos puntos consecutivos (::) para la versión comprimida:

FEC0::FEDC:BA98:7654:3210/48

Aunque puede utilizar CIDR para direcciones de subred, IPv6 también soporta un campo de subred que se utiliza para subredes.

Obtención de una dirección IP

IANA asigna oficialmente las direcciones IP y administra todos los aspectos del direccionamiento de Internet (iana.org). IANA supervisa a Internet Registries (IR), que, a su vez, mantiene direcciones de Internet en niveles regionales o locales. Internet Registry for the Americas es American Registry for Internet Numbers (ARIN), cuyo sitio Web es arin.net. Los proveedores de servicios de Internet (ISP, Internet Service Providers) proporcionan estas direcciones a los usuarios. Se obtiene su propia dirección de Internet de un ISP, o si está en una red ya conectada a Internet, su administrador de red le asignará una. Si está utilizando un ISP, éste puede asignar una temporalmente, tomada de un conjunto que tiene a la mano con cada uso.

Direcciones IPv4 reservadas

Ciertos números están reservados. Los números 127, 0 o 255 no son parte de una dirección de red IP oficial. El número 127 se utiliza para designar la dirección de red para la interfaz de retroalimentación de su sistema. Esta interfaz permite a los usuarios de su sistema comunicarse con

otros dentro del sistema sin tener que enrutar a través de una conexión de red. Su dirección de red sería 127.0.0.0, y su dirección IP es 127.0.0.1. Para un direccionamiento IP basado en clases, el número 255 es un identificador de transmisión especial que se utiliza para difundir mensajes a todos los sitios en una red. Al utilizar 255 para cualquier parte de la dirección IP, se hace referencia a todos los nodos conectados en ese nivel. Por ejemplo, 192.168.255.255 difunde un mensaje a todos los equipos en la red 192.168, todas sus subredes y sus hosts. La dirección 192.168.187.255 difunde a cada equipo de una red local. Si se utiliza 0 para la parte de red de la dirección, el número de host hace referencia al equipo dentro de su red local. Por ejemplo, 0.0.0.6 hace referencia a la sexta computadora de su red local. Si quiere transmitir a todos sus equipos de su red local, se utiliza el número 0.0.0.255. Para direccionamiento IP CIDR, la dirección de transmisión puede parecerse mucho a una dirección IP normal. Como se indica en la sección anterior, el direccionamiento CIDR permite el uso de cualquier número de bits para integrar la dirección IP para la parte de la red o del host. Para una dirección de transmisión, la parte de host debe tener todos sus bits en 1.

Un conjunto especial de números está reservado para uso en LAN sin Internet (RFC 1918). Éstos son los números que comienzan con el número de red especial 192.168 (para redes de clase C), como se utilizan en estos ejemplos. Si está configurando una LAN, como una red casera o de un negocio pequeño, es libre de utilizar estos números para sus máquinas locales. Se configura una intranet al utilizar tarjetas de red, como tarjetas y concentradores Ethernet, y después se configuran sus máquinas con las direcciones IP que empiezan en 192.168.1.1. El segmento de host llega hasta 256. Si tiene tres máquinas en su red casera, puede darles las direcciones 192.168.1.1, 192.168.1.2 y 192.168.1.3. Se implementan servicios de Internet, como FTP, Web y servicios de correo, en sus máquinas locales y se utiliza cualquier herramienta de Internet para usar esos servicios. Todos utilizan los mismos protocolos TCP/IP que se usan en Internet. Por ejemplo, con herramientas FTP, se transfieren archivos entre máquinas en su red. Con herramientas de correo, se envían mensajes de una máquina a otra y con un explorador Web se accede a sitios Web locales que estén instalados en una máquina que ejecuta sus propios servidores Web. Si quiere tener una de sus máquinas conectadas a Internet o alguna otra red, se configura para que sea una máquina de puerta de enlace. Por convención, a esta máquina suele dársele la dirección 192.168.1.1. Con un método llamado *enmascaramiento IP*, se hace que cualquiera de las máquinas sin Internet utilice una puerta de enlace para conectarse a Internet.

Los números también se reservan para redes locales sin Internet de clase A y B. En la tabla 34-4 se presenta una lista de estas direcciones. Las direcciones disponibles posibles abarcan de 0 a 255 en el segmento de host de la dirección. Por ejemplo, las direcciones de red de clase B van de 172.16.0.0 a 172.31.255.255, para darle un total de 32,356 hosts posibles. La red de clase C va de 192.168.0.0 a 192.168.255.255, para darle 256 posibles subredes, cada una con 256 posibles hosts. La dirección de red 127.0.0.0 se reserva para la interfaz de retroalimentación del sistema, que permite comunicarse consigo mismo, al habilitar a los usuarios en el mismo sistema para enviarse mensajes entre sí.

Direcciones de red privada IPv4	Clases de red
10.0.0.0	Red clase A
172.16.0.0–172.31.255.255	Red clase B
192.168.0.0	Red clase C
127.0.0.0	Red clase de retroalimentación (para comunicación dentro del propio sistema)

TABLA 34-4 Direcciones IP de red local IPv4

Direcciones de transmisión

La dirección de transmisión permite a un sistema enviar el mismo mensaje a todos los sistemas de su red al mismo tiempo. Con el direccionamiento IP basado en clases IPv4, se determina de forma sencilla la dirección de transmisión al utilizar su dirección de host: la dirección de transmisión tiene la parte de host de su dirección en 255. La parte de red permanece intacta. Así que la dirección de transmisión para la dirección de host 192.168.1.72 es 192.168.1.255 (se combina la parte de red de la dirección con 255 en la parte de host). Para el direccionamiento IP CIDR, necesita conocer el número de bits en la máscara de red. Los bits restantes se establecen en 1 (véase la figura 34-3). Por ejemplo, una dirección IP de 192.168.4.6/22 tiene la dirección de transmisión 192.168.7.255/22. En este caso, los primeros 22 bits son la dirección de red y los últimos 10 son la parte de host establecida en el valor de transmisión (sólo 1).

En realidad, se piensa en una dirección de transmisión de clase C como una dirección CIDR que utiliza 24 bits (los primeros tres segmentos) para la dirección de red y los últimos 8 bits (el cuarto segmento) como dirección de transmisión. El valor 255 expresado en términos binarios está integrado con 8 bits que todos son 1. 255 es igual a 11111111.

Dirección IP	Dirección de transmisión	Número de transmisión IP	Equivalente binario
192.168.1.72	192.168.1.255	255	11111111
192.168.4.6/22	192.168.7.255/22	7.255 (últimos 2 bits en 7)	1111111111

Direcciones de puerta de enlace

Algunas redes tienen un equipo diseñado como la puerta de enlace a otras redes. Cada conexión entre redes pasa a través de este equipo de puerta de enlace. Casi todas las redes locales utilizan puertas de enlace para establecer una conexión a Internet. Si está en este tipo de red, debe proporcionar la dirección de puerta de enlace. Si su red no tiene una conexión a Internet, o una red más grande, tal vez no necesite una dirección de puerta de enlace. Ésta es la dirección del sistema host que proporciona el servicio de puerta de enlace a la red. En muchas redes, a este host se le da un ID de host de 1: la dirección de puerta de enlace para una red con la dirección 192.168.1 sería 192.168.1.1, pero esto es sólo una convención. Para estar seguro de su dirección de puerta de enlace, pregunte a su administrador de red.

Asigne nombre a las direcciones de servidor

Muchas redes, incluido Internet, tienen equipos que proporcionan un servicio de nombre de dominio (DNS, Domain Name Service) que traduce los nombres de dominio de redes y hosts a

FIGURA 34-3
Direccionamiento basado en clase y de transmisión CIDR

Direccionamiento de transmisión basado en clase			
Dirección de transmisión 192.168.1.255			
binario	11000000	10101000	00000111 11111111
numérico	192	168	1 255
Direccionamiento de transmisión CIDR			
Dirección de transmisión 192.168.7.255/22			
	Red		Host
binario	11000000 10101000 00000001		11 11111111
numérico	192 168 1		255

direcciones IP. A éstos se les conoce como *servidores de nombre de dominio* de la red. DNS hace su equipo identificable en una red, al utilizar sólo su nombre de dominio, en vez de su dirección IP.

También se utilizan los nombres de dominio de otros sistemas para hacer referencia a éstos, así que no necesita conocer sus direcciones IP. Sin embargo, debe conocer las direcciones IP de cualquier servidor de nombre de dominio para su red. Se obtienen las direcciones de su administrador de sistema (a menudo existe más de uno). Aunque esté utilizando un ISP, debe conocer la dirección de los servidores de nombre de dominio que opera su ISP para Internet.

Direccionamiento IPv6

Las direcciones IPv6 introducen cambios mayores en el formato y método de direccionar sistemas bajo el protocolo de Internet (consulte RFC 3513 en ietf.org/rfc o faqs.org para conocer más detalles). Hay varios tipos diferentes de direccionamiento con diversos campos para el segmento de red. El segmento de host se ha expandido a una dirección de 64 bits, permitiendo direccionamiento directo para una gran cantidad de sistemas. Cada dirección comienza con un campo de tipo que especifica la clase de dirección, que después determinará cómo se organiza su segmento de red. Estos cambios no sólo están diseñados para expandir el espacio de dirección sino también para proporcionar mayor control sobre transmisiones en el nivel de dirección.

NOTA Casi todas las distribuciones ya tienen el soporte a IPv6 habilitado en el kernel. El soporte de kernel para IPv6 se proporciona con el módulo del kernel IPv6. El soporte a la configuración de kernel se encuentra bajo *Device drivers* | *Networking Support* | *Networking Options* | *The IPv6 Protocol*.

Formato de dirección IPv6

Una dirección IPv6 consta de 128 bits, más que los 32 bits utilizados en las direcciones IPv4. Los primeros 64 bits se utilizan para direccionamiento de red; de ellos, los primeros bits se reservan para indicar el tipo de dirección. Los últimos 64 bits se utilizan para la dirección de interfaz, conocida como el campo identificador de interfaz. La cantidad de bits utilizada para subredes se ajusta con una máscara CIDR, de manera muy parecida al direccionamiento CIDR IPv4 (consulte la sección anterior).

Una dirección IPv6 se escribe como ocho segmentos que representan 16 bits cada uno (128 bits totales). Para representar de forma más sencilla los números binarios de 16 bits, se utilizan números hexadecimales. Éstos usan 16 números únicos, en lugar de los 8 utilizados en la numeración octal. Se integran con los dígitos del 0 al 9, seguidos con los caracteres de la A a la F.

En el siguiente ejemplo, los primeros cuatro segmentos representan la parte de red de la dirección IPv6, y los siguientes cuatro representan la dirección de interfaz (host):

FEC0:0000:0000:0008:0800:200C:417A

Puede cortar cualquier cero inicial, pero no los ceros de la derecha, en cualquier segmento dado. Los segmentos que sólo incluyen ceros se reducen a uno solo.

FEC0:0:0:0:8:800:200C:417A

La dirección de retroalimentación utilizada para el host local se escribe con siete ceros antes del 1.

0:0:0:0:0:0:0:1

Muchas direcciones tendrán secuencias de ceros. IPv6 soporta un símbolo taquigráfico para representar una secuencia de varios ceros en campos adyacentes. Esto consta de dos puntos consecutivos (::). Sólo se utiliza un símbolo :: por dirección.

FEC0::8:800:200C:417A

La dirección de retroalimentación 0000000000000001 se reduce sólo a lo siguiente:

::1

Para facilitar la transición de direccionamiento IPv4 a IPv6, también se da soporte a una forma de direccionamiento que incorpora direcciones IPv4. En este caso, la dirección IPv4 (32 bits) se reduce a representar los dos últimos segmentos de una dirección IPv6 y se escribe al utilizar la notación IPv4.

FEC0::192.168.0.3

Identificadores de interfaz IPv6

La parte del identificador de la dirección IPv6 toma los segundos 64 bits, que constan de cuatro segmentos que contienen cuatro números hexadecimales. La interfaz ID es un identificador único extendido de 64 bits (cuatro segmentos) (EUI-64, Extended Unique Identifier) generado a partir de una dirección de control de acceso a medios (MAC, Media Access Control) del dispositivo de red.

Tipos de dirección IPv6

Existen tres tipos básicos de direcciones IPv6, unidifusión, multidifusión y difusión a cualquier lugar. Éstos, a su vez, tienen sus propios tipos de direcciones.

- Una dirección *unidifusión* se utiliza para un paquete que se envía a un solo destino.
- Una dirección *difusión a cualquier lugar* se utiliza para un paquete que se envía a más de un destino.
- Una dirección *multidifusión* se utiliza para transmitir un paquete a varios destinos.

En IPv6, el direccionamiento se controla con el prefijo de formato que opera como una especie de tipo de dirección. El prefijo de formato es el primer campo de la dirección IP. Los tres tipos de direcciones de red de unidifusión son global, de vínculo local y de sitio local; cada uno se indica con su propio prefijo de formato (consulte la tabla 34-5).

- Las direcciones globales comienzan con el tipo de dirección 3, de sitio local con FEC y de vínculo local con FE8. Las direcciones globales se envían a través de Internet.
- Las direcciones de vínculo local se utilizan para sistemas conectados físicamente a una red local.
- Sitio local se utiliza para cualquier host en una red local. Las direcciones de sitio local operan como direcciones privadas IPv4; se utilizan sólo para acceso local y no para transmitir a través de Internet.

Además, IPv6 tiene dos direcciones especiales reservadas. 0000000000000001 se reserva para la dirección de retroalimentación utilizada para el host local del sistema y 0000000000000000 es la dirección no especificada.

Prefijos de formato y direcciones reservadas de IPv6	Descripción
3	Direcciones globales de unidifusión
FE8	Direcciones de vínculo local de unidifusión; se utilizan para hosts conectados físicamente en una red
FEC	Direcciones de sitio local de unidifusión, comparables a las direcciones privadas IPv4
0000000000000001	Dirección de retroalimentación de unidifusión (para comunicación propia del sistema, host local)
0000000000000000	Dirección sin especificar
FF	Direcciones de multidifusión

TABLA 34-5 Prefijos de formato y direcciones reservadas de IPv6

Direcciones globales de unidifusión de IPv6

Las direcciones globales de unidifusión de IPv6 utilizan cuatro campos: el prefijo de formato, un prefijo de enrutamiento global, el identificador de subred y el identificador de interfaz. El prefijo de formato de una dirección global de unidifusión es 3 (3 bits). El prefijo de enrutamiento global hace referencia a la dirección de red (45 bits) y el ID de subred hace referencia a una subred dentro del sitio (16).

Direcciones de uso local de unidifusión IPv6: direcciones de vínculo local y de sitio local

Para uso local, IPv6 proporciona direcciones de vínculo local y de sitio local. El direccionamiento de vínculo local se utiliza para interfaces (hosts) que se conectan físicamente a una red. Esto suele ser una red local pequeña. Una dirección de vínculo local utiliza sólo tres campos, el prefijo de formato FE8 (10 bits), un campo vacío (54 bits) y el identificador de interfaz (dirección de host, 64 bits). En efecto, la sección de red está vacía.

Las direcciones de sitio local IPv6 tienen tres campos: el prefijo de formato (10 bits), el identificador de subred (54 bits) y el identificador de interfaz (64 bits). Excepto por cualquier subred local, no existen direcciones de red.

Direcciones de multidifusión IPv6

Las direcciones de multidifusión tienen un prefijo de formato de FF (8 bits) con campos de marca y alcance para indicar que el grupo de multidifusión es permanente o temporal y si tiene alcance local o global. Un identificador de grupo (112 bits) hace referencia al grupo de multidifusión. Para el alcance, 2 es de vínculo local, 5 es de sitio local y E es global. Además de sus identificadores de interfaz, los hosts también tendrán un ID de grupo que se utiliza como dirección de transmisión. Se utiliza esta dirección para difundir los hosts. En el siguiente ejemplo se transmitirán sólo los hosts en la red local (5) con el ID de grupo 101:

FF05:0:0:0:0:0:101

Para difundir todos los hosts en un alcance de vínculo local, se utilizaría la dirección de transmisión:

FF02:0:0:0:0:0:0:1

Para un alcance de sitio local, una red local, se utilizaría

FF05:0:0:0:0:0:2

Métodos de coexistencia entre IPv6 e IPv4

En la transición de IPv4 a IPv6, muchas redes encontrarán lo necesario para dar soporte a ambas. Algunas se conectarán a redes que utilizan el protocolo contrario, y otras tendrán que conectarse a través de otras conexiones de red que utilizan ese protocolo. Existen varios métodos IETF oficiales para proporcionar cooperación entre IPv6 e IPv4, que caen en tres categorías principales:

- **Pila dual** Permite que IPv4 e IPv6 coexistan en las mismas redes.
- **Traducción** Permite que los dispositivos IPv6 se comuniquen con dispositivos IPv4.
- **Entunelamiento** Permite transmisiones de una red IPv6 a otra a través de redes IPv4, además de permitir que los hosts IPv6 operen en redes IPv4, o a través de éstas.

En métodos de pila dual, las direcciones IPv4 e IPv6 tienen soporte en la red. Los servidores de aplicaciones y DNS utilizan cualquiera de las dos para transmitir datos.

La traducción utiliza tablas NAT (consulte el capítulo 20) para traducir direcciones IPv6 a las direcciones IPv4 correspondientes, y viceversa, de acuerdo con lo necesario. Entonces las aplicaciones IPv4 interactúan libremente con aplicaciones IPv6. Las transmisiones de IPv6 a IPv6 pasan directamente, permitiendo funcionalidad IPv6 completa.

El entunelamiento se utiliza cuando una red IPv6 necesita transmitir a otra a través de una red IPv4 que no maneja direcciones IPv6. Con el entunelamiento, el paquete IPv6 se encapsula dentro de un paquete IPv4, donde la red IPv4 entonces utiliza la dirección IPv4 para pasar el paquete. Se utilizan varios métodos para entunelamiento, como se muestra aquí, además de manipulación directa:

- **6 sobre 4** Se utiliza dentro de una red para utilizar multidifusión de IPv4 para implementar una LAN virtual que dé soporte a hosts IPv6, sin un enrutador IPv6 (RFC 2529).
- **6 a 4** Se utiliza para permitir que redes IPv6 se conecten a una red IPv4 más grande (Internet), y a través de ellas, al utilizar la dirección de red IPv4 como un prefijo de red IPv6 (RFC 3056).
- **Agentes de túnel** Servicios basados en red que crean túneles (RFC 3053)

Archivos de configuración TCP/IP

Para configurar y administrar su red TCP/IP, se utiliza un conjunto de archivos de configuración en el directorio **/etc**, como se muestra en la tabla 34-6. Estos archivos de configuración especifican información de red como host y nombres de dominio, direcciones IP y opciones de interfaz. En estos archivos se insertan las direcciones IP y los nombres de dominio para otros hosts de Internet a los que quiere acceder. Si configuró su red durante la instalación, ya se encuentra esa información en estos archivos.

Identificación de nombres de host: **/etc/hosts**

Sin la dirección IP única que la red TCP/IP utiliza para identificar equipos, no se podría localizar un equipo particular. Como es difícil utilizar o recordar las direcciones IP, se utilizan en cambio nombres de dominio. Para cada dirección IP, existe un nombre de dominio. Cuando se utiliza un

Dirección	Descripción
Dirección de host	Dirección IP de su sistema; tiene una parte de red para identificar la red en la que está y una parte de host para identificar su propio sistema
Dirección de red	Dirección IP de su red
Dirección de transmisión	Dirección IP para enviar mensajes a todos los hosts en su red a la vez
Dirección de puerta de enlace	Dirección IP de su sistema de puerta de enlace, si tiene una (por lo general, la parte de red de su dirección IP de host con la parte de host establecida en 1)
Direcciones de servidor de nombre de dominio	Direcciones IP de servidores de nombre de dominio que utiliza su red
Máscara de red	Se utiliza para determinar la red y las partes de host de su dirección IP
Archivo	Descripción
/etc/hosts	Asocia nombres de host con direcciones IP; presenta una lista de nombres de dominio para host remotos con sus direcciones IP
/etc/host.conf	Lista de opciones para resolver problemas
/etc/nsswitch.conf	Configuración de servicio de intercambio de nombres
/etc/resolv.conf	Lista de nombres de servidores de nombre de dominio, direcciones IP (servidor de nombres) y nombres de dominios donde los hosts remotos pueden estar ubicados (búsqueda)
/etc/protocols	Lista de protocolos disponibles en su sistema
/etc/services	Lista de servicios de red disponibles, como FTP y Telnet, y los puertos que utilizan

TABLA 34-6 Direcciones y archivos de configuración TCP/IP

nombre para hacer referencia a un equipo en la red, su sistema lo utiliza para traducirlo en su dirección IP asociada. Esta dirección se utiliza para que su red localice el equipo.

Originalmente, cada equipo en la red era responsable de mantener una lista de nombres de host y sus direcciones IP. Esta lista todavía se mantiene en el archivo **/etc/hosts**. Cuando se utiliza un nombre de dominio, su sistema busca su dirección IP en el archivo **hosts**. El administrador del sistema es responsable de mantener esta lista. Debido al crecimiento explosivo de Internet y el desarrollo de más y más redes grandes, la responsabilidad de asociar nombres de dominios y direcciones IP fue tomada por los servidores de nombre de dominio. Sin embargo, el archivo **hosts** todavía se utiliza para almacenar nombres de dominio y direcciones IP de host a los que se accede con frecuencia. Su sistema suele buscar en su archivo **hosts** la dirección IP de un nombre de dominio antes de dar el paso adicional de acceder a un servidor de nombres.

El formato de una entrada de nombre de dominio en el archivo **hosts** es la dirección IP seguida por el nombre de dominio, separados por un espacio. Luego se agregan alias al nombre de host. Después de la entrada, en la misma línea, se inserta un comentario. Un comentario siempre lleva un símbolo **#** antes. Ya se encuentra una entrada en su archivo **hosts** para localhost.localdomain y localhost con la dirección IP 127.0.0.1; localhost es una identificación especial utilizada por su

equipo para permitir que los usuarios de su sistema se comuniquen entre sí de forma local. La dirección IP 127.0.0.1 es una dirección reservada especial utilizada por cada equipo para este propósito. Identifica lo que técnicamente se conoce como un *dispositivo de retroalimentación*. La dirección de host local IPv6 correspondiente es `::1`, que tiene el nombre de host **localhost6**. Nunca debe eliminar las entradas **localhost** ni **localhost6**. Aquí se muestra un ejemplo del archivo `/etc/hosts`:

```
/etc/hosts
127.0.0.1      localhost.localdomain localhost tortuga.mytrek.com
::1           localhost6.localdomain6 localhost6
192.168.0.1    tortuga.mipista.com
192.168.0.2    conejo.mipista.com
192.168.34.56  pango1.mitren.com
```

/etc/resolv.conf

El archivo `/etc/resolv.conf` almacena las direcciones IP de sus servidores DNS junto con los dominios en que se buscará. Una entrada DNS comenzará con el término `nameserver` seguido por la dirección IP del nombre del servidor. Una entrada de búsqueda presentará una lista con las direcciones de dominio de red. Revise este archivo para ver si sus servidores DNS de red aparecen correctamente. Si tiene un enrutador para una red local, DHCP colocará de manera automática una entrada para él en este archivo. El enrutador, a su vez, hará referencia a su servidor de nombre del ISP.

```
/etc/resolv.conf
search mipista.com mitren.com
nameserver 192.168.0.1
nameserver 192.168.0.3
```

NOTA En Red Hat y Fedora, el directorio `/etc/sysconfig/network-scripts` almacena la información de configuración para diferentes dispositivos de conexión de red, como direcciones IP y de red.

/etc/services

El archivo `/etc/services` presenta una lista de servicios de red disponibles en su sistema, como FTP y Telnet, y asocia cada uno con un puerto particular. Aquí, puede saber qué puerto está revisando el servidor Web y qué puerto se utiliza para su servidor FTP. Se le da un alias al servicio, que se especifica después del número de puerto. Después se hace referencia al servicio al utilizar el alias.

/etc/protocols

El archivo `/etc/protocols` presenta una lista de los protocolos TCP/IP soportados por su sistema. Cada entrada muestra el número de protocolo, su identificador de palabra clave y una descripción breve. Visite iana.org/assignments/protocol-numbers para obtener una lista completa.

Servicio de nombres de dominio (DNS)

Cada equipo conectado a una red TCP/IP, como Internet, se identifica con su propia dirección IP. Las direcciones IP son difíciles de recordar, así que una versión de nombre de dominio de cada dirección IP también se utiliza para identificar un host. Un nombre de dominio consta de dos

partes, el nombre de host y el dominio. El nombre de host es el que específico para el equipo y el dominio identifica la red en que participa éste. Los dominios utilizados para Estados Unidos suelen tener extensiones que identifican el tipo de host. Por ejemplo, **.edu** se utiliza para instituciones educativas y **.com** se usa para negocios. Los dominios internacionales suelen tener extensiones que indican el país en que se ubican, como **.de** para Alemania o **.au** para Australia. La combinación de nombre de host, dominio y formas de extensiones forma un nombre único con el que se hace referencia a un equipo. El dominio, a su vez, se divide en más subdominios.

Como ya sabe, un equipo en una red aún puede identificarse sólo con su dirección IP, aunque tenga un nombre de dominio. Se utiliza un nombre de host para hacer referencia a un equipo en una red, pero esto requiere que se utilice el nombre de host para buscar la dirección IP correspondiente en la base de datos. Entonces la red utiliza la dirección IP, no el nombre de host, para acceder al equipo. Antes de la llegada de las redes TCP/IP grandes, como Internet, cada equipo de una red podía mantener un archivo con una lista de todos los nombres de host y direcciones IP de las equipos conectadas en su red. Siempre que se usaba un nombre de host, se buscaba en este archivo y se localizaba la dirección IP correspondiente. Todavía sucede así en su propio sistema para el caso de sistemas remotos a los que accede con frecuencia.

A medida que las redes fueron creciendo, se volvió poco práctico (y, en el caso de Internet, imposible) que cada equipo mantuviera su propia lista con todos los nombres de dominio y direcciones IP. Para proporcionar el servicio de traducir direcciones de dominio a direcciones IP, se desarrollaron las bases de datos de nombres de dominio y se colocaron en sus propios dispositivos. Para encontrar la dirección IP de un nombre de dominio, se envía una consulta a un servidor de nombres, que después busca la dirección IP y la devuelve. En una red grande, varios servidores de nombres cubren diferentes partes de la red. Si un servidor de red no encuentra una dirección IP particular, envía una consulta a otro servidor de nombres que es más probable que la tenga.

Si está administrando una red y necesita configurar un servidor de nombres para ésta, se configura un sistema Linux para que opere como un servidor de nombres. Para ello, debe iniciar un daemon de servidor de nombres y después esperar consultas de nombres de dominio. Un servidor de nombres hace uso de varios archivos de configuración que le permiten contestar peticiones. El software de servidor de nombres que se utiliza en sistemas Linux es el servidor Berkeley Internet Name Domain (BIND), distribuido por Internet Software Consortium (isc.org).

Los sistemas de resolución consultan los servidores de nombres. Se trata de programas diseñados especialmente para obtener direcciones de servidores de nombres. Para utilizar nombres de dominio en su sistema, debe configurar su propio sistema de resolución. Su sistema de resolución local se configura con sus archivos **/etc/host.conf** y **/etc/resolv.conf**. Se utiliza **/etc/nsswitch** en lugar de **/etc/host.conf**.

host.conf

Su archivo **host.conf** presenta una lista de opciones de sistemas de resolución (como se muestra en la tabla 34-7). Cada opción tiene varios campos, separados por espacios o tabuladores. Se utiliza un **#** al principio de una línea para insertar un comentario. Las opciones le indican al sistema de resolución qué servicios usar. El orden de la lista es importante. El sistema de resolución comienza con la primera opción y se mueve a las siguientes por turno. El archivo **host.conf** se encuentra en su directorio **/etc**, junto con otros archivos de configuración.

En el siguiente ejemplo de archivo **host.conf**, la opción **order** indica a su sistema de resolución que busque primero nombres en su archivo **/etc/hosts** local, y después, si eso falla, que consulte servidores de dominio. El sistema no tiene varias direcciones.

Opción	Descripción
order	Especifica métodos de resolución de secuencia de nombres: hosts Revisa un nombre en el archivo /etc/host . bind Consulta a un servidor de nombres DNS en busca de una dirección. nis Utiliza el protocolo de servicio de información de red (NIS, Network Information Service) para obtener una dirección
alert	Revisa direcciones de sitios remotos que intentan obtener acceso a su sistema; se activa o desactiva con las opciones on y off .
nospoof	Confirma direcciones de sitios remotos que intentan acceder a su sistema.
trim	Revisa el archivo de su host local; elimina el nombre de dominio y sólo busca el nombre de host; le permite utilizar sólo un nombre de host en su archivo para una dirección IP
multi	Revisa el archivo de su host local; permite a un host tener varias direcciones IP; se activa o desactiva con las opciones on y off .

TABLA 34-7 Opciones del sistema de resolución host.conf

/etc/host.conf

```
# archivo host.conf
# Busca nombres en el archivo de host y después revisa el DNS
order bind host
# No hay varias direcciones
multi off
```

/etc/nsswitch.conf: intercambio de servicio de nombres

Funciones diferentes en la biblioteca estándar de C deben configurarse para operar en su sistema Linux. Antes, servicios parecidos a base de datos, como el soporte de contraseña, y servicios de nombres como NIS o DNS, accedían directamente a estas funciones, al utilizar un orden de búsqueda fijo. Para GNU C Library 2.x, utilizada en versiones actuales de Linux, esta configuración se realiza mediante un esquema llamado intercambio de servicio de nombres (NSS, Name Service Switch), que está basado en el método del mismo nombre utilizado por Sun Microsystems Solaris 2 OS. Las fuentes de bases de datos y su orden de búsqueda se incluyen en el archivo **/etc/nsswitch.conf**.

El archivo **/etc/nsswitch.conf** almacena entradas para diferentes archivos de configuración que se controlan con NSS. Los archivos de configuración de sistema a los que da soporte NSS se presentan en la tabla 34-8. Una entrada consta de dos campos; el servicio y la especificación de configuración. El servicio consta del archivo de configuración seguido por dos puntos. El segundo campo es la especificación de configuración para ese archivo, que almacena instrucciones sobre la manera en que trabajará el procedimiento de búsqueda. La especificación de configuración contiene especificaciones de servicio y elementos de acción. Las especificaciones de servicio son los servicios que se buscan. Actualmente, algunas especificaciones de servicio válidas son **nis**, **nisplus**, **files**, **db**, **dns** y **compat** (consulte la tabla 34-9). No todas son válidas para cada archivo de configuración. Por ejemplo, el servicio **dns** sólo es válido para el archivo **hosts**, mientras que **nis** es válido para todos los archivos. En el siguiente ejemplo se revisará primero el archivo local **/etc/passwd** y después NIS:

```
passwd: files nisplus
```

Archivo	Descripción
aliases	Alias de correo, utilizados por Sendmail
ethers	Números de Ethernet
group	Grupos de usuarios
hosts	Nombres de host y números
netgroup	Lista de toda la red de hosts y usuarios, utilizada para reglas de acceso; las bibliotecas C antes de glibc 2.1 sólo soportan grupos de red bajo NIS
network	Nombres y números de red
passwd	Contraseñas de usuario
protocols	Protocolos de red
publickey	Claves secretas y públicas para SecureRPC utilizadas por NFS y NIS+
rpc	Llamada a procedimiento remoto de nombres y números
services	Servicios de red
shadow	Contraseñas de usuario shadow

TABLA 34-8 Archivos de soporte para NSS

Un elemento de acción especifica la acción que habrá de tomar un servicio específico; se coloca dentro de corchetes después de un servicio. Una especificación de configuración presenta una lista de varios servicios, cada uno con su propio elemento de acción. En el siguiente ejemplo, la entrada del archivo de red tiene una especificación de configuración que indica que revise NIS y, si no se encuentra, revise el archivo **/etc/protocols**:

protocols: nisplus [NOTFOUND=return] files

Un elemento de acción consta de un estado y una acción. El estado almacena los posibles resultados de un servicio de búsqueda y la acción es la que se tomará si el estado es cierto.

Servicio	Descripción
files	Revisa el archivo /etc correspondiente en busca de la configuración (por ejemplo, /etc/hosts para host); este servicio es válido para todos los archivos.
db	Revisa bases de datos /var/db correspondientes para la configuración; válida para todos los archivos, excepto netgroup .
compat	Válido sólo para archivos passwd , group y shadow .
dns	Revisa el servicio DNS; válido sólo para el archivo hosts .
nis	Revisa NIS; válido para todos los archivos.
nisplus	NIS versión 3.
hesiod	Utiliza Hesiod para búsqueda.

TABLA 34-9 Servicios de configuración de NSS

Actualmente, los valores de estado posibles son SUCCESS, NOTFOUND, UNAVAIL y TRYAGAIN (servicio no disponible temporalmente). Las acciones posibles son return y continue: return detiene el proceso de búsqueda para el archivo de configuración, mientras que continue sigue en el siguiente servicio de la lista. En el ejemplo anterior, si el registro no se encuentra en NIS, el proceso de búsqueda termina.

Aquí se muestra una copia del archivo `/etc/nsswitch.conf`, que presenta una lista de las entradas de uso común. Los comentarios y las entradas comentadas comienzan con un signo #:

```
/etc/nsswitch.conf
# /etc/nsswitch.conf
#
# An example Name Service Switch config file.
passwd:          db files nisplus nis
shadow:          db files nisplus nis
group:           db files nisplus nis
hosts:           files nisplus dns
bootparams:      nisplus [NOTFOUND=return] files
ethers:          files
netmasks:        files
networks:        files
protocols:       files
rcp:             files
services:        files
netgroup:        nisplus
publickey:       nisplus
automount:       files
aliases:         files nisplus
```

Interfaces de red y enrutadores: ifconfig y route

Su sistema se conecta a una red a través de una interfaz de hardware particular, como una tarjeta Ethernet o un módem. Después, los datos que pasan a través de esta interfaz se enrutan a su red. El comando **ifconfig** configura sus interfaces de red y **route** configura las conexiones de red correspondientes. Si configura una interfaz con una herramienta de configuración de red proporcionada por su distribución de Linux, no necesita utilizar **ifconfig** o **route**. Sin embargo, puede configurar directamente las interfaces al utilizar **ifconfig** y **route**, si así lo desea. Cada vez que inicia su sistema, deben establecerse las interfaces de red y sus enrutadores. Esto lo hacen automáticamente los comandos **ifconfig** y **route** ejecutados para cada interfaz por el archivo de inicialización `/etc/rc.d/init.d/network`, que se ejecuta siempre que inicia su sistema. Si está agregando manualmente sus propias interfaces, primero debe configurar la secuencia de comandos de red para que aplique las operaciones **ifconfig** y **route** para sus nuevas interfaces.

ifconfig

El comando **ifconfig** toma como argumentos el nombre de una interfaz y una dirección IP, además de las opciones. Entonces el comando **ifconfig** asigna la dirección IP a la interfaz. Ahora su sistema sabe que esa interfaz existe y que hace referencia a una dirección IP particular. Además, se especifica si una dirección IP es de host o de red. Se utiliza un nombre de dominio para la

dirección IP, suponiendo que el nombre de dominio se encuentra junto con su dirección IP en el archivo `/etc/hosts`. La sintaxis para el comando `ifconfig` es la siguiente:

```
# ifconfig interface -host_net_flag address options
```

Las marcas para `host_net_flag` pueden ser `-host` o `-net` para indicar una dirección IP de host o de red. La marca `-host` es la predeterminada. El comando `ifconfig` tiene varias opciones, que establecen diferentes características de la interfaz, como el número máximo de bytes que se transfiere (`mtu`) o la dirección de transmisión. Las opciones `up` y `down` activan y desactivan la interfaz. En el siguiente ejemplo, el comando `ifconfig` configura una interfaz Ethernet:

```
# ifconfig eth0 192.168.0.1
```

Para una configuración simple como ésta, `ifconfig` genera automáticamente una dirección de transmisión y una máscara de red estándares. La dirección de transmisión estándar es la dirección de red con el número 255 para la dirección de host. En el caso de una red clase C, la máscara de red estándar es 255.255.255.0, mientras que para una red clase A, la máscara de red estándar es 255.0.0.0. Sin embargo, si está conectado a una red con una máscara de red particular y una dirección de transmisión, debe especificarlas cuando utiliza `ifconfig`. La opción para especificar la dirección de transmisión es `broadcast`; para la máscara de red, es `netmask`. En la tabla 34-10 se presenta una lista de diferentes opciones `ifconfig`. En el siguiente ejemplo, `ifconfig` incluye la máscara de red y la dirección de transmisión:

```
# ifconfig eth0 192.168.0.1 broadcast 192.168.0.255 netmask 255.255.255.0
```

Opción	Descripción
<i>Interfaz</i>	Nombre de la interfaz de red, como eth0 para el primer dispositivo Ethernet o ppp0 para el primer dispositivo PPP (módem)
up	Activa una interfaz; está implícito si se especifica la dirección IP
down	Desactiva una interfaz
allmulti	Activa o desactiva el modo promiscuo; un guión (-) antes lo desactiva; permite monitoreo de red
mtu <i>n</i>	Número máximo de bytes que se envían a esta interfaz por transmisión
dstaddr <i>dirección</i>	Dirección IP de destino en una conexión de punto a punto
netmask <i>dirección</i>	Máscara de red IP; un guión Dirección de transmisión; un guión (-) antes lo desactiva antes lo desactiva
broadcast <i>dirección</i>	Dirección de transmisión; un guión (-) antes lo desactiva
point-to-point <i>dirección</i>	Modo de punto a punto para la interacción; si se incluye dirección, se asigna al sistema remoto
hw	Establece la dirección de hardware de la interfaz
<i>Dirección</i>	Dirección IP asignada a una interfaz

TABLA 34-10 Las opciones de `ifconfig`

Una vez que configura su interfaz, se utiliza **ifconfig** con la opción **up** para activarla y con la opción **down** para desactivarla. Si especifica una dirección IP en una operación **ifconfig**, como en el ejemplo anterior, la opción **up** está implícita.

```
# ifconfig eth0 up
```

Las interfaces de punto a punto como IP paralelo (PLIP, Parallel IP), IP de línea serial (SLIP, Serial Line IP) y protocolo de punto a punto (PPP) requieren que incluya la opción **pointopoint**. Un nombre de interfaz PLIP se identifica con el nombre **plip** con un número adjunto. Por ejemplo, **plip0** es la primera interfaz PLIP. Las interfaces SLIP utilizan **slip0**. Las interfaces PPP comienzan con **ppp0**. Las interfaces de punto a punto son las que suelen operar entre dos hosts, como dos equipos conectados a un módem. Cuando se especifica una opción **pointopoint**, necesita incluir la dirección IP del host. En el siguiente ejemplo, una interfaz PLIP se configura para conectar el equipo en la dirección IP 192.168.1.72 con uno en 204.166.254.14. Si se incluyen las direcciones de dominio para estos sistemas en **/etc/hosts**, esos nombres de dominio se utilizan en lugar de direcciones IP.

```
# ifconfig plip0 192.168.1.72 pointopoint 204.166.254.14
```

Si lo necesita, también se utiliza **ifconfig** para configurar su dispositivo de retroalimentación. El nombre del dispositivo de retroalimentación es **lo**, y su dirección IP es la dirección especial 127.0.0.1. En el siguiente ejemplo se muestra la configuración:

```
# ifconfig lo 127.0.0.1
```

El comando **ifconfig** es útil para revisar el estado de una interfaz. Si inserta el comando **ifconfig** junto con el nombre de la interfaz, se despliega información acerca de una interfaz:

```
# ifconfig eth0
```

Para ver si su interfaz de retroalimentación está configurada, se utiliza **ifconfig** con el nombre de interfaz de retroalimentación, **lo**:

```
# ifconfig lo
```

Enrutamiento

Un paquete que es parte de una transmisión toma una cierta *ruta* para llegar a su destino. En redes más grandes, los paquetes se transmiten de un equipo a otro hasta que se llega al equipo deseado. La ruta determina dónde inicia el proceso y a qué equipo necesita enviar su sistema el paquete para que llegue a su destino. En redes más chicas, el enrutamiento puede ser estático (es decir, la ruta de un sistema a otro es fija). Un sistema sabe cómo llegar a otro, al moverse por rutas fijas. Sin embargo, en redes más grandes y en Internet, el enrutamiento es dinámico. Su sistema conoce el primera equipo adonde enviar el paquete, y después ese equipo toma el paquete de ahí, lo pasa a otro equipo, que después determina adónde pasarlo. En el caso de un enrutamiento dinámico, su sistema necesita saber poco. Sin embargo, el enrutamiento estático se vuelve más complejo debido a que tiene que mantener seguimiento de las conexiones de red.

Sus rutas se presentan en su tabla de enrutamiento, en el archivo **/proc/net/route**. Para desplegar la tabla de enrutamiento, inserte **route** sin argumentos (el comando **netstat -r** también desplegará la tabla de enrutamiento):

```
# route
Kernel routing table
Destination Gateway      Genmask      Flags Metric Ref Use  Iface
192.168.0.0   *                255.255.255.0 U        0      0  0   eth0
127.0.0.0     *                255.0.255.0  U        0      0  0   lo
default      192.168.0.1 0.0.0.0      UG        0      0  0   eth0
```

Cada entrada de la tabla de enrutamiento tiene varios campos, que proporcionan información como el destino de la ruta y el tipo de interfaz utilizada. Los diferentes campos se muestran en la tabla 34-11.

Con el argumento **add**, se agregan rutas ya sea para redes con la opción **-net** o con la opción **-host** para interfaces IP (host). La opción **-host** es la predeterminada. Además, después se especifican varios parámetros para información, como la máscara de red (**netmask**), la puerta de enlace (**gw**), el dispositivo de interfaz (**dev**) y la ruta predeterminada (**default**). Si tiene más de una interfaz IP en su sistema, como varias tarjetas Ethernet, debe especificar el nombre de la interfaz al utilizar el parámetro **dev**. Si su red tiene un host de puerta de enlace, se utiliza el parámetro **gw** para especificarlo. Si su sistema está conectado a una red, al menos una entrada debe estar en su tabla de enrutamiento que especifique la ruta predeterminada. Ésta es la ruta tomada por un paquete de mensaje cuando ninguna otra entrada de ruta lleva a su destino. En el siguiente ejemplo se muestra el enrutamiento de una interfaz Ethernet:

```
# route add 192.168.1.2 dev eth0
```

Si su sistema sólo tiene el dispositivo Ethernet como su interfaz IP, se deja fuera el parámetro **dev eth0**:

```
# route add 192.168.1.2
```

Puede eliminar cualquier ruta que establezca al invocar **ifconfig** con el argumento **del** y la dirección IP de esa ruta, como en este ejemplo:

```
# route del 192.168.1.2
```

Campo	Descripción
Destination	Dirección IP de destino de la ruta
Gateway	Dirección IP o nombre de host de su puerta de enlace que utiliza la ruta; * indica que no se utiliza puerta de enlace
Genmask	La máscara de red para la ruta
Flags	Tipo de ruta: U = up, H = host, G = puerta de enlace, D = dinámica, M = modificada
Metric	Costo métrico de la ruta
Ref	Número de rutas que dependen de ésta
Window	TVentana TCP para redes AX.25
Use	Número de veces utilizada
Iface	Tipo de interfaz que utiliza esta ruta

TABLA 34-11 Entradas de tabla de enrutamiento

En el caso de una puerta de enlace, primero se agrega una ruta a la interfaz de puerta de enlace y después una que especifica que es una puerta de enlace. La dirección de la interfaz de puerta de enlace en este ejemplo es 192.168.1.1:

```
# route add 192.168.1.1
# route add default gw 192.168.1.1
```

Si utiliza la puerta de enlace para acceder a la subred, agregue la dirección de esa red (en este ejemplo 192.168.23.0):

```
# route add -net 192.168.23.0 gw dev eth1
```

Para agregar otra dirección IP a una interfaz de red diferente en su sistema, utilice los comandos **ifconfig** y **route** con la nueva dirección IP. En el siguiente comando se configura una segunda tarjeta de red (**eth1**) con la dirección IP 192.168.1.3:

```
ifconfig eth1 192.168.1.3
route add 192.168.1.3 dev eth1
```

Red inalámbrica

Configuración de la red (GNOME) o KNetworkManager (KDE) suelen detectar y configurar automáticamente las conexiones de red. Si es necesario, también se configura una conexión inalámbrica de forma manual. Este tipo de conexión opera de forma muy parecida a una conexión Ethernet estándar, que sólo requiere una dirección IP e información de servidor DNS para conectarse a Internet. Además, la información de conexión inalámbrica es necesaria, como el nombre de la red y el canal utilizado. Una conexión inalámbrica se configura al establecer la información de modo, nombre de red, canal, frecuencia de transmisión y clave.

- **Modo** En el caso de una red simple, que no requiere roaming, el modo suele ser Ad Hoc. Las redes administradas permiten roaming entre diferentes puntos de acceso.
- **Nombre de red (SSID)** El nombre de red se utiliza para identificar una celda como parte de una red virtual.
- **Canal** A partir de 1, seleccione uno con la menor interferencia.
- **Frecuencia de transmisión** Por lo general se establece en Auto para ajustar automáticamente a transmisiones degradadas, pero también se establece una frecuencia específica entre 11 M y 1 M del menú desplegable.
- **Clave** Es la clave de cifrado para la red inalámbrica. Debe de ser la misma para cada celda de su red.

Configuración de la red: GNOME

Casi todas las distribuciones utilizan Configuración de la red para detectar sus conexiones de red, ya sean alámbricas o inalámbricas. Configuración de la red utiliza las capacidades de detección automática de dispositivos de udev y HAL para configurar sus conexiones. Una vez iniciado, Configuración de la red desplegará un Icono de red en la parte derecha del panel superior. Haga clic en este icono para ver una lista de todas las posibles conexiones de red, incluidas las

inalámbricas disponibles. Haga clic con el botón derecho para ver opciones para apagar la conexión (trabajar fuera de línea) o para ver información acerca de la conexión.

NOTA *La versión de KDE de Configuración de la red, KNetworkManager, también detecta conexiones de red. Además le permite configurar conexiones de marcación PPP, aparte de administrar conexiones inalámbricas.*

Con varios puntos de acceso inalámbricos para conexiones de Internet, un sistema tiene varias conexiones de red diferentes para seleccionar, en lugar de una conexión de una sola línea como DSL o cable. Esto es cierto, sobre todo para equipos portátiles que acceden a diferentes conexiones de red inalámbricas en distintas ubicaciones. En lugar de configurar manualmente una nueva conexión cada vez que se encuentra, la herramienta Configuración de la red configura automáticamente y selecciona una conexión que habrá de usarse.

Como opción predeterminada, se preferirá una conexión Ethernet, si está disponible. Las líneas directas que dan soporte a conexiones Ethernet suelen considerarse más rápidas que las inalámbricas. En el caso de conexiones inalámbricas, todavía necesitará seleccionar la que quiere.

La Configuración de la red está diseñada para trabajar en segundo plano, al proporcionar información de estado para su conexión y cambiar de una conexión configurada a otra cuando se necesite. Para configuración inicial, detecta la mayor información posible acerca de la nueva conexión. Opera como una applet de panel de GNOME, que monitorea su conexión y funciona en cualquier distribución de Linux.

La Configuración de la red opera como un daemon con el nombre NetworkManager. Se administra con la secuencia de comandos de servicio NetworkManager.

service NetworkManager start

Si no hay conexiones Ethernet disponibles, Configuración de la red buscará una conexión inalámbrica y revisará los identificadores de conjunto extendido de servicio (ESSIDs, Extended Service Set Identifiers). Si ESSID identifica una conexión ya utilizada, entonces se selecciona automáticamente. Si se encuentran varias, entonces se seleccionará la que se ha usado más recientemente. Si sólo está disponible una conexión nueva, Configuración de la Red espera a que el usuario seleccione una. Sólo se selecciona una conexión si el usuario ha iniciado sesión. Si después se crea una conexión Ethernet, Configuración de la red cambiará a ésta de la conexión inalámbrica.

Configuración de la red es específica del usuario. Cuando un usuario inicia sesión, selecciona la que éste prefiere. La primera vez que un usuario ejecuta Configuración de la red, la applet de notificación desplegará una lista de conexiones posibles para que el usuario seleccione.

Al hacer clic en el icono Configuración de la red, en el panel, se presentará una lista de las conexiones de red disponibles. Los puntos de acceso protegidos por contraseña desplegarán un candado después del nombre. Tendrá que configurar los puntos de acceso escondidos. Seleccione Otras redes inalámbricas, de la lista de applets, para abrir un cuadro de diálogo donde se insertan el ESSID de la red, el tipo de clave y la contraseña.

El hardware de conexión de interfaz de red (NIC, Network Interface Connection) se detecta con el uso de HAL. La información que proporciona Configuración de la red queda a disposición de otras aplicaciones mediante D-Bus. Las características bajo desarrollo incluyen VPN y notificación de aplicación. La Configuración de la red utiliza el cliente DHCPd para obtener información de la red. Para interacción de usuario y notificación, utiliza NetworkManagerInfo.

Configuraciones inalámbricas manuales

La Configuración de la red detectará automáticamente y configurará sus conexiones inalámbricas, al igual que lo hará KNetworkManager. Sin embargo, puede configurar de forma manual sus conexiones con herramientas inalámbricas como `iwconfig`. La configuración inalámbrica hace uso del mismo conjunto de extensiones inalámbricas. El paquete Wireless Tools es un conjunto de herramientas de configuración de red y creación de informes para dispositivos inalámbricos en sistemas Linux. Se les da soporte y se desarrollan como parte del Linux Wireless Extension and Wireless Tools Project, proyecto de fuente abierta mantenido por Hewlett-Packard.

Las herramientas inalámbricas constan de herramientas de configuración y creación de informes que se muestran aquí:

Herramienta	Descripción
<code>iwconfig</code>	Establece las opciones básicas de configuración de la mayor parte de los dispositivos inalámbricos.
<code>iwlist</code>	Despliega la información de estado actual de un dispositivo.
<code>iwspy</code>	Establece la lista de direcciones IP en una red inalámbrica y revisa la calidad de sus conexiones.
<code>iwpriv</code>	Opciones de configuración de acceso específicas para un dispositivo particular.

El dispositivo inalámbrico LAN tendrá un nombre Ethernet, al igual que una tarjeta Ethernet. Los módulos apropiados se cargarán automáticamente, con una lista de sus alias en el archivo `/etc/modprobe.conf`.

iwconfig

El comando **iwconfig** funciona de forma más parecida a **ifconfig**, que configura una conexión de red. System-config-network utiliza esta herramienta para configurar una tarjeta inalámbrica. Como opción, se ejecuta **iwconfig** directamente en una línea de comandos, que especifica ciertos parámetros. Los parámetros agregados le permiten establecer características específicas de redes inalámbricas como el nombre de la red (`nwid`), la frecuencia o canal que utiliza la tarjeta (`freq` o `channel`) y una tasa de bits para la transmisión (`rate`). Consulte la página Man de **iwconfig** para conocer una lista completa de parámetros aceptados. Algunos parámetros de uso común se presentan en la tabla 34-12.

Por ejemplo, para establecer el canal utilizado para el dispositivo inalámbrico instalado como primer dispositivo Ethernet, se utiliza lo siguiente, al configurar el canal en 2:

```
iwconfig eth0 channel 2
```

También se utiliza **iwconfig** para desplegar estadísticas de sus dispositivos inalámbricos, como lo hace **ifconfig**. Inserte el comando **iwconfig** sin argumentos o con el nombre del dispositivo. Se presenta información como nombre, frecuencia, sensibilidad y la tasa de bits. Revise también `/proc/net/wireless` para conocer estadísticas. En lugar de utilizar **iwconfig** directamente para configurar parámetros, éstos se especifican en el archivo de configuración del dispositivo inalámbrico.

iwpriv

El comando **iwpriv** trabaja junto con **iwconfig**, que le permite establecer opciones específicas para un tipo particular de red o dispositivo inalámbrico. Con **iwpriv**, también puede activar

Parámetro	Descripción
essid	Un nombre de red
freq	La frecuencia de la conexión
channel	El canal utilizado
nwid o domain	El ID de red o dominio
mode	El modo operativo utilizado por el dispositivo, como Ad Hoc, Managed o Auto. Ad Hoc = una celda sin punto de acceso, Managed = red con varios puntos de acceso y da soporte a roaming, Master = el nodo es un punto de acceso, Repeater = el nodo reenvía paquetes a otros nodos, Secondary = crea copias de seguridad del maestro o el repetidor, Monitor = sólo recibe paquetes
sens	La sensibilidad, la señal más baja a la que se reciben los datos
key o enc	La clave de cifrado utilizada
frag	Corta los paquetes en fragmentos más pequeños para incrementar una mejor transmisión
bit o rate	Velocidad a la que se transmiten los bits; la opción autorregresa automáticamente a tasas más bajas para canales ruidosos
ap	Un punto de acceso específico
power	Administración de energía para operaciones de reanudación e hibernación

TABLA 34-12 Parámetros de uso común

roaming o seleccionar los puertos que habrán de usarse. Se utiliza el parámetro *privado-comando* para insertar las opciones específicas del dispositivo. En el siguiente ejemplo se activa roaming:

```
iwpriv eth0 roam on
```

iwspy

Su dispositivo inalámbrico puede revisar su conexión con otro dispositivo inalámbrico de donde está recibiendo datos, al reportar la calidad, fuerza de la señal y nivel de ruido de las transmisiones. Su dispositivo mantiene listas de direcciones para diferentes dispositivos de los que recibe datos. Se utiliza la herramienta **iwspy** para establecer o agregar las direcciones que desea que se revisen. Se presentan listas de direcciones IP o versiones de hardware. Un signo + agregará la dirección, en lugar de remplazar la lista completa:

```
iwspy eth0 +192.168.2.5
```

Para desplegar los niveles de calidad, señal y ruido para sus conexiones, se utiliza el comando **iwspy** con el nombre de dispositivo:

```
iwspy eth0
```

iwlist

Para obtener información más detallada acerca de su dispositivo inalámbrico, como todas las frecuencias o canales disponibles, se utiliza la herramienta **iwlist**. Al utilizar el nombre de dispositivo con un parámetro particular, se obtiene información específica de un dispositivo, incluidos la frecuencia, los puntos de acceso, la tasa, las características de energía, los límites de

reintentos y las claves de cifrado utilizadas. **iwlist** permite obtener información de conexiones que fallan. Con el siguiente ejemplo se presentan las frecuencias utilizadas en el dispositivo inalámbrico **eth0**.

```
iwlist eth0 freq
```

linux-wlan

El proyecto linux-wlan (linux-wlan.org) ha desarrollado un conjunto separado de controladores inalámbricos para tarjetas inalámbricas basadas en Prism que dan soporte al nuevo estándar inalámbrico 802.11. El paquete de código fuente original está disponible en el sitio de linux.wlan, en linux-wlan.org. El paquete actual es linux-wlan-ng. Tendrá que desempaquetar y compilar los controladores como se indicó para los paquetes de software de código fuente en el capítulo anterior.

Acceso PPP de línea de comandos: wvdial

Si, por alguna razón, no ha podido configurar una conexión de módem en su sistema X Window System, tal vez tenga que configurarlo en la interfaz de línea de comandos en lugar de un escritorio. Para una conexión PPP de marcado, se utiliza **wvdial**, que es un marcador inteligente que no sólo marca a un servicio de ISP sino que también realiza operaciones de inicio de sesión, al pedirle su nombre de usuario y contraseña. El programa **wvdial** carga primero su configuración del archivo **/etc/wvdial.conf**. Aquí, se coloca información de módem y de la cuenta, incluida la velocidad del módem y el dispositivo serial, además del número telefónico de ISP, nombre de usuario y contraseña. El archivo **wvdial.conf** se organiza en secciones, que comienzan con una etiqueta de sección encerrada en llaves. Una sección almacena variables para diferentes parámetros que son valores asignados, como **username = carlos**. La sección predeterminada almacena valores predeterminados heredados por otras secciones, así que no necesita repetirlas. En la tabla 34-13 se muestran las variables de **wvdial**.

Se usa la utilidad **wvdialconf** para crear un archivo **wvdial.conf** predeterminado automáticamente; **wvdialconf** detectará su módem y establecerá valores predeterminados para características básicas. Puede editar el archivo **wvdial.conf** y modificar las entradas Phone, Username y Password con su información de marcación ISP. Elimine el punto y coma (;) que se encuentra antes para que la entrada deje de ser una cita. Cualquier línea que comience con un punto y coma se ignora, porque es un comentario.

```
$ wvdialconf
```

También se crea un marcador con nombre, como *miisp* en el siguiente ejemplo. Esto es útil si tiene diferentes IPS para iniciar sesión. En el siguiente ejemplo se muestra el archivo **/etc/wvdial.conf**:

```
/etc/wvdial.conf [Modem0]
Modem = /dev/ttyS0
Baud = 57600
Init1 = ATZ
SetVolume = 0
Dial Command = ATDT

[Dialer Defaults]
Modem = /dev/ttyS0
Baud = 57600
```



```
Init1 = ATZ
SetVolume = 0
Dial Command = ATDT

[Dialer miisp]
Username = chris
Password = micontraseña
Modem = /dev/ttyS0
Phone = 5555-55-55
Area Code = 555
Baud = 57600
Stupid mode = 0
```

Para iniciar wvdial, inserte el comando **wvdial**, con el que leerá la información de configuración de conexión del archivo **/etc/wvdial.conf**; wvdial marca al ISP e inicia la conexión PPP, al proporcionar su nombre de usuario y contraseña cuando se le pida.

Variable	Descripción
Inherits	Hereda explícitamente la sección especificada. Como opción predeterminada, las secciones se heredan de la sección [Dialer Defaults].
Modem	El dispositivo wvdial debe usarse como su módem. El predeterminado es /dev/modem .
Baud	La velocidad a la que se comunica wvdial con el módem. La opción predeterminada es 57 600 bauds.
Init1...Init9	Especifica la cadena de inicialización que habrá de usar su módem; wvdial utiliza hasta 9. La opción predeterminada es ATZ para Init1.
Phone	El número predeterminado que quiere que wvdial marque.
Area Code	El código de área, si existe.
Dial Prefix	Cualquier prefijo de marcación necesario (por ejemplo, 70 para deshabilitar la espera de llamada o 9 para una línea externa).
Dial Command	Operación de marcación. La predeterminada es ATDT.
Logín	El nombre de usuario que se utiliza en su ISP.
Logín Prompt	Si su ISP tiene un indicador de comandos de inicio de sesión inusual, se especifica aquí.
Password	La contraseña que se utiliza para su IPS.
Password Prompt	Si su ISP tiene un indicador de comandos de contraseña inusual, se especifica aquí.
Force Address	Dirección IP estática que es obligatoria (para ISP que proporcionan direcciones IP estáticas a usuarios).
Stupid Mode	wvdial no intenta interpretar ninguna petición del servidor de terminal e inicia pppd después de que el módem se conecta.
Auto Reconnect	Si está habilitado, wvdial intenta restablecer una conexión automáticamente, si la otra parte lo desconecta de forma aleatoria. Esta opción está activa como opción predeterminada.

TABLA 34-13 Variables para **wvdial**

```
$ wvdial
```

Puede establecer las configuraciones de conexión para cualquier cantidad de conexiones en el archivo `/etc/wvdial.conf`. Para seleccionar una, inserte su etiqueta como un argumento en el comando `wvdial`, como se muestra aquí:

```
$ wvdial miisp
```

Monitoreo de su red: ping, netstat, tcpdump, EtherApe, Ettercap y Wireshark

Varias aplicaciones disponibles en Linux le permiten monitorear su actividad de red. Las aplicaciones gráficas como EtherApe, Ettercap y Wireshark proporcionan despliegue y registros para permitirle el análisis y la detección de patrones de uso de red. Otras herramientas como ping ofrecen servicios específicos.

Se accede a las herramientas EtherApe, Ettercap y Wireshark en el menú Aplicaciones | Internet. Herramientas como ping, traceroute y netstat se acceden mediante la aplicación Herramientas de red, de GNOME, que se encuentra en el menú Aplicaciones | Herramientas del sistema, además de que también se pueden ejecutar individualmente en la línea de comandos (ventana de terminal).

EtherApe proporciona un despliegue gráfico simple para su actividad de protocolo. El diálogo Preferencias le permite establecer preferencias como el protocolo para verificar y la clase de tráfico para reportar.

ping

Con el programa ping, se revisa si se accede a otro host de su red. El programa ping envía una petición de respuesta al host. Este host regresa entonces una respuesta y se despliega en su pantalla. El programa ping envía continuamente esas solicitudes hasta que las detiene con el comando **break**, CTRL-C. Se ve que una respuesta tras otra recorren su pantalla hasta que detiene el programa. Si ping no accede al host, envía un mensaje que indica que éste no se encuentra disponible. Si ping falla, puede ser una indicación de que su conexión de red no está trabajando. Tal vez sólo sea la interfaz particular, un problema de configuración básico o una mala conexión física. La utilidad ping utiliza el protocolo de control de mensajes (ICMP), que se analizó en el capítulo 20. Las redes pueden bloquear estos protocolos como una medida de seguridad, que también evita que ping funcione. Es probable que una falla de ping sólo indique una precaución de seguridad en la parte de la red consultada.

Para usar ping, inserte **ping** y el nombre del host.

```
$ ping ftp.redhat.com
```

Ettercap

Ettercap es un programa olfateador diseñado para detectar ataques de intermediario. En este tipo de ataque, los paquetes se detectan y se modifican en el tránsito para permitir a un usuario no autorizado acceder a la red. Puede utilizar su interfaz gráfica o su interfaz de línea de comandos. Ettercap realiza olfateo unificado en todas las conexiones, u olfateo de puenteo en una conexión entre interfaces de red. Ettercap utiliza plug-ins para tareas específicas, como `dos_attack` para detectar los ataques denial-of-service y `dns-spoof` para la detección de DNS falso. Revise el panel Ayuda de los plug-ins, o inserte **ettercap -P list** para ver una lista completa. Ettercap se

ejecuta en varios modos, incluidos un modo de texto, uno de cursor de línea de comandos, uno de secuencia de comandos al utilizar éstos en un archivo, e incluso un daemon que registra resultados automáticamente.

Wireshark

Wireshark es un analizador de protocolo de red que permite capturar paquetes transmitidos en su red, al seleccionar y examinar los protocolos que quiere revisar. Puede examinar paquetes de transmisiones particulares, desplegando los datos en formatos legibles. La interfaz Wireshark despliega tres paneles: una lista de paquetes actuales, el árbol de protocolo del paquete seleccionado y un despliegue de los contenidos de paquetes seleccionados. El primer panel ordena las entradas por hora, fuente, destino y protocolo. Existen botones de encabezados para cada uno. Para ordenar un conjunto de entradas por una categoría particular, haga clic en su encabezado. Por ejemplo, para agrupar las entradas por protocolo, haga clic en el botón Protocol; para destinos, use el botón Destination.

Opciones de captura

Para configurar Wireshark, se selecciona la entrada Options en el menú Capture. Esto abre una ventana Options, donde se selecciona la interfaz de red que habrá de verse. Aquí también se seleccionan opciones como el archivo para almacenar su información capturada y el tamaño límite de la captura, junto con un filtro para seleccionar paquetes. Con el modo Promiscuous seleccionado, se ve todo el tráfico de red que pasa por ese dispositivo; en cambio, cuando está desactivado sólo verá los paquetes destinados para ese dispositivo. Luego se hace clic en el botón Start para iniciar Wireshark. Para iniciar y detener Wireshark, se seleccionan las entradas Stop y Start en el menú Capture.

- La opción Capture Files le permite seleccionar un archivo donde se guardan sus capturas. Si no se selecciona un archivo, los datos sólo se despliegan en la ventana de Wireshark. Si quiere mantener en ejecución continua una instantánea de su tráfico de red, se utilizan búferes de anillo. Se trata de series de archivos que se utilizan para guardar datos capturados. Cuando se llenan, la captura comienza a guardar de nuevo desde el principio, etc. Revise “Uso de varios archivos” para habilitar esta opción.
- Las opciones de Display controlan si los paquetes se despliegan en tiempo real en la ventana Wireshark.
- Limits le permite establecer un límite para el tamaño del paquete de captura.
- El filtro Capture le permite seleccionar el tipo de protocolo que quiere revisar.
- Name resolution habilita el despliegue de nombres de host y de dominio, en lugar de direcciones IP, si es posible.

Filtros de Wireshark

Un filtro le permite seleccionar paquetes que relacionan criterios específicos, como paquetes de un host particular. Los criterios se especifican al utilizar expresiones compatibles con la Packet Capture Library y se implementan con **tcpdump**. Los filtros de Wireshark utilizan expresiones similares a las utilizadas por este comando. Revise la página Man de **tcpdump** para conocer descripciones detalladas.

Puede configurar Search filter, en el panel Find (menú Edit), para buscar ciertos paquetes, o configurar Capture Filter, en el panel Opciones (menú Capture), para seleccionar qué paquetes registrar. La ventana de filtro es la misma para ambos. En esta ventana se selecciona el protocolo

que quiere buscar o capturar. El nombre de filtro y la cadena aparecerán en el segmento Properties. También puede insertar sus propias cadenas, al configurar un nuevo filtro propio. La cadena debe ser una expresión de filtro.

Para crear un nuevo filtro, inserte el nombre que quiera darle, en el cuadro Filter Name. Después, en el cuadro Filter String, inserte la expresión de filtro, como **icmp**, y haga clic en New. Su nuevo filtro aparecerá en la lista. Para cambiar un filtro, selecciónelo y cambie su expresión en el cuadro Filter String y después haga clic en Change.

Una expresión de filtro consta de un ID, como el nombre o número de host y un calificador. Los calificadores vienen en tres tipos: **type**, **direction** y **protocol**. El tipo hace referencia al host, red o puerto. Los calificadores de tipo son **host**, **net** y **port**. **Direction** selecciona los paquetes fuente, destino, o ambos. El calificador de fuente es **src** y el de destino es **dst**. Sin un calificador de destino, se seleccionan ambas direcciones. El protocolo le permite especificar paquetes para un cierto protocolo. Los protocolos se representan al utilizar sus nombres en minúsculas, como **icmp** para ICMP. Por ejemplo, la expresión para mostrar todos los paquetes que vienen de un host particular sería **src host nombredelhost**, donde *nombredelhost* es el host de la fuente. En el siguiente ejemplo se desplegarán los paquetes del host 192.168.0.3:

```
src host 192.168.0.3
```

Al utilizar sólo **host** se revisarán todos los paquetes salientes y entrantes de ese host. El calificador **port** revisará los paquetes que pasan a través de un puerto particular. Para revisar un protocolo particular, se utiliza el nombre de éste. Por ejemplo, para revisar todos los paquetes ICP se utilizaría la expresión

```
icmp
```

También existen varios calificadores especiales que le permiten controlar más su selección. El calificador **gateway** le permite detectar paquetes que pasan por la puerta de enlace. Los calificadores **broadcast** y **multi-cast** detectan paquetes transmitidos a una red. Los calificadores **greater** y **less** se aplican a números como puertos o direcciones IP.

Se combinan expresiones en una sola, booleana, al utilizar **and**, **or** o **not**. Esto le permite crear un filtro más definido. Por ejemplo, para capturar sólo los paquetes ICMP que vienen del host 192.168.0.2, se utiliza

```
src host 192.168.0.3 and icmp
```

tcpdump

Al igual que Wireshark, **tcpdump** capturará paquetes de red, guardándolos en un archivo donde se examinan. **tcpdump** opera totalmente desde la línea de comandos. Tendrá que abrir una ventana de terminal para ejecutarlo. Mediante el uso de varias opciones, se refina su captura, al especificar los tipos de paquetes que quiere. **tcpdump** utiliza un conjunto de opciones para especificar acciones que quiere tomar, que incluyen limitar el tamaño de la captura, decidir a qué archivo guardar y seleccionar cualquier filtro que quiera aplicar. Revise la página Man de **tcpdump** para ver una lista completa.

- La opción **-i** le permite especificar una interfaz en que se escuchará.
- Con la opción **-c**, se limita el número de paquetes que habrá de capturarse.
- Los paquetes se dirigirán a la salida estándar, como opción predeterminada. Para guardarlos en un archivo, se utiliza la opción **-w**.

- Después se lee un paquete al utilizar la opción **-r** y aplicar una expresión de filtro a éste.

El comando **tcpdump** toma como argumento una expresión de filtro que se utiliza para refinar su captura. Wireshark utiliza la misma expresión de filtro que **tcpdump** (consulte el análisis acerca de los filtros en Wireshark).

netstat

El programa **netstat** proporciona información en tiempo real del estado de sus conexiones de red, además de las estadísticas de red y la tabla de enrutamiento. El comando **netstat** tiene varias opciones que se utilizan para obtener diferentes tipos de información de su red.

```
# netstat
Active Internet connections
Proto Recv-Q Send-Q Local Address Foreign Address (State) User
tcp 0 0 tortuga. mytrek.com:01 pangol.mytrain.com. :ftp ESTABLISHED daniel
Active UNIX domain sockets
Proto RefCnt Flags Type State Path
unix 1 [ ACC ] SOCK_STREAM LISTENING /dev/printer
unix 2 [ ] SOCK_STREAM CONNECTED /dev/log
unix 1 [ ACC ] SOCK_STREAM LISTENING /dev/nwapi
unix 2 [ ] SOCK_STREAM CONNECTED /dev/log
unix 2 [ ] SOCK_STREAM CONNECTED
unix 1 [ ACC ] SOCK_STREAM LISTENING /dev/log
```

El comando **netstat** sin opciones presenta una lista de las conexiones de red en su sistema. Primero, se aparecen las conexiones activas TCP y después se incluyen los conectores de dominio activos. Los conectores de dominio contienen procesos utilizados para configurar comunicaciones entre su sistema y otros. Se utiliza **netstat** con la opción **-r** para desplegar la tabla de enrutamiento y **netstat** con la opción **-i** despliega el uso de diferentes interfaces de red.

Alias de IP

En algunos casos, tal vez quiera asignar dos o más direcciones IP a un solo sistema Linux, que sólo tiene una interfaz de red. Por ejemplo, tal vez quiera ejecutar diferentes sitios Web que se acceden con direcciones IP separadas en este sistema. En efecto, está configurando un alias para su sistema, otra dirección por la cual se accede. En realidad, está asignando dos direcciones IP a la misma interfaz de red (por ejemplo, al asignar dos direcciones IP a una tarjeta Ethernet). Este procedimiento, conocido como *alias de IP*, se utiliza para configurar varios hosts virtuales basados en IP para servicios de Internet. Este método le permite ejecutar varios servidores Web en la misma máquina al utilizar una sola interfaz (o más de una en varias interfaces). Consulte los capítulos 22 y 23 para conocer información de FTP y servidores Web acerca de hosts virtuales.

Para configurar un alias IP basta con configurar una interfaz de red en su sistema para que escuche la dirección IP agregada. Su sistema necesita conocer qué direcciones IP debe escuchar y en qué interfaz. Es posible configurar los alias de IP con los comandos **ifconfig** y **route** o una herramienta administrativa de red.

Para agregar otras direcciones a la misma interfaz, necesita calificar ésta al agregar dos puntos y un número. Por ejemplo, para agregar otra dirección IP a la primera tarjeta Ethernet (**eth0**) se agrega **:0** a su nombre de interfaz, **eth0:0**. En el siguiente ejemplo se muestran los comandos **ifconfig** y **route** para la interfaz Ethernet 192.168.1.2 y dos alias IP agregados: 192.168.1.100 y 192.168.1.101. Para agregar otra dirección IP a la misma interfaz, se utiliza **eth0:1**, que incrementa el calificador, etc.

El primer comando **ifconfig** asigna la dirección IP principal, 192.168.1.2, al primer dispositivo Ethernet, **eth0**. Después, otras dos direcciones IP se asignan al mismo dispositivo. En el primer comando **route**, la ruta de red se configura para el dispositivo Ethernet y después las rutas se configuran para cada interfaz IP. Las interfaces para dos alias se indican con **eth0:0** y **eth0:1**:

```
ifconfig eth0 192.168.1.2
ifconfig eth0:0 192.168.1.100
ifconfig eth0:1 192.168.1.101
route add -net 192.168.1.0 dev eth0
route add -host 192.168.1.2 dev eth0
route add -host 192.168.1.100 dev eth0:0
route add -host 192.168.1.101 dev eth0:1
```

Es necesario que el kernel dé soporte al alias de IP antes de utilizarlo. Si su kernel no es compatible, tal vez tenga que reconstruir el kernel (incluyendo el soporte a alias de IP), o utilizar módulos que se cargan para agregar alias de IP.

Soporte a InfiniBand

Desde el kernel 2.6.10, Linux incluye soporte a InfiniBand. Esta nueva arquitectura E/S se utiliza para remplazar las de bus utilizadas en los sistemas actuales. A menudo, InfiniBand se utiliza como reemplazo para conexiones de red locales. Se implementa actualmente en superequipos y grupos de servidor de red. Encuentre más acerca de InfiniBand en la página infiniband.sourceforge.net del proyecto Linux InfiniBand Project. El soporte para InfiniBand se está proporcionando como un proyecto de fuente abierta por OpenFabrics Alliance, openib.org.

Los sistemas de hoy en día utilizan el bus PCI o sus versiones mejoradas, PCI X o PCI Express. Esta arquitectura de PCI de E/S utiliza un bus compartido que sólo alcanza medio gigabit de rendimiento. Los servidores de grupos ya están llegando a los límites de este método de E/S. Una tecnología alterna es la arquitectura de E/S de InfiniBand. InfiniBand utiliza canales seriales en lugar de un bus compartido. Las velocidades empiezan en 2.6 Gb por segundo y llegan hasta 30 Gb por segundo. En lugar de tener un bus procesando transacciones controladas por un solo host, InfiniBand utiliza una arquitectura de canal de puerto a puerto donde varias conexiones se administran mediante el uso de diferentes canales. Esta fábrica de arquitectura de conexión permite a InfiniBand conectar diferentes nodos. PCI Express está limitado a usar un bus local, que conecta un CPU con periféricos. InfiniBand, en contraste, soporta conexiones de red, permitiéndole implementar esencialmente una intranet de alta velocidad local, además de conexiones de alta velocidad compartidas a dispositivos de almacenamiento independientes como discos duros. Al utilizar un cable InfiniBand en lugar de un cable Ethernet, se conectan sus hosts y dispositivos compartidos (con un máximo de hasta 15.24 metros). El protocolo IPoIB (IP sobre InfiniBand) le permite implementar redes IP mediante conexiones InfiniBand, y el protocolo RDMA se utiliza para dispositivos de almacenamiento remotos. La velocidad más alta de una conexión InfiniBand reviste particular importancia para servidores con una elevada necesidad de ancho de banda. Además, el protocolo de conectores seguros configura conexiones InfiniBand de alta velocidad para flujos, y el protocolo SCSI RDMA Protocol (SRP) administra conexiones a discos duros.

Las máquinas con PCI Express pueden manejar el mayor ancho de banda proporcionado por una conexión InfiniBand. Una tarjeta de adaptador de canal de host (HCA, Host Channel Adapter) colocada en una ranura PCI tiene conectores InfiniBand e interactuará con transmisiones InfiniBand con el bus PCI Express. Los controladores de varios HCA ya están incorporados en el kernel, al igual que los controladores de protocolo.

Configuración automática de red con IPv6, DHCPv6 y DHCP

Muchas redes proporcionan ahora configuración automática de IPv6 o el servicio DHCP (Dynamic Host Configuration Protocol, protocolo de configuración dinámica de host), que proporciona configuración de red automática para todos sus hosts conectados. La configuración automática es sin estado, como es el caso de IPv6, o con estado, como con DHCP. La configuración automática IPv6 sin estado no requiere un servicio o fuente independiente para conectarse a la red. Es una operación plug-and-play, donde las interfaces de red y los enrutadores de hardware determinan directamente las direcciones correctas. DHCP es un método más antiguo, que requiere un servidor separado para administrar y asignar todas las direcciones. Si este servidor alguna vez falla, los hosts no se conectan.

Con el protocolo DHCP, un administrador utiliza un grupo de direcciones IP de donde el administrador asigna una a un host, cuando se necesita. El protocolo también se utiliza para proporcionar toda la información necesaria de conexión de red, como la dirección de puerta de enlace para la red o la máscara de red. En lugar de tener que configurar cada host de forma separada, un servidor DHCP central maneja la configuración de red. El tiempo que una dirección se utiliza se controla por medio de arrendamientos, que hacen efectivo el uso de direcciones disponibles. Si su red está configurando su sistema con DHCP, no tendrá que configurarlo.

Existen dos versiones de DHCP, una para el protocolo IPv4 original y otra, conocida como DHCPv6, para el protocolo IPv6. Éste incluye información para configuración dinámica que le falta al protocolo IPv4. En este aspecto, el protocolo IPv4 es mucho más dependiente de DHCP que IPv6.

Configuración automática sin estado de IPv6

En una red IPv6, el protocolo IPv6 incluye información que configura directamente un host. Con IPv4 tiene que configurar de forma manual cada host o depender de un servidor DHCP para proporcionar información de configuración. Con IPv6, la información de configuración se integra directamente en el protocolo de Internet. La configuración automática de una dirección IPv6 se describe con detalle en RFC 2462.

Las capacidades de configuración automática de IPv6 se consideran sin estado, lo que significa que configuran directamente un host sin recurrir a un servidor externo. De manera alterna, DHCP, incluido DHCPv6, es con estado, porque el host depende de un servidor externo DHCP para proporcionar información de configuración. La configuración automática sin estado tiene la ventaja de que los hosts no dependen de un servidor DHCP para mantener conexiones con una red. Las redes incluso se vuelven móviles, al conectarse en una subred u otra, generando automáticamente direcciones de acuerdo con lo necesario. Los hosts ya no están atados a un servidor DHCP particular.

Generación de la dirección local

Para configurar de forma automática hosts en una red local, IPv6 usa cada dirección MAC de hardware del dispositivo de red. Esta dirección se utiliza para generar una dirección temporal, con la que es posible consultar y configurar el host.

La dirección MAC se utiliza para crear una dirección de vínculo local, que tiene un prefijo **FE80::0**, seguida por un identificador de interfaz. El prefijo de vínculo local se utiliza para hosts conectados de forma física, como en una red local pequeña.

Luego se realiza una prueba de exclusividad en la dirección generada. Al utilizar el protocolo de descubrimiento de entorno (NDP, Neighbor Discovery Protocol), se revisan otros hosts en la red para ver si alguno ya está utilizando una dirección de vínculo local. Si ningún otro host está utilizando la dirección, entonces ésta se asigna a la red local. En este punto, el host sólo tiene una dirección local válida dentro de la red física. Las direcciones de vínculo local no se enrutan a una red más grande.

Generación de dirección completa: anuncios de enrutador

Una vez que se ha determinado la dirección de vínculo local, se consulta al enrutador de la red en busca de información de configuración adicional. La información puede ser con estado, sin él, o ambas. En el caso de una configuración sin estado, se proporciona directamente información como la dirección de red, mientras que para la configuración con estado, el host se envía a un servidor DHCPv6 donde obtiene información de configuración. Las dos trabajan juntas. A menudo el método sin estado se utiliza para direcciones y el servidor DHCPv6 con estado se utiliza para proporcionar otra información de configuración como las direcciones de servidor DNS.

En el caso de direcciones sin estado, el enrutador proporciona la dirección mayor de red, como la dirección de Internet de la red. Esta dirección se agrega después a la local, al reemplazar el prefijo de vínculo local original, al darle una dirección de Internet global completa o, en el caso de las redes privadas, direcciones de sitio local. Los enrutadores anunciarán de forma rutinaria esta información de dirección, aunque también se pide específicamente. NDP se utiliza para consultar la información. Antes de que la dirección se asigne oficialmente, un procedimiento de detección de direcciones duplicadas revisa si la dirección ya está en uso. El proceso depende de que el enrutador proporcione la información de direccionamiento apropiada en forma de anuncios de enrutador. Si no existe un enrutador, o no hay anuncios de enrutador, entonces debe usarse un método con estado, como DHCPv6, o configuración manual, para proporcionar las direcciones.

En la figura 35-1 se muestra una red configurada con configuración automática de dirección sin estado. Cada host determina primero su identificador de interfaz al utilizar su propia dirección de hardware MAC para crear direcciones temporales de vínculo local para cada host que utiliza el prefijo **FE08::0**. Esto permite comunicación inicial con el enrutador de la red. Luego, el enrutador utiliza su prefijo de red para crear direcciones de Internet completas, al reemplazar el prefijo de vínculo local.

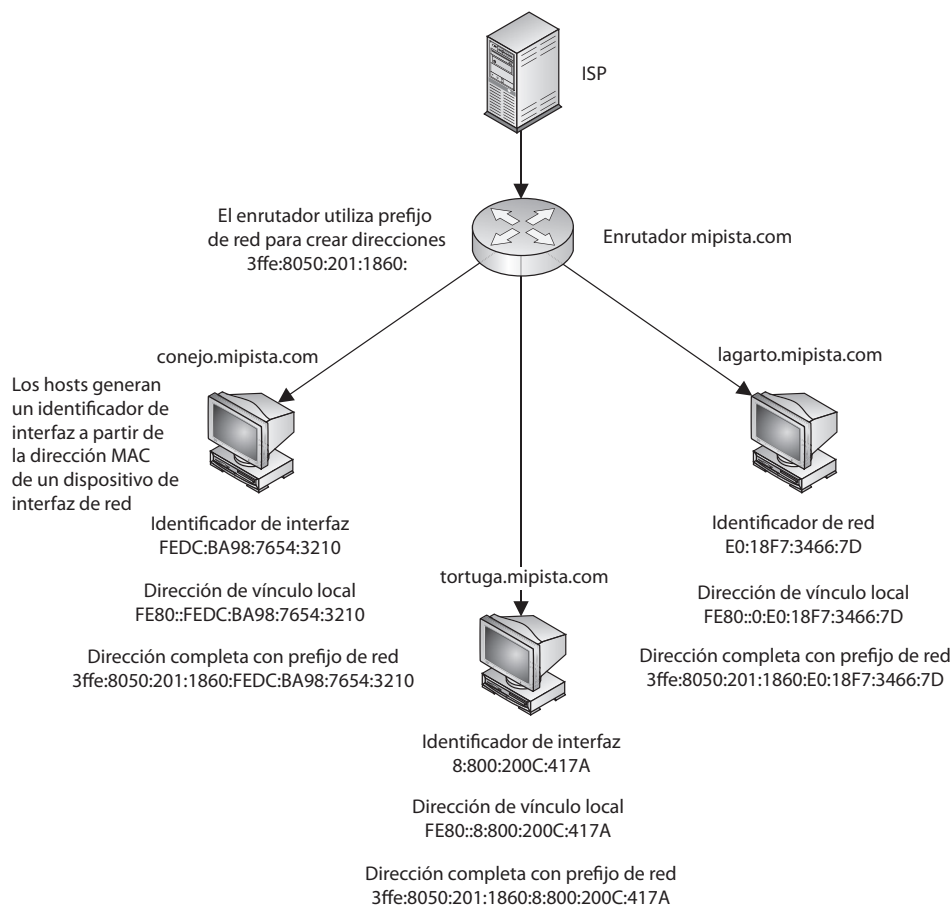


FIGURA 35-1 Configuración automática de direcciones IPv6 sin estado

Renumeración de enrutador

Con IPv6, los enrutadores tienen la capacidad de volver a numerar las direcciones de sus redes al cambiar el prefijo de red. La renumeración se lleva a cabo a través del protocolo de renumeración del enrutador (RR, Router Renumbering). (Consulte RFC 2894 para conocer una descripción completa de la renumeración de enrutador.) A menudo, ésta se utiliza cuando una red cambia los proveedores de ISP y es necesario cambiar la dirección de red de todos los host (véase la figura 35-2). También se utiliza para redes móviles donde una red se conecta a diferentes redes más grandes, volviéndose a numerar en cada ocasión.

Con la renumeración, los enrutadores colocan un límite de tiempo a las direcciones, de manera similar al tiempo de arrendamiento en DHCP, al especificar un límite de expiración para el prefijo de red cuando se genera la dirección. Para facilitar la transición, las interfaces mantienen sus direcciones antiguas como direcciones obsoletas, mientras se utilizan primero las nuevas. Éstas

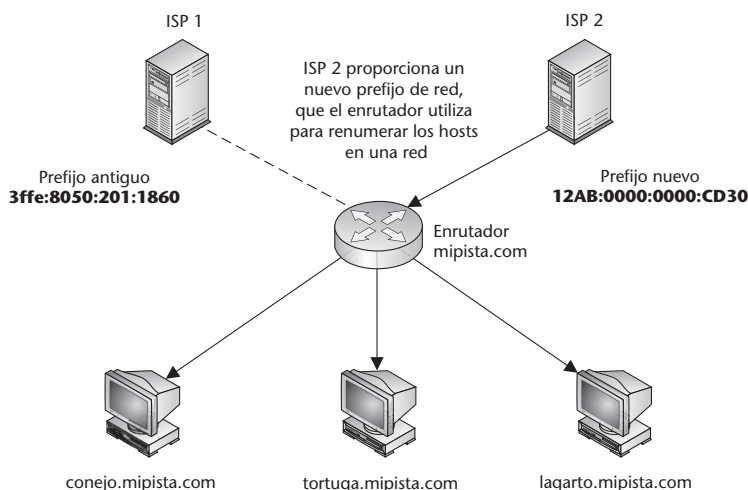


FIGURA 35-2 Renumeración de enrutador con configuración automática IPv6

serán las direcciones preferidas para cualquier conexión nueva, mientras las obsoletas se utilizan para conexiones antiguas. En efecto, un host tiene dos direcciones, una obsoleta y otra preferida. La regeneración de direcciones en realidad cambia el nombre del host.

Configuración automática con estado de IPv6: DHCPv6

La versión IPv6 de DHCP (DHCPv6) proporciona configuración automática con estado para las redes que todavía quieran un servicio como DHCP en redes IPv6. DHCP para IPv6 proporciona información de configuración de un servidor, como DHCP, pero es un protocolo totalmente diferente de la versión IPv4, con distintas opciones y capacidades. Como un proceso de configuración con estado, la información la proporciona un servidor independiente.

DHCPv6 utiliza su propio conjunto de opciones para el cliente y el servidor. El servidor DHCP para IPv6 se llama **dhcp6s** y el cliente DHCP para IPv6 es **dhcp6c**. Los archivos de configuración correspondientes son **/etc/dhcp6c** y **/etc/sysconfig/dhcp6s**. Una secuencia de comandos de servicio, **/etc/init.d/dhcp6s**, se utiliza para administrar el servidor **dhcp6s**.

Como en el caso de la configuración automática IPv6, primero se genera automáticamente el identificador de host para una dirección local. Ésta contiene un identificador de dirección de host generada a partir de la dirección MAC de la interfaz de host.

Una vez determinada la dirección de vínculo local, se consulta el enrutador en busca del servidor DHCPv6. Esta información se proporciona en anuncios de enrutador que se difunden regularmente. En este punto, el servidor proporciona los dos diferentes tipos de información con estado: información de dirección y de otra configuración. Al host se le notifica qué tipos de información con estado se proporcionan. Si el servidor DHCPv6 no proporciona la información de dirección, entonces las direcciones se determinarán al usar el método de configuración automática sin estado descrita en la sección anterior. Si se proporciona la información de dirección, entonces ésta se obtendrá del servidor en lugar de que se genere directamente. Antes de arrendar una dirección, el servidor ejecutará un procedimiento de detección de dirección duplicada para asegurarse de que la dirección es única.

NOTA El direccionamiento con estado DHCPv6 es útil para situaciones en que se necesita mantener control estricto sobre la dirección IP del host, mientras que el direccionamiento IPv6 es mejor para situaciones donde la dirección IP real no es importante, siempre y cuando las conexiones sean efectivas.

Linux como enrutador IPv6: radvd

En el caso de un sistema Linux que opera como enrutador, se utiliza **radvd** (Router ADvertisement Daemon, daemon de anuncio de enrutador) para anunciar direcciones, especificando un prefijo de red en el archivo **/etc/radvd.conf**. El daemon **radvd** detectará las solicitudes hechas por los hosts de dirección de red de enrutador, conocidas como solicitudes de enrutador, y les proporcionará una dirección de red al utilizar un anuncio de enrutador. Estos anuncios de enrutador también se difundirán para proporcionar la dirección de red a cualquier host que no envíe peticiones. Para que **radvd** trabaje, tendrá que activar el reenvío de IPv6. Utilice **sysctl** y asigne un valor de 1 a **net.ipv6.conf.all.forwarding**. Para iniciar el daemon **radvd**, se utiliza la secuencia de comandos **radvd**. Para revisar las direcciones de red que **radvd** está enviando, se utiliza **radvdump**.

Tendrá que configurar el daemon **radv** usted mismo, al especificar la dirección de red para la transmisión. Sin embargo, la configuración es muy simple, porque la dirección completa se generará automáticamente al utilizar la dirección de hardware del host. Una configuración consta de entradas de interfaz, que a su vez presentan listas de opciones de interfaz, definiciones de prefijo y opciones, junto con definiciones de enrutador, si son necesarias. La configuración se coloca en el archivo **/etc/radvd.conf**, que busca algo como esto:

```
interface eth0 {
    AdvSendAdvert on;
    prefix fec0:0:0:0::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
    };
};
```

Para esto se supone que una interfaz se utiliza para la red local, **eth0**. Esta configuración de interfaz muestra una opción de interfaz (**AdvSendAdvert**) y una definición de prefijo, junto con otras dos opciones de prefijo (**AdvOnLink** y **AdvAutonomous**). Si desea especificar opciones de prefijo para un prefijo específico, agréguelas entre paréntesis después de la definición del prefijo. Esta definición especifica su dirección de red IPv6. Si una red de área local tiene su propia dirección de red, necesitará proporcionar su dirección de prefijo de red IPv6. En el caso de una red privada, como una casera, se utiliza el prefijo IPv6 de sitio local, que opera como las direcciones de red privada IPv4, 192.168.0. En el ejemplo anterior utiliza una dirección de sitio local utilizada por las redes IPv6 privadas, **fec0:0:0:0::**, que tiene un tamaño de 64 bits.

La opción de interfaz **AdvSendAdvert** activa los anuncios de dirección de red a los hosts. La opción de prefijo de red **AdvAutonomous** proporciona configuración de dirección automática y **AdvOnLink** sólo significa que las solicitudes de host se reciben en la interfaz de red especificada.

Una segunda interfaz de red se utiliza para conectar el sistema Linux a un ISP o una red más grande. Si el ISP da soporte a IPv6, basta con enviar una solicitud de enrutador al enrutador del ISP. Esto genera automáticamente su dirección de Internet al utilizar la dirección de hardware de la interfaz de red que se conecta a Internet y la dirección de red anunciada del enrutador ISP. En la figura 35-2, que ya se mostró, la interfaz de red **eth0** se conecta a la red local, mientras que **eth1** se conecta a Internet.

DHCP para IPv4

DHCP proporciona información de configuración a sistemas conectados a una red TCP/IP IPv4, ya sea Internet o una intranet. Las máquinas en la red operan como clientes DHCP, que obtienen su información de configuración de un servidor DHCP en su red. Una máquina en la red ejecuta un daemon de cliente DHCP que recibe automáticamente su información de configuración de red del servidor DHCP de la red. La información incluye su dirección IP, junto con el servidor de nombres, la puerta de enlace y las direcciones proxy de la red, incluida la máscara de red. No tiene que configurarse nada de forma manual en el sistema local, excepto especificar el servidor DHCP de donde debe obtenerse su configuración de red. Esto tiene la ventaja agregada de centralizar el control sobre la configuración de red para sistemas diferentes en la red. Un administrador de red administra las configuraciones de todos los sistemas en la red del servidor DHCP.

NOTA DHCP se basa en BOOTP desarrollado para estaciones de trabajo sin disco. Revise la documentación de DHCP para conocer más opciones específicas de máquinas diseñadas para trabajar con BOOTP.

Un servidor DHCP también da soporte a varios métodos para asignación de dirección IP: automática, dinámica y manual. La automática asigna una dirección IP permanente a un host. La manual asigna una dirección IP diseñada por el administrador de red. Con la dinámica, un servidor DHCP asigna una dirección IP a un host en la red sólo cuando éste necesita utilizarla. La asignación dinámica toma las direcciones de un almacén de direcciones IP que los hosts utilizan cuando se necesitan y se dejan cuando termina.

La versión actual de DHCP ahora da soporte al protocolo DHCP de protección contra fallas, en que los dos servidores DHCP dan soporte al mismo almacén de direcciones. Si uno falla, el otro continúa proporcionando servicios DHCP a una red. Ambos servidores están sincronizados y tienen la misma copia de información de soporte de red para cada host en la red. Los servidores primario y secundario en este esquema están designados con la expresión primaria y secundaria.

Hay varios servidores y clientes DHCP para diferentes sistemas operativos. Para Linux, se obtiene software DHCP de Internet Software Consortium (ISC) en isc.org. El paquete de software incluye un servidor DHCP, un cliente y un agente de transmisión. Linux incluye un servidor y un cliente DHCP. El cliente se llama **dhclient**, y el servidor IPv4 se llama **dhcpcd**.

Configuración de host de clientes IPv4 de DHCP

Para que los hosts usen un servidor DHCP basta con configurar opciones para el dispositivo de interfaz de red del host, como una tarjeta Ethernet. En el caso de un host de Linux, puede utilizar una herramienta de red de la distribución para determinar que el host acceda de forma automática al servidor DHCP para obtener información de red. En el panel de una herramienta de red para configurar la conexión de Internet, suele encontrarse una casilla de verificación para seleccionar DHCP. Al hacer clic en esta casilla se habilitará DHCP.

El soporte a clientes lo proporciona la herramienta **dhclient**. Cuando su red inicia, utiliza **dhclient** para configurar su conexión DHCP. Aunque las opciones predeterminadas suelen ser adecuadas, se configura más el cliente DHCP al utilizar el archivo **/etc/dhclient.conf**. Consulte la página Man de **dhclient.conf** para conocer una lista detallada de opciones de configuración. **dhclient** mantiene la información de arrendamiento en la conexión DHCP del archivo **/var/lib/dhcp/dhclient.leases**. También se ejecuta **dhclient** directamente para configurar las conexiones DHCP.

dhclient

Configuración del servidor IPv4 DHCP

Para detener o iniciar el servidor DHCP, se utiliza el comando **dhcpd** en el directorio **/etc/rc.d/init.d**. Utilice la herramienta **redhat-config-services** o el comando **service** con las opciones **start**, **restart** y **stop**. En el siguiente ejemplo se inicia el servidor DHCP. Se utiliza la opción **stop** para apagarlo y **restart** para reiniciarlo.

```
service dhcpd start
```

Las direcciones IP asignadas de forma dinámica, conocidas como *arrendamientos*, se asignarán por un tiempo determinado. Cuando un arrendamiento expira, se extiende o se genera una nueva. Los arrendamientos actuales se encuentran en el archivo **dhcp.leases** ubicado en el directorio **/var/lib/dhcp**. Una entrada de arrendamiento especificará la dirección IP y los tiempos de inicio y fin del arrendamiento, junto con el nombre de host del cliente.

Los argumentos y opciones del servidor DHCP se especifican en el archivo **/etc/sysconfig/dhcpd**. Los argumentos de dispositivo de red especifican en cuál de estos últimos se debe ejecutar el servidor DHCP. También se especifican opciones como el archivo de configuración que se usará o el puerto en que se escuchará. Los argumentos de dispositivo de red son necesarios si tiene dos o más interfaces de red en su sistema, pero quiere que el servidor DHCP sólo opere en las conexiones seleccionadas. Estos argumentos se incluyen en el archivo **/etc/sysconf/dhcpd** al utilizar la configuración DHCPARGS. En el siguiente ejemplo se indica que sólo se ejecute el servidor DHCP en el dispositivo de red Ethernet **eth0**:

```
DHCPARGS=eth0
```

Este tipo de configuración es útil para sistemas de puerta de enlace que se conectan a una red local y una red más grande, como Internet, mediante diferentes dispositivos de red. En la conexión de Internet, tal vez quiera ejecutar el cliente DHCP para que reciba una dirección IP de un ISP, y en la conexión de red local quizás quiera ejecutar el servidor DHCP para que asigne direcciones IP a hosts locales.

El archivo de configuración del servidor DHCP es **/etc/dhcpd.conf**; en él, se especifican parámetros y declaraciones que definen la manera en que el servidor DHCP accederá a diferentes clientes DHCP de su red, junto con opciones que definen la información que el servidor DHCP pasa a los clientes. Estos parámetros, declaraciones y opciones se definen de forma global para ciertas subredes o para hosts específicos. Los parámetros globales, declaraciones y opciones aplican a todos los clientes, a menos que los invaliden otras declaraciones y opciones correspondientes en las instrucciones de host y subred. En el aspecto técnico, todas las entradas de un archivo **dhcp.conf** son instrucciones que pueden ser declaraciones o parámetros. Todas las instrucciones terminan con punto y coma. Las opciones se especifican en instrucciones del parámetro **options**. Los parámetros son diferentes de las declaraciones porque definen si se realiza una tarea, y la manera en que se realiza ésta, como el tiempo que se asigna un arrendamiento. Las declaraciones describen características de red como el rango de direcciones que se asignará o las redes accesibles. Consulte la tabla 35-1 para ver una lista de declaraciones y opciones de uso común.

Las declaraciones proporcionan información para el servidor DHCP o designan las acciones que habrán de realizarse. Por ejemplo, la declaración **range** se utiliza para especificar el rango de direcciones IP que se asignarán dinámicamente al host:

```
range 192.168.0.5 192.168.0.128;
```

Con parámetros, puede especificar la manera en que el servidor tratará a los clientes. Por ejemplo, la declaración **default-lease-time** establece el número de segundos que un

Declaraciones	Descripción
shared-network <i>nombre</i>	Indica si algunas subredes comparten la misma red física.
subnet <i>subnet-número-subred</i> <i>mask</i> <i>red</i>	Hace referencia a una subred completa de direcciones.
range [<i>bootp-dinámico</i>] <i>dirección-baja</i> [<i>dirección alta</i>];	Proporciona las direcciones IP mayores y menores asignadas de forma dinámica.
host <i>nombredehost</i>	Hace referencia a un host particular.
group	Le permite etiquetar un grupo de parámetros, declaraciones y después usar la etiqueta para aplicarlas a subredes y host.
allow unknown-clients; deny unknown-clients;	No asigna de forma dinámica direcciones a clientes desconocidos.
allow bootp; deny bootp;	Determina si responde a consultas bootp.
allow booting; deny booting;	Determina si responde a consultas de clientes.
Parámetros	
default-lease-time <i>tiempo</i> ;	Asigna la duración en segundos a un arrendamiento.
max-lease-time <i>tiempo</i> ;	Asigna el tamaño máximo del arrendamiento.
hardware <i>tipo-hardware</i> <i>dirección-hardware</i> ;	Especifica el tipo de hardware de red (Ethernet o token ring) y la dirección.
filename " <i>nombredearchivo</i> ";	Especifica el nombre de un archivo de arranque inicial.
server-name " <i>nombre</i> ";	Especifica el nombre del servidor en que está arrancando un cliente.
next-server <i>nombre-servidor</i> ;	Especifica el servidor que carga el archivo de arranque inicial determinado por el nombre de archivo.
fixed-address <i>dirección</i> [, <i>dirección ...</i>];	Asigna una dirección fija a un cliente.
get-lease-hostnames <i>marca</i> ;	Determina si se buscan y usan direcciones IP de clientes.
authoritative; not authoritative;	Niega solicitudes de direcciones no válidas.
server-identifier <i>hostname</i> ;	Especifica el servidor.
Opciones	
option <i>subnet-mask</i> <i>dirección-ip</i> ;	Especifica una máscara de subred del cliente.
option <i>routers</i> <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Especifica una lista de direcciones de IP enrutador en la subred del cliente.
option <i>domain-name-servers</i> <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Especifica listas de servidores de nombres de dominio utilizados por el cliente.

TABLA 35-1 Declaraciones, parámetros y opciones de DHCP (continúa...)

Opciones	Descripción
option log-servers <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Especifica listas de servidores de registro utilizados por el cliente.
option host-name <i>cadena</i> ;	Especifica el nombre de host del cliente.
option domain-name <i>cadena</i> ;	Especifica el nombre de dominio del cliente.
option broadcast-address <i>dirección-ip</i> ;	Especifica la dirección de difusión del cliente.
option nis-domain <i>cadena</i> ;	Especifica el dominio del servicio de información de red del cliente.
option nis-servers <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Especifica servidores NIS que usa el cliente.
option smtp-server <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Presenta una lista de servidores SMTP utilizados por el cliente.
option pop-server <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Presenta una lista de servidores POP utilizados por el cliente.
option nntp-server <i>dirección-ip</i> [, <i>dirección-ip...</i>];	Presenta una lista de servidores NNTP utilizados por el cliente.
option www-server <i>ip-address</i> [, <i>ip-address...</i>];	Presenta una lista de servidores Web utilizados por el cliente.

TABLA 35-1 Declaraciones, parámetros y opciones de DHCP (*continuación*)

arrendamiento se asigna a un cliente. La declaración **filename** especifica el archivo de arranque que usará el cliente. La declaración **server-name** informa al cliente desde cuál host está arrancando. La declaración **fixed-address** se utiliza para asignar una dirección IP estática a un cliente. Consulte la página Man de **dhcp.conf** para conocer una lista completa.

Las opciones proporcionan información a clientes que tal vez necesiten acceder a servicios de red, como el nombre de dominio de una red, los servidores de nombres de dominio que los clientes usan o la dirección de difusión. Consulte la página Man de **dhcp-options** para conocer una lista completa. Esta información se se proporciona mediante los parámetros **option**, como se muestra aquí:

```
option broadcast-address 192.168.0.255;
option domain-name-servers 192.168.0.1, 192.168.0.4;
option domain-name "mipista.com";
```

Su archivo **dhcpd.conf** suele comenzar con declaraciones, parámetros y opciones que se definen para su red a la que da servicio el servidor DHCP. En el siguiente ejemplo se proporciona información de enrutador (puerta de enlace), máscara de red, nombre de dominio y servidor DNS a clientes. Los parámetros adicionales definen los tiempos de arrendamiento predeterminados y máximos para direcciones IP asignadas de forma dinámica.

```
option routers 192.168.0.1;
option subnet-mask 255.255.255.0;
option domain-name "mipista.com";
```



```
option domain-name-servers 192.168.0.1;
default-lease-time 21600;
max-lease-time 43200;
```

Con las declaraciones de subred, host y grupo, se hace referencia a clientes en una red particular, clientes particulares o grupos diferentes de clientes a través de las redes. Dentro de estas declaraciones, se insertan parámetros, declaraciones y opciones que aplicarán sólo a esos clientes. Las declaraciones, los parámetros y las opciones definidos se encierran entre corchetes. Por ejemplo, para definir una declaración para un host particular, se utiliza la declaración **host**, como se muestra aquí:

```
host conejo {
    declaraciones, parámetros u opciones;
}
```

Se recolectan diferentes declaraciones de subred, global y host en grupos al utilizar la declaración **group**. En este caso, las declaraciones globales sólo se aplican a las redes y los hosts declarados dentro del grupo.

Direcciones IPv4 dinámicas para DHCP

Su servidor DHCP se configura para seleccionar direcciones IP de un rango determinado y asignarles diferentes clientes. Dada una situación donde se tienen muchos clientes que no siempre están conectados a la red, puede darles servicio de forma efectiva con un conjunto pequeño de direcciones IP, que sólo se asignan cuando es necesario. Con la declaración **range**, se especifica un rango de direcciones que se asignan de forma dinámica a clientes. La declaración toma dos argumentos, la primera y la última dirección del rango.

```
range 192.168.1.5 192.168.1.128;
```

Por ejemplo, si está configurando su propia red casera pequeña, utilizaría una dirección de red que comenzaría con 192.168. El rango especificaría las direcciones IP dentro de una red. Así que, para una red con la dirección 192.168.0.0, se coloca una declaración **range** junto con cualquier otra información que quiera dar a sus hosts de clientes. En el siguiente ejemplo, se asigna un rango de direcciones IP que va de 192.168.0.1 a 192.168.0.128 a los hosts de la red:

```
range 192.168.0.5 192.168.0.128;
```

También debe definir sus tiempos de arrendamiento, los predeterminados y los máximos:

```
default-lease-time 21600;
max-lease-time 43200;
```

En el caso de una red casera pequeña y simple, sólo necesita incluir la declaración **range** junto con cualquier opción global, como se muestra aquí. Si su servidor DHCP está administrando varias subredes, tendrá que utilizar las declaraciones **subnet**.

Para asignar direcciones dinámicas a una red, el servidor DHCP requerirá el mapeo de su topología de red. Esto significa que debe saber qué direcciones pertenecen a una red determinada. Aunque sólo utilice una red, necesitará especificar el espacio de direcciones de ésta. Aun si utiliza sólo una red, necesitará especificar la dirección a una red dada. Se define una red con la declaración **subnet**. Dentro de esta declaración, se especifica cualquier parámetro, declaración u opción que

habrá de usar dicha red. La declaración **subnet** informa al servidor DHCP las posibles direcciones IP incluidas en una subred. La dirección IP de red y la máscara de red determinan esto. En el siguiente ejemplo se define una red local con espacio de direcciones de 192.168.0.0 a 192.168.0.255. La declaración **range** permite que las direcciones se asignen de 192.168.0.5 a 192.168.0.128.

```
subnet 192.168.1.0 netmask 255.255.255.0 {  
    range 192.168.0.5 192.168.0.128;  
}
```

Las versiones de DHCP anteriores a 3.0 requerían que incluso se mapearan las interfaces de red conectadas a las que DHCP no daba servicio. Por tanto, cada interfaz de red debía tener una declaración **subnet** correspondiente. Los que no recibían servicio de DHCP no tenían un parámetro **not authoritative**, como se muestra aquí (DHCP no daba servicio a 192.168.2.0). En la versión 3.0 o posterior, DHCP sólo ignora las interfaces de red no mapeadas:

```
Subnet 192.168.2.0 netmask 255.255.255.0 {  
    not authoritative  
}
```

La implementación de un servidor DHCP muy simple para direcciones dinámicas se muestra en este archivo de ejemplo **dhcpd.conf**:

```
/etc/dhcpd.conf  
  
option routers 192.168.0.1;  
option subnet-mask 255.255.255.0;  
option domain-name "mipista.com";  
option domain-name-servers 192.168.0.1;  
  
subnet 192.168.1.0 netmask 255.255.255.0 {  
    range 192.168.0.5 192.168.0.128;  
    default-lease-time 21600;  
    max-lease-time 43200;  
}
```

Actualizaciones de DNS dinámico de DHCP

En el caso de redes que dan soporte a DNS, la asignación dinámica de direcciones IP necesita atender una restricción mayor: DHCP debe sincronizarse con un servidor DNS. Un servidor DNS asocia los nombres de host con una dirección IP; en cambio, en una asignación dinámica, el servidor DHCP asigna de forma aleatoria sus propias direcciones IP a diferentes hosts. Pueden o no ser las mismas direcciones IP que el servidor DNS espera asociar con un nombre de host. Se está desarrollando una solución a este problema, llamada DNS dinámico. Con éste, el servidor DHCP puede actualizar automáticamente el servidor DNS con la dirección IP que el servidor DHCP ha asignado a los diferentes hosts.

NOTA Como opción, si quiere sincronizar de forma estática sus servidores DHCP y DNS con direcciones fijas, se configura DHCP para asignar esas direcciones fijas a un host. Entonces se hace que el servidor DHCP realice una búsqueda DNS para obtener la dirección IP que debe asignarse, o se asigna manualmente la misma dirección IP en el archivo de configuración DHCP. La realización de una búsqueda de DNS tiene la ventaja de especificar la dirección IP en un lugar, el servidor DNS.

El servidor DHCP tiene la capacidad de actualizar de forma dinámica los archivos de configuración de zona del servidor DNS BIND. Se habilitan las actualizaciones dinámicas en un servidor DNS para un archivo de zona al especificar la opción **allow-update** en el archivo **named.conf**. Además, se recomienda que utilice las claves de firma TSIG para hacer referencia a los servidores BIND y DHCP y para autentificarlos. Actualmente, DHCP utiliza el esquema de actualización temporal de DNS para realizar las actualizaciones DNS dinámicas, al remplazar al esquema de actualización Ad-Hoc de DNS. Una versión acabada se implementará en lanzamientos DHCP futuros. Encuentre información más detallada acerca de DNS dinámico en la página Man **dhcpcd.conf**.

La habilitación del uso de una clave TSIG requiere la sincronización de configuraciones para sus servidores DHCP y DNS. Ambos tienen que configurarse para utilizar la misma clave para los mismos dominios. Primero necesita crear una clave de firma TSIG secreta al usar **dnssec-keygen**. En el servidor DNS, se colocan las declaraciones de clave TSIG y las entradas **allow-update** en el archivo **named.conf** del servidor, como se muestra en este ejemplo:

```
key miservidordhcp {
    algorithm HMAC-MD5;
    secret "ONQAfbBLnvWU9H8hRqg/WA==";
};

zone "mipista.com" {
    type master;
    file "mipista.com";
    allow-update {key miservidordhcp;};
};

zone "1.168.192.IN-ADDR.ARPA" {
    type master;
    file "192.168.0";
    allow-update {key miservidordhcp;};
};
```

En el servidor DHCP, se coloca una declaración de clave TSIG correspondiente y entradas **allow-update** en el archivo **dhcpcd.conf** del servidor, como se muestra en este ejemplo. La declaración **key** tiene la misma sintaxis que el servidor DNS. Después, se utilizan las expresiones **zone** de DHCP para especificar la dirección IP del dominio y la clave TSIG que habrá de usarse. Los nombres de dominio y las direcciones IP deben coincidir exactamente en los archivos de configuración para los servidores DNS y DHCP. A diferencia del archivo **named.conf**, no existen comillas en el nombre de dominio o la dirección IP en el archivo **dhcpcd.conf**. En éste, los nombres de dominio y las direcciones IP que se usan en la expresión **zone** también deben terminar con un punto, como en los archivos de zona DNS. La expresión **key** muestra la clave que habrá de usar. Aunque el servidor DHCP intentará determinar los servidores DNS que se actualizarán, se recomienda que los identifique explícitamente con una expresión primaria en la entrada **zone**.

```
key miservidordhcp {
    algorithm HMAC-MD5;
    secret "ONQAfbBLnvWU9H8hRqg/WA==";
};

zone mipista.com. {
    primary 192.168.0.1;
    key miservidordhcp;
};
```

#zona de dominio DNS que se actualizará
#dirección del servidor DNS
#clave de firma TSIG

```
zone 1.168.192.IN-ADDR.ARPA. {      #zona PTR de dominio que se actualizará
    primary 192.168.0.1;           #dirección del servidor DNS
    key miservidordhcp;            #clave de firma TSIG
};
```

Para generar el nombre de host plenamente calificado que se utilizará en una actualización DNS, el servidor DHCP suele utilizar su propio nombre de dominio y el nombre de host proporcionado por un cliente DHCP (consulte la página Man de **dhcpd.conf** para conocer excepciones). Si quiere asignar un nombre específico a un host, se utiliza la expresión **ddns-hostname** para especificar la sección de hardware del host. El nombre de dominio se especifica en la opción **domain-name**:

```
option domain-name "mipista.com"
```

La capacidad de actualización de DNS se habilita o deshabilita para todos los dominios con la expresión **ddns-update-style**. Está habilitada como opción predeterminada. Para deshabilitar las actualizaciones DNS para dominios particulares, se utiliza la expresión **ddns-updates**. También está habilitada como opción predeterminada.

Aquí se muestra una configuración simple de actualización de DNS para un servidor DHCP en el archivo **dhcpd.conf**:

```
/etc/dhcpd.conf
option routers 192.168.0.1;
option subnet-mask 255.255.255.0;
option domain-name "mipista.com";
option domain-name-servers 192.168.0.1;
key miservidordhcp {
    algorithm HMAC-MD5;
    secret "ONQAfbBLnvWU9H8hRqq/WA==";
};

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.0.5 192.168.0.128;
    default-lease-time 21600;
    max-lease-time 43200;
    zone mipista.com. {
        primary 192.168.0.1;
        key miservidordhcp;
    }
    zone 1.168.192.IN-ADDR.ARPA. {
        primary 192.168.0.1;
        key miservidordhcp;
    }
}
```

Subredes DHCP

Si está dividiendo su espacio de red en varias subredes, se utiliza un solo servidor DHCP para administraras. En ese caso, deberá tener una declaración **subnet** para cada subred. Si está configurando una red pequeña, se utiliza una dirección de red que comienza con 192.168. El rango especifica las direcciones IP posibles dentro de esa red; así que, para una red con la dirección 192.168.0.0, se crea una declaración **subnet** con la máscara de red 255.255.255.0. Dentro de esta

declaración, se coloca una declaración **range** junto con cualquier otra información que quiera dar a su cliente host. En el siguiente ejemplo, se asigna al host local de la red un rango de direcciones IP que va de 192.168.0.1 a 192.168.0.75:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.5 192.168.0.75;
}
```

Tal vez quiera especificar diferentes directivas para cada subred (por ejemplo, diferentes tiempos de arrendamiento). Cualquier entrada de una declaración **subnet** invalidará las configuraciones globales. Por ello, si ya tiene establecido el tiempo de arrendamiento global, una configuración en una declaración **subnet** lo invalidará para esa subred. En el siguiente ejemplo se establecen diferentes tiempos de arrendamiento para distintas subredes, además de diferentes asignaciones de dirección. Los tiempos de arrendamiento para la primera subred se toman de las configuraciones de tiempo de arrendamiento global, mientras la segunda subred define sus propios tiempos de arrendamiento:

```
default-lease-time 21600;
max-lease-time 43200;

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.0.5 192.168.0.75;
}
subnet 192.168.1.128 netmask 255.255.255.252 {
    range 192.168.0.129 192.168.0.215;
    default-lease-time 5600;
    max-lease-time 62000;
}
```

Si sus subredes son parte de la misma red física, debe informarlo al servidor al declarar las redes compartidas. Se hace esto al colocar declaraciones de subred dentro de una declaración **shared-network**, especificando el nombre de la red compartida. Éste puede ser cualquier nombre descriptivo, aunque es posible utilizar el nombre de dominio. Cualquier opción especificada con la declaración **shared-network** y las declaraciones de subred externas serán globales para esas subredes. En este ejemplo, las subredes son parte de la misma red física y por eso se colocan dentro de una declaración **shared-network**:

```
shared-network mipista.com
{
    default-lease-time 21600;
    max-lease-time 43200;
    subnet 192.168.1.0 netmask 255.255.255.0 {
        range 192.168.0.5 192.168.0.75;
    }
    subnet 192.168.1.128 netmask 255.255.255.252 {
        range 192.168.0.129 192.168.0.215;
        default-lease-time 56000;
        max-lease-time 62000;
    }
}
```

NOTA Dentro de una red, puede tener varias subredes que ejecutan servidores DHCP y otras que no. En ese caso, se utiliza el agente de retransmisión DHCP para permitir a los clientes DHCP de una subred sin un servidor DHCP usar uno que se ejecuta en otra subred. El agente de retransmisión de DHCP, **dhcrelay**, se administra con el comando **service**. Se configura en el archivo **/etc/sysconfig/dhcrelay**, donde se especifican las interfaces de red en que se recibirán solicitudes y en que se usará el servidor DHCP.

Direcciones fijas DHCP

En vez de utilizar un almacén de direcciones IP posibles para sus hosts, tal vez quiera darle a cada uno una dirección específica. Si usa el servidor DHCP todavía tendrá control sobre la dirección que se asignará a un host determinado. Sin embargo, para asignar una dirección a un host particular, necesita conocer la dirección de hardware de la tarjeta de interfaz de red (NIC, Network Interface Card) del host. En efecto, tiene que informar al servidor DHCP que debe asociar un dispositivo de conexión de red particular a una dirección IP especificada. Para hacerlo, el servidor DHCP necesita conocer el dispositivo de red al que está haciendo referencia. Se identifica un dispositivo de red por su dirección de hardware, conocida como dirección MAC. Para encontrar una dirección de hardware del cliente, se inicia sesión en el cliente y se utiliza el comando **ifconfig** para encontrar más información acerca de sus dispositivos de red. Para presentar una lista de todos los dispositivos de red, se utiliza la opción **-a**. Si conoce su nombre de dispositivo de red, se utiliza éste. En el siguiente ejemplo se desplegará una lista con toda la información acerca del primer dispositivo Ethernet, **eth0**:

```
ifconfig eth0
```

Esto mostrará una lista con la información en dispositivos de conexión de red de todos los clientes. La entrada con el término **HWaddr** (por lo general, la primera) desplegará la dirección MAC. Una vez que tiene la dirección MAC, se utiliza en el servidor DHCP para asignar una dirección IP específica a ese dispositivo.

En el archivo **dhcpd.conf**, se utiliza una declaración **host** para configurar la dirección fija de un cliente. Con la declaración **host**, se coloca una opción **hardware** que despliega una lista de su tipo de dispositivo de conexión de red y su dirección MAC. Después se utiliza el parámetro **fixed-address** para especificar la dirección IP que se asignará a ese dispositivo. En el siguiente ejemplo, al dispositivo de red del cliente con una dirección MAC de 08:00:2b:4c:29:32 se le asigna la dirección IP 192.168.0.2:

```
host conejo {
    option host-name "conejo.mipista.com"
    hardware ethernet 08:00:2b:4c:29:32;
    fixed-address 192.168.0.2;
}
```

También se utiliza el servidor DHCP para realizar una búsqueda DNS que permita obtener la dirección IP del host. Esto tiene la ventaja de que le permite administrar direcciones IP en un solo lugar, el servidor DNS. Por supuesto, esto requiere que el servidor DNS esté operando para que el servidor DHCP determine la dirección IP. Por ejemplo, una conexión de servidor proxy (que proporciona acceso Web directo) sólo necesita una dirección IP, no un nombre de host DNS, para operar. Si fallara el servidor DNS, con el ejemplo anterior todavía se asignaría una dirección IP al host, mientras que con el de la página siguiente no se asignaría.

```
host conejo {
    option host-name "conejo.mipista.com"
    hardware ethernet 08:00:2b:4c:29:32;
    fixed-address conejo.mipista.com;
}
```

También se utiliza la declaración **host** con el objetivo de definir información de red para una estación de trabajo o terminal sin disco. En este caso, se agrega el parámetro **filename** al especificar el archivo de arranque que usará esa estación de trabajo o terminal. Aquí la terminal llamada **miterminal** obtiene información de arranque del servidor **tortuga.mipista.com**:

```
host miterminal {
    option host-name "miterminal.mipista.com"
    filename "/boot/vmlinuz";
    hardware ethernet 08:00:2b:4c:29:32;
    server-name "tortuga.mipista.com";
}
```

Aquí se muestra un archivo **dhcpd.conf** de ejemplo de una implementación de un servidor DHCP simple para direcciones fijas. En la segunda declaración **host**, el DHCP realizará una búsqueda DNS para obtener la dirección IP de **conejo.mipista.com**:

```
/etc/dhcpd.conf
option routers 192.168.0.1;
option subnet-mask 255.255.255.0;
option domain-name "mipista.com";
option domain-name-servers 192.168.1.1;

subnet 192.168.1.0 netmask 255.255.255.0 {
    host tortuga {
        option host-name "tortuga.mipista.com"
        hardware ethernet 08:00:2b:4c:29:32;
        fixed-address 192.168.0.1;
    }
    host conejo {
        option host-name "conejo.mipista.com"
        hardware ethernet 00:80:AD:30:17:2A;
        fixed-address conejo.mipista.com;
    }
    host lagarto {
        option host-name "lagarto.mipista.com"
        hardware ethernet 00:70:2b:4b:29:14;
        fixed-address 192.168.0.3;
    }
}
```

También se coloca una declaración **host** dentro de una declaración **subred** para proporcionar información de un host particular en esa subred.

Un candidato común para una dirección fija es el servidor DNS de una red. Por lo general, querrá que el servidor DNS se ubique en la misma dirección IP, para que se acceda a éste directamente. Entonces el servidor DHCP proporciona esta dirección IP a sus clientes.

Linux proporciona varias herramientas para acceder a archivos en sistemas remotos conectados a una red. El sistema de archivos de red (NFS, Network File System) le permite conectarse y acceder directamente a recursos; por ejemplo, archivos o dispositivos como CD-ROM que residen en otra máquina. La nueva versión, NFS4, proporciona mayor seguridad, porque su firewall permite el acceso. El servicio de información de red (NIS, Network Information Service) mantiene archivos de configuración para todos los sistemas en una red.

Sistema de archivos de red: NFS y /etc/exports

NFS le permite montar un sistema de archivos en un equipo remoto como si fuera su sistema local propio. Luego accede directamente a cualquiera de los archivos en ese sistema de archivos remoto. Esto presenta la ventaja de que tiene diferentes sistemas en una red para acceder directamente a los mismos archivos, sin que cada uno tenga su propia copia. Sólo habrá una copia en un sistema de archivos remoto, a la que cada equipo tiene acceso. Encontrará más acerca de NFS en su sitio Web en nfs.sourceforge.net.

NOTA Ahora Mac OS X para Macintosh, que está basado en BSD Unix, da soporte a NFS para compartir archivos. Si desea acceder a sistemas de archivos e impresoras de Apple utilizando sistemas operativos de Apple más antiguos, se utiliza Netatalk, que implementa los protocolos de red IP AppleTalk y AppleShare clásicos en sistemas Unix y Linux. El sitio Web actual de Netatalk es netatalk.sourceforge.net, con vínculos para secciones FAQ y HOWTO.

NFSv4

La versión 4 de NFS es una nueva versión del protocolo NFS con características mejoradas, como mayor seguridad, confiabilidad y velocidad. Casi todos los comandos son los mismos, con algunos cambios. Por ejemplo, cuando se monta un sistema de archivo NFSv4, debe especificar el tipo de archivo **nfs4**. Además, debe montar el sistema de archivos NFSv4 completo en una ubicación. Para volver a montar direcciones específicas en diferentes ubicaciones (formato NFS), se utiliza la **opción de unión**.

```
/home/richlp
```

```
*(fsid=0, ro, sync)
```


La entrada anterior le permite montar el sistema de archivos en el directorio `/home/riclp` sin tener que especificarlo en la operación de montaje.

```
# mount -t nfs4 conejo.mipista.com:/ /home/daniel/proyectos
```

Daemons de NFS

NFS opera sobre una red TCP/IP. El equipo remoto que almacena el sistema de archivos lo pone a disposición de otros equipos en la red. Lo hace al exportar el sistema de archivos, que requiere la creación de entradas en un archivo de configuración NFS llamado `/etc/exports`, además de ejecutar varios daemons que dan soporte al acceso por parte de otros sistemas. Éstos incluyen **rpc.mountd**, **rpc.nfsd** y **rpc.portmapper**. Los archivos `/etc/hosts.allow` y `/etc/hosts.deny` controlan el acceso a su servidor NFS. Los daemons NFS se listan aquí:

- **rpc.nfsd** Recibe solicitudes NFS de sistemas remotos y los traduce a peticiones para el sistema local.
- **rpc.mountd** Realiza operaciones de montaje y desmontaje solicitadas.
- **rpc.portmapper** Asigna solicitudes remotas al daemon NFS apropiado.
- **rpc.rquotad** Proporciona administración de cuota de disco.
- **rpc.statd** Proporciona servicios de bloqueo cuando un host remoto reinicia.
- **rpc.lockd** Maneja recuperación de bloqueo para sistemas que han fallado.

NOTA Es recomendable utilizar NFS sólo en una red segura local. Si se usa sobre Internet, NFS abre su sistema a acceso inseguro.

Inicio y detención de NFS

La secuencia de comandos de servicio **nfs** inicia los daemons **portmapper**, **nfsd**, **mountd** y **rquotad**. Para habilitar el bloqueo NFS, se utiliza la secuencia de comandos **nfslock**. Esto iniciará los daemons **statd** y **lockd**. El bloqueo de NFS proporciona una mejor recuperación de operaciones interrumpidas que ocurren debido a fallas de sistema en hosts remotos. Se utiliza **chkconfig** en Red Hat, Fedora, SUSE y distribuciones similares, o **services-admin** o **sysv-rc-conf** en Debian, Ubuntu y otras distribuciones para que NFS inicie de manera automática.

Para ver si realmente se está ejecutando NFS, se utiliza el comando **rpcinfo** con la opción **-p**. Debe ver entradas para **mountd** y **nfs**. Si no, NFS no se está ejecutando.

Configuración de NFS: `/etc/exports`

Una entrada en el archivo `/etc/exports` especifica el sistema de archivos que se exportará y el host en la red al que accede. En el caso del sistema de archivos, inserte su *punto de montaje*, el directorio en que se montó en el sistema host. Esto es seguido por una lista de hosts que acceden a este sistema de archivos junto con opciones para controlar ese acceso. Es posible que a cada host le siga una lista separada por comas de opciones de exportación colocadas dentro de un conjunto de paréntesis. Por ejemplo, tal vez quiera darle a un host acceso de sólo lectura y a otro acceso de lectura y escritura. Si las opciones tienen un símbolo ***** antes, se aplican a cualquier host. Una lista de opciones se proporciona en la tabla 36-1. Aquí se muestra el formato de una entrada en el archivo `/etc/exports`:

```
nombrederuta-directorio designacion-host(opciones)
```

Opciones generales	Descripción
secure	Requiere que las solicitudes se originen en puertos seguros, menores a 1024. Está habilitado, como opción predeterminada.
insecure	Desactiva la opción secure .
ro	Permite acceso de sólo lectura. Es la opción predeterminada.
rw	Permite acceso de lectura/escritura.
sync	Escribe todo cuando se pide. Es la opción predeterminada.
async	Escribe todo cuando el servidor está listo.
no_wdelay	Escribe inmediatamente, sin revisar si está relacionado lo escrito.
wdelay	Revisa si lo escrito está relacionado; en ese caso, espera para realizarlas juntas. Degrada el rendimiento. Es la opción predeterminada.
hide	Esconde automáticamente un directorio exportado que es subdirectorio de otro directorio exportado. Es necesario montar explícitamente el subdirectorio para acceder a él. Si se monta el directorio principal no se permite el acceso. Es la opción predeterminada.
no_hide	No oculta un directorio exportado que es el subdirectorio de otro directorio exportado (lo opuesto a hide). Sólo trabaja con hosts únicos y no es confiable.
subtree_check	Revisa directorios principales en un sistema de archivos para validar un subdirectorio exportado. Es la opción predeterminada.
no_subtree_check	No revisa los directorios principales en un sistema de archivos para validar un subdirectorio exportado.
insecure_locks	No requiere autenticación de solicitudes de bloqueo. Se utiliza para versiones NFS.
Asignación de ID de usuario	Descripción
all_squash	Asigna todos los UID y GID al usuario anónimo. Es útil para directorios FTP públicos, directorios de noticias, etc., exportados por NFS.
no_all_squash	Lo contrario a la opción all_squash . Es la configuración predeterminada.
root_squash	Asigna solicitudes de un usuario root remoto al UID/GID anónimo. Es la opción predeterminada.
no_root_squash	Desactiva la fragmentación de root. Permite el usuario root acceder como el root remoto.
anonuid anongid	Establece de manera explícita el UID y GID de la cuenta anónima utilizada para las opciones all_squash y root_squash . Las opciones predeterminadas son nobody y nogroup .

TABLA 36-1 Las opciones de /etc/exports

Entradas de host NFS

Puede tener varias entradas de host en el mismo directorio, cada uno con acceso a ese directorio:

```
nombrederuta-directorio host(opciones) host(opciones) host(opciones)
```

Se tiene una gran flexibilidad cuando se especifican los hosts. En el caso de los hosts dentro de su dominio, sólo se utiliza el nombre de host, mientras que para los externos, se debe usar un nombre de dominio plenamente calificado. También puede usar sólo la dirección IP del host. En lugar de un solo host, se hace referencia a los hosts dentro de un dominio específico, permitiendo el acceso de una red completa. Una forma simple de hacer esto es mediante el uso del `*` para el segmento de host, seguido por el nombre de dominio de la red, como `*.mipista.com` para todos los hosts de la red **mipista.com**. En lugar de nombres de dominio se utilizan las direcciones IP con un formato CNDR donde se especifica la máscara de red para indicar un rango de direcciones IP. También se utiliza un nombre de grupo de red NIS para hacer referencia a una colección de hosts. Se pone un signo `@` antes del nombre de grupo de red NIS.

```
directorio      host(opciones)
directorio      *(opciones)
directorio      *.dominio(opciones)
directorio      192.168.1.0/255.255.255.0(opciones)
directorio      @grupodered (opciones)
```

Opciones de NFS

Las opciones en `/etc/exports` operan como permisos para controlar el acceso a directorios exportados. El acceso de sólo lectura se establece con la opción `ro`, y el acceso de lectura/escritura con `rw`. Las opciones `sync` y `async` especifican si una operación de escritura se realiza de inmediato (`sync`) o cuando el servidor está listo para manejarlo (`async`). Como opción predeterminada, se revisa si las solicitudes de escritura están relacionadas; en ese caso, se escriben juntas (`wdelay`). Esto degrada el rendimiento. Se invalida esta opción predeterminada con `no_wdelay` y se ejecutan las escrituras cuando se pide. Si se exportan dos directorios, donde uno es el subdirectorio de otro, no se accede al subdirectorio a menos que esté montado explícitamente (`hide`). En otras palabras, el hecho de montar el directorio principal no hace que el subdirectorio quede disponible. Éste permanece oculto hasta que también se monta. Se invalida esta restricción con la opción `no_hide` (aunque esto causa problemas con algunos sistemas de archivos). Si un directorio exportado es realmente un subdirectorio de un sistema de archivos más grande, también se revisan sus directorios principales para asegurarse de que el subdirectorio es el directorio válido (`subtree_check`). Esta opción funciona bien con sistemas de archivos de sólo lectura pero causa problemas en sistemas de archivos con escritura habilitada, donde cambian los nombres de archivo y directorios. Se cancela la revisión con la opción `no_subtree_check`.

Acceso en el nivel usuario de NFS

Junto con las opciones generales, también hay opciones que se aplican al acceso en el nivel usuario. Como medida de seguridad, el servidor NFS trata al usuario root del cliente como un usuario anónimo. A esto se le conoce como *fragmentación* del usuario. En el caso del usuario root del cliente, la fragmentación evita que el cliente trate de aparecer como usuario root del servidor NFS. Si quiere que un usuario root de un cliente particular tenga control en el nivel de root sobre el servidor NFS, se especifica la opción `no_root_squash`. Para evitar que cualquier usuario de cliente trate de aparecer como usuario en el servidor NFS, se clasifican como usuarios anónimos (la opción `all_squash`). Estos usuarios anónimos sólo tienen acceso a directorios y archivos que son parte del grupo anónimo.

Por lo general, si un usuario de un sistema cliente tiene una cuenta de usuario en el servidor NFS, ese usuario puede montarse y acceder a sus archivos en el servidor NFS. Sin embargo, NFS requiere que el ID de usuario sea el mismo en ambos sistemas. Si no es así, se considera que son dos usuarios. Para evitar este problema, se utiliza un servicio NIS, que mantiene la información de ID de usuario en un solo lugar, el archivo de contraseña NIS (consulte la siguiente sección para conocer más información sobre NIS).

Ejemplo de /etc/exports NFS

Aquí se muestran ejemplos de entradas en el archivo `/etc/exports`. A todos los hosts se da acceso de sólo lectura al sistema de archivos montado en el directorio `/pub`, un nombre común utilizado para acceso público. Sin embargo, se trata a los usuarios como anónimos (`all_squash`). Se da acceso de lectura y escritura al equipo `lagarto.mipista.com` para el sistema de archivos montado en el directorio `/home/foodstuff`. En la siguiente entrada se permite el acceso de `conejo.mipista.com` al CD-ROM del servidor NFS, utilizado el acceso de sólo lectura. La última entrada permite a cualquiera el acceso seguro a `/home/richlp`.

```
/etc/exports
/pub                *(ro,insecure,all_squash,sync)
/home/foodstuff     lagarto.mipista.com(rw,sync)
/media/cdrom        conejo.mipista.com(ro,sync)
/home/richlp        *(secure,sync)
```

Aplicación de cambios

Cada vez que su sistema inicia el servidor NFS (por lo general, cuando el sistema inicia), se lee el archivo `/etc/exports` y se exportan los directorios especificados. Cuando se exporta un directorio, se hace una entrada para éste en el archivo `/var/lib/nfs/xtab`. Este es el archivo que lee NFS y se utiliza para realizar las exportaciones reales. Las entradas se leen de `/etc/exports` y se crean las entradas correspondientes en `/var/lib/nfs/xtab`. El archivo `xtab` mantiene una lista de exportaciones reales.

Si quiere exportar de inmediato entradas agregadas al archivo `/etc/exports`, sin reiniciar, utilice el comando `exportfs` con la opción `-a`. Es útil agregar la opción `-v` para desplegar las acciones que NFS toma. Se utilizan las opciones para efectuar cualquier cambio que haga al archivo `/etc/exports`.

```
exportfs -a -v
```

Si después hace cambios al archivo `/etc/exports`, se utiliza la opción `-r` para volver a exportar sus entradas. La opción `-r` volverá a sincronizar el archivo `/var/lib/nfs/xtab` con las entradas `/etc/exports`, al eliminar cualquier otra exportación o cualquier opción diferente.

```
exportfs -r -v
```

Para exportar entradas añadidas y volver a exportar los cambios, se combinan las opciones `-r` y `-a`.

```
exportfs -r -a -v
```

Exportación manual de sistemas de archivos

También se utiliza el comando `exportfs` para exportar sistemas de archivos de forma manual, en lugar de usar entradas para éstos en el archivo `/etc/exports`. Las entradas de exportación se agregarán directamente al archivo `/var/lib/nfs/xtab`. Con la opción `-o`, se incluyen permisos y

después pueden incluirse el host y el sistema de archivos que habrá de exportarse. El sistema de archivos y el host se separan por dos puntos. Por ejemplo, para exportar de forma manual el directorio **/home/misproyectos** a **golf.mipista.com** con los permisos **ro** e **insecure**, se utiliza lo siguiente:

```
exportfs -o rw,insecure golf.mipista.com:/home/misproyectos
```

Se utiliza **exportfs** para dejar de exportar un directorio que ya se ha exportado, ya sea manualmente o mediante el archivo **/etc/exports**. Sólo utilice la opción **-u** con el host y el directorio exportado. La entrada para la exportación se eliminará del archivo **/var/lib/nfs/xtab**. En el siguiente ejemplo se dejará de exportar el directorio **/home/cosasdecomida** que fue exportado a **lagarto.mipista.com**:

```
exportfs -u lagarto.mipista.com:/home/cosasdecomida
```

Seguridad de directorios y archivos NFS con listas de acceso de NFS4

Con NFS4 se configura una lista de control de acceso (ACL) para directorios y archivos particulares. Se utiliza la herramienta ACL de NFS4 para administrar estas listas (el paquete **nfs4-acl-tools**). Entre las herramientas de ACL del sistema de archivos de NFS4 se incluyen **nfs4_getfacl**, **nfs4_setfacl** y **nfs4_editfacl**. Revise la página Man de cada una para conocer opciones y ejemplos. **nfs4_getfacl** mostrará los controles de acceso para un archivo o directorio específico. **nfs4_setfacl** creará controles de acceso para un directorio o archivo y **nfs4_editfacl** le permitirá cambiarlos. **nfs4_editfacl** sólo invoca **nfs_setfacl** con la opción **-e**. Cuando se editan los controles de acceso, se le coloca en un editor donde se hacen los cambios. Para establecer controles de acceso se leen de un archivo, de la entrada estándar o de la lista de entradas de control en la línea de comandos.

Los controles de acceso de archivos y directorios son más refinados que los permisos estándar descritos en el capítulo 28. Las entradas ACL siguen la sintaxis descrita con detalle en la página Man **nfs4_acl**. Una entrada ACL comienza con un tipo de entrada, como una entrada **accept** o **deny** (**A** o **D**); seguido por una marca de ACL, que especifica capacidad de grupo o herencia y después el principal al que se aplica ACL; por último, la lista de opciones de acceso, como **r** para lectura o **w** para escritura. El principal suele ser un usuario URL al que se le permite o niega el acceso. También se especifican los grupos, pero necesita establecer la marca de grupo **g**. Los URL especiales **OWNER@**, **GROUP@** y **EVERYONE@** corresponden al propietario, grupo y otro acceso utilizado en permisos estándar. En el siguiente ejemplo se proporciona acceso completo al propietario, pero se da acceso de sólo lectura y ejecución al usuario **jorge@conejo.com**. Se niega el acceso de escritura y ejecución al grupo.

```
A::OWNER::rwadtTnNcCy
A::jorge@conejo.com:rxtncy
D:g:GROUP::waxtc
```

Además de los permisos de lectura, escritura y ejecución (**r**, **w**, **x**), las listas ACL también proporcionan atributos de lectura (**t**, **n**) y escritura (**T**, **N**), además de acceso de lectura (**c**) y escritura (**C**) de ACL. La sincronización de lectura y escritura de NFS se habilita con la opción **y**. La opción **d** proporciona la capacidad de eliminar archivos y directorios, y la opción **D**, la de eliminar subdirectorios. La opción **a** le permite adjuntar datos y crear subdirectorios. Tenga en cuenta que **rtncy** son todas opciones de lectura, en tanto que **wadDTNC** son opciones de escritura, mientras **x** permanece como opción de ejecución. Necesitará **y** para acceso sincronizado. Sobre todo, la opción **C** es muy poderosa porque permite que el usuario cambie los controles de acceso (**c** minúscula permite sólo lectura de los controles de acceso).

Control de acceso a servidores NFS

Puede utilizar varios métodos para controlar el acceso a sus servidores NFS, como utilizar `hosts.allow` y `hosts.deny` para permitir o negar el acceso, además de utilizar su firewall para interceptar el acceso.

/etc/hosts.allow y /etc/hosts.deny

Los archivos `/etc/hosts.allow` y `/etc/hosts.deny` se utilizan para restringir el acceso a servicios proporcionados por su servidor para hosts en su red o en Internet (si se accede a Internet). Por ejemplo, se utiliza el archivo `hosts.allow` para permitir el acceso a ciertos hosts en su servidor FTP. Las entradas del archivo `hosts.deny` niegan explícitamente el acceso a ciertos hosts. En el caso de NFS, se proporciona el mismo tipo de seguridad al controlar el acceso a daemons NFS específicos.

NOTA Se aseguran más sus transmisiones NFS al hacer que operen sobre TCP en lugar de UDP. Se utiliza la opción `tcp` para montar sus sistemas de archivos NFS (UDP es la opción predeterminada). Sin embargo, el rendimiento se degrada para NFS cuando utiliza TCP.

Servicio portmapper

La primera línea de defensa es controlar el acceso al servicio portmapper. Éste indica a los hosts dónde se encuentran los servicios NFS en el sistema. Al restringir el acceso no se permite que un host remoto localice NFS. Para un nivel fuerte de seguridad, debe negar el acceso a todos los hosts, excepto los que se permiten explícitamente. En el archivo `hosts.deny`, se coloca la siguiente entrada, que niega el acceso a todos los hosts, como opción predeterminada. ALL es una palabra clave especial que denota a todos los hosts.

```
portmap:ALL
```

En el archivo `hosts.allow`, puede insertar los hosts de su red, o cualquier otro al que quiera permitir el acceso a su servidor NFS. Una vez más, se especifica el servicio portmapper y después la lista de direcciones IP de los hosts a los que se les permite el acceso. Se incluyen las direcciones IP específicas, o un rango de red al utilizar una máscara de red. En el siguiente ejemplo sólo se permite el acceso a hosts en la red local, 192.168.0.0 y al host 10.0.0.43. Se separan las direcciones con comas:

```
portmap: 192.168.0.0/255.255.255.0, 10.0.0.43
```

Otros servicios, como NIS, también utilizan portmapper. Si cierra todo el acceso a portmapper en `hosts.deny`, también deberá permitir el acceso a servicios NIS en `hosts.allow`, si los ejecuta. Entre éstos se incluyen `ybind` y `yserver`. Además, tal vez tenga que agregar entradas para comandos remotos como `ruptime` y `rusers`, si da soporte a éstos.

También es recomendable agregar el mismo nivel de control para servicios NFS específicos. En el archivo `hosts.deny`, se agregan entradas para cada servicio, como se muestra aquí:

```
mountd:ALL
rquotad:ALL
statd:ALL
lockd:ALL
```

Luego, en el archivo `hosts.allow`, se agregan entradas para cada servicio:

```
mountd: 192.168.0.0/255.255.255.0, 10.0.0.43
rquotad: 192.168.0.0/255.255.255.0, 10.0.0.43
statd: 192.168.0.0/255.255.255.0, 10.0.0.43
lockd: 192.168.0.0/255.255.255.0, 10.0.0.43
```

Reglas de Netfilter

Puede controlar mucho más el acceso al utilizar Netfilter para revisar transmisiones de ciertos hosts en los puertos usados por servicios NFS. (Consulte el capítulo 20 para conocer una explicación de Netfilter.) portmapper usa el puerto 111 y nfsd el 2049. Netfilter es útil si tiene una red privada con una conexión de Internet y quiere protegerla. Por lo general, un dispositivo de red específico, como una tarjeta Ethernet, está dedicado a la conexión de Internet. En los siguientes ejemplos se supone que el dispositivo **eth1** está conectado a Internet. Se negará cualquier paquete al que trate de acceder en el puerto 111 o 2049.

```
iptables -A INPUT -i eth1 -p 111 -j DENY
iptables -A INPUT -i eth1 -p 2049 -j DENY
```

Para habilitar NFS para su red local, tendrá que permitir fragmentos de paquetes. Suponiendo que **eth0** es un dispositivo utilizado para la red local, se puede usar el siguiente ejemplo:

```
iptables -A INPUT -i eth0 -f -j ACCEPT
```

Montaje de sistemas de archivos NFS: clientes NFS

Una vez que NFS pone los directorios a disposición de diferentes hosts, estos últimos pueden montar los directorios en sus propios sistemas y acceder a éstos. El host necesita la capacidad de operar como un cliente NFS. Todos los kernels de Linux actuales tienen capacidad de cliente NFS integrada. Esto significa que cualquier cliente NFS puede montar un directorio NFS remoto al que se tiene acceso mediante una simple operación de montaje.

Montaje automático de NFS: /etc/fstab

Se monta un directorio NFS mediante una entrada en el archivo **/etc/fstab** o un comando **mount** explícito. Montará automáticamente su sistema de archivos NFS al colocar entradas para éstos en el archivo **/etc/fstab**. Una entrada NFS en el archivo **/etc/fstab** tiene un tipo de montaje de NFS. Un nombre de sistema de archivos NFS consta del nombre de host del equipo donde está ubicado, seguido por el nombre de ruta del directorio donde está montado. Los dos se separan por dos puntos. Por ejemplo, **conejo.trek.com:/home/proyecto** especifica un sistema de archivos montado en **/home/proyecto** en el equipo **conejo.pista.com**. El formato de una entrada NFS en el archivo **/etc/fstab** se presenta a continuación. Observe que el tipo de archivo es **nfs**.

```
host:directorio-remoto    directorio-local    nfs    opciones    0    0
```

También se incluyen varias opciones de montaje específicas NFS con su entrada NFS. Puede especificar el tamaño de los datagramas enviados de ida y vuelta y la cantidad de tiempo que su computadora esperará una respuesta de su sistema host. También se especifica si un sistema de archivos tiene un montaje fuerte o débil. En el caso de un sistema de archivos *de montaje fuerte*, su equipo intenta continuamente hacer contacto si por alguna razón el sistema remoto deja de responder. Un sistema de archivos *de montaje débil*, después de un intervalo específico, deja de tratar de hacer contacto y envía un mensaje de error. La opción predeterminada es un montaje fuerte. Un sistema que hace un intento de montaje fuerte que sigue fallando dejará de responder a la entrada del usuario mientras intenta archivar continuamente el montaje. Por esta razón, tal vez sean preferibles los montajes suaves, porque simplemente dejarán de intentar un montaje que falla continuamente. En la tabla 36-2 y en las páginas Man para **mount** se incluye una lista de estas opciones de cliente NFS. Ellas difieren de las opciones de servidor de NFS indicadas antes.

Opción	Descripción
rsize=<i>n</i>	El número de bytes que NFS utiliza cuando lee archivos de un servidor NFS. La opción predeterminada es 1024 bytes. Un tamaño de 8192 mejora enormemente el rendimiento.
wsiz=<i>n</i>	El número de bytes que NFS utiliza cuando escribe archivos en un servidor NFS. La opción predeterminada es 1024 bytes. Un tamaño de 8192 mejora enormemente el rendimiento.
timeo=<i>n</i>	El valor en décimas de segundo antes de enviar la primera retransmisión después de una pausa. El valor predeterminado es siete décimas.
retry=<i>n</i>	El número de minutos que transcurre para reintentar una operación de montaje NFS antes de dejar de intentar. La opción predeterminada es 10 000 minutos (una semana).
retrans=<i>n</i>	El número de retransmisiones o pausas menores en una operación de montaje NFS antes de una pausa mayor (la opción predeterminada es 3). En ese momento, la conexión se cancela o se despliega un mensaje “el servidor no responde”.
soft	Utiliza un montaje débil para el sistema.
hard	Utiliza un montaje fuerte para el sistema. Es la opción predeterminada.
intr	Permite que NFS interrumpa la operación de archivo y regrese al programa que llama. La opción predeterminada es no permitir que se interrumpan las operaciones de archivos.
bg	Si se agota el tiempo de espera en los primeros intentos de montaje, sigue intentando el montaje en segundo plano. La opción predeterminada es que no se pase a segundo plano tras la falla.
tcp	Mounts the NFS file system using the TCP protocol, instead of the default UDP protocol.

TABLA 36-2 Opciones de montaje de NFS

A continuación se presenta un ejemplo de entrada NFS. El sistema remoto es **conejo.mipista.com** y el sistema de archivos está montado en **/home/proyectos**. Este sistema de archivos se monta en el sistema local como el directorio **/home/daniel/proyectos**. El directorio **/home/daniel/proyectos** debe estar ya creado en el sistema local. El tipo de sistema es NFS y la opción **timeo** especifica que el sistema local espera 20 décimas de segundo (dos segundos) para una respuesta. El montaje es suave y NFS tiene la capacidad de interrumpirlo.

```
conejo.mipista.com:/home/proyectos /home/daniel/proyectos nfs
soft,intr,timeo=20
```

Montaje manual de NFS: mount

También se utiliza el comando **mount** con la opción **-t nfs** para montar un sistema de archivos NFS de forma explícita. Para un sistema de archivos NFSv4 se utiliza **-t nfs4**. Para montar la entrada anterior de forma explícita se utiliza el siguiente comando:

```
# mount -t nfs -o soft, intr, timeo=20 \
    conejo.mipista.com:/home/proyectos /home/daniel/proyectos
```


Por supuesto, se desmonta un directorio NFS con el comando **umount**. Puede especificar el punto de montaje local o el host remoto y el directorio, como se muestra aquí:

```
umount /home/daniel/proyectos
umount conejo.mipista.com:/home/proyectos
```

Montaje de NFS bajo pedido: autofs

También puede montar los sistemas de archivos NFS al utilizar el servicio de montaje automático, autofs. Esto requiere una configuración agregada en la parte del cliente. El servicio autofs montará un sistema de archivos sólo cuando intente acceder a éste. Una operación de cambio de directorio (**cd**) a un directorio especificado accionará la operación de montaje, que monta el sistema de archivos remoto en ese momento.

El servicio autofs se configura al utilizar un archivo maestro para mostrar archivos de mapa, que a su vez muestran los sistemas de archivos que se montan. El archivo **/etc/auto.master** es el maestro autofs. El archivo maestro presentará una lista de los nombres de ruta raíz donde los sistemas de archivos se montan junto con un archivo de mapa para cada uno de esos nombres de ruta. Entonces el archivo de mapa mostrará una clave (subdirectorio), opciones de montaje y los sistemas de archivos que se montan en ese directorio de nombre de ruta raíz. En algunas distribuciones, el directorio **/auto** ya está implementado como el nombre de ruta raíz para sistemas de archivos que se montan automáticamente. Puede agregar sus propios sistemas de archivos en **/etc/auto.master** junto con sus propios archivos de mapa, si así lo desea. Encontrará que el archivo **/etc/auto.master** contiene la siguiente entrada para el directorio **/auto**, que presenta a **auto.misc** como archivo de mapa:

```
/auto auto.misc --timeout 60
```

Después del archivo de mapa, se agregan opciones, como se muestra en el ejemplo anterior. La opción **timeout** especifica el número de segundos de inactividad que habrá de esperar antes de desmontar automáticamente.

En el archivo de mapa, se incluyen la clave, las opciones de montaje y los sistemas de archivos que se montan. La clave será el subdirectorio en el sistema local donde los sistemas de archivos se montan. Por ejemplo, para montar el directorio **/home/proyectos** del host **conejo.mipista.com** en el directorio **/auto/proyectos**, se utiliza la siguiente entrada:

```
proyectos soft, intr, timeo=20 conejo.mipista.com:/home/proyectos
```

También se crea una nueva entrada en el archivo maestro para un sistema de archivos NFS, como se muestra aquí:

```
/misproyectos auto.misproyectos --timeout 60
```

Luego se crea un archivo **/etc/auto.misproyectos** y se colocan entradas en éste para los montajes del sistema de archivos NFS, como en el siguiente ejemplo:

```
daniel soft,intr,rw conejo.mipista.com:/home/proyectos
nuevojuego soft,intr,ro lagarto.mipista.com:/home/superjuego
```

Servicio de información de red: NIS

En redes que dan soporte a NFS, los mismos sistemas comparten muchos recursos y dispositivos. Por lo general, cada sistema necesita sus propios archivos de configuración para cada dispositivo o recurso. Los cambios incluyen la actualización individual de cada sistema. Sin embargo, NFS

proporciona un servicio especial llamado servicio de información de red (NIS, Network Information Service) que mantiene esos archivos de configuración para toda la red. Para hacer cambios, sólo necesita actualizar los archivos NIS. Éste funciona con la información necesaria para la mayor parte de las tareas administrativas, como las relacionadas con usuarios, el acceso de red o dispositivos. Por ejemplo, puede mantener información de usuario y contraseña con un servicio NIS, al sólo tener que actualizar los archivos de contraseña NIS.

NOTA *NIS+ es una forma más avanzada de NIS que proporciona soporte a cifrado y autenticación. Sin embargo, es mucho más difícil de administrar.*

NIS fue desarrollado por Sun Microsystems y se le conoció originalmente como Yellow Pages (YP) de Sun. Los archivos de NIS se mantienen en un servidor NIS (a los servidores NIS aún se les llega a conocer como servidores YP). Los sistemas individuales de una red utilizan clientes NIS para hacer solicitudes del servidor NIS. El servidor NIS mantiene su información en archivos de base de datos especiales llamados *mapas*. Existen versiones de Linux para ambos, clientes y servidores NIS. Los clientes NIS se conectan de forma sencilla a cualquier red al utilizar NIS.

NOTA *En lugar de NIS, muchas redes ahora usan LDAP para administrar información de usuario y autenticación.*

El cliente NIS se instala como parte de la instalación inicial en casi todas las distribuciones de Linux. Los programas de cliente NIS son `ypbind` (el daemon de cliente NIS), `ypwhich`, `ypcat`, `yppoll`, `ypmatch`, `ypasswd` y `ypset`. Cada uno tiene su propia página Man con detalles de su uso. Los programas de servidor NIS son `ypserv` (el servidor NIS), `ypinit`, `ypasswdd`, `yppush`, `ypxfr` y `netgroup` (cada uno también tiene páginas Man).

Servidores NIS

Tiene mucha flexibilidad para la configuración de servidores NIS. Si tiene una red pequeña, tal vez sólo necesite un dominio NIS, para el cual tendría un servidor NIS. En el caso de redes más grandes, se divide su red en varios dominios NIS, cada uno con su propio servidor. Aunque sólo tenga un dominio, tal vez sólo quiera varios servidores esclavos NIS. Para un dominio NIS, puede tener un servidor NIS maestro y varios servidores esclavos NIS. Los segundos actúan como respaldos, en caso de que el servidor maestro falle. Un servidor esclavo sólo contiene copias de archivos de configuración del servidor maestro NIS.

La configuración de un servidor NIS incluye varios pasos:

1. Defina el nombre de dominio NIS para el cual trabajará el servidor NIS.
2. Inicie el daemon `ypserv`.
3. En el archivo `/var/yp/Makefile`, establezca cualquier opción del servidor NIS y especifique los archivos de configuración que habrán de administrarse.
4. Utilice `/usr/lib/ypinit` para crear las versiones de NIS para los archivos de configuración.

Definición del dominio NIS

Primero tiene que definir un nombre de dominio NIS. Puede definir el dominio NIS siempre que se inicia su sistema, al definir la variable `NIS_DOMAIN` en el archivo `/etc/sysconfig/network`.

Para esta variable, se asigna el nombre que quiere darle a su dominio NIS. En el siguiente ejemplo se define el dominio NIS llamado **mistortugas.nis**:

```
NIS_DOMAIN=mistortugas.nis
```

Cuando configura por primera vez el servidor, tal vez quiera definir su nombre de dominio NIS sin tener que reiniciar su sistema. Se hace esto con el comando **domainname**, como se muestra aquí:

```
domainname mistortugas.nis
```

Las opciones de servidor NIS se mantienen en el archivo **/etc/ypserv.conf**. Revise la página Man de dicho archivo para conocer más detalles.

Configuración de opciones de servidor NIS

Después, se edita el archivo **/var/yp/Makefile** para seleccionar los archivos de configuración que el servidor NIS mantendrá y se establece cualquier opción de servidor NIS. Ya estarán configurados las opciones estándar y los archivos de configuración de uso común.

Primero aparecen las opciones de servidor NIS. La opción **NOPUSH** se establecerá en true, lo que indica que no son servidores esclavo NIS. Si quiere configurar cualquier servidor esclavo NIS para este dominio, tendrá que establecer esta opción en false:

```
NOPUSH = true
```

A los ID de usuario y grupo mínimos se les asigna un valor de 500. Éstos se establecen al usar las variables **MINUID** y **MINGID**:

```
MINUID=500
MINGID=500
```

Casi todas las distribuciones utilizan contraseñas y archivos de grupo seguros para cifrar contraseñas y grupos; las configuraciones **MERGE_PASSWD** y **MERGE_GROUP** serán true. NIS combinará información de contraseña seguro de su archivo de contraseña.

```
MERGE_PASSWD=true
MERGE_GROUP=true
```

Los directorios donde NIS encontrará la contraseña y otros archivos de configuración entonces se definen al utilizar las variables **YPSRCDIR** y **YPPWDDIR**. Por lo general, el directorio **/etc/** almacena sus archivos de configuración:

```
YPSRCDIR = /etc
YPPWDDIR = /etc
```

Después, se incluyen los archivos de configuración que NIS administra. Aquí, encontrará entradas como **PASSWD** para contraseña, **GROUP** para sus grupos y **PRINTCAP** para sus impresoras. Aquí se muestra un ejemplo de entradas:

```
GROUP      = $(YPPWDDIR)/group
PASSWD     = $(YPPWDDIR)/passwd
SHADOW     = $(YPPWDDIR)/shadow
GSHADOW    = $(YPPWDDIR)/gshadow
ALIASES    = /etc/aliases
ETHERS     = $(YPSRCDIR)/ethers      # direccion ethernet (para rarpd)
BOOTPARAMS = $(YPSRCDIR)/bootparams # para arrancar Sun boxes (bootparamd)
```

```
HOSTS      = $(YPSRCDIR)/hosts
NETWORKS   = $(YPSRCDIR)/networks
PRINTCAP   = $(YPSRCDIR)/printcap
PROTOCOLS  = $(YPSRCDIR)/protocols
```

Especificación de archivos compartidos

Los archivos reales que se comparten en la red se incluyen en la entrada **all:**, que sigue después de la lista de archivos de configuración. Sólo algunos de los archivos definidos aparecen como compartidos, los que están en la primera línea después de **all:**. Las líneas restantes se vuelven comentarios de forma automática (con un signo # antes). Se agregan archivos al eliminar el signo # o eliminar sus entradas en la primera línea.

```
all:  passwd group hosts rpc services netid protocols mail \
      # netgrp shadow publickey networks ethers bootparams printcap \
      # amd.home auto.master auto.home auto.local passwd.adjust \
      # timezone locale netmasks
```

Asegúrese de no tocar el resto del archivo Makefile.

Creación de la base de datos de NIS

Luego se inserta el comando **ypinit** con la opción **-m** para crear la base de datos de NIS que consta de archivos de configuración NIS. Se detectará su servidor NIS y se le pedirá que se inserten los nombres de cualquier servidor NIS esclavo utilizado en este dominio NIS. Si hay alguno, insértelo. Cuando termine, oprima CTRL-D. Entonces se crean los archivos de base de datos de NIS. El comando **ypinit** se ubica en el directorio **/usr/lib/yp**.

```
ypinit -m
```

Para un servidor esclavo NIS, se utiliza:

```
ypinit -s hostmaestro
```

Si recibe el siguiente error, lo más probable es que su servidor NIS no haya estado en ejecución. Asegúrese de iniciar **ypserv** antes de ejecutar **ypinit**.

```
failed to send 'clear' to local ypserv: RPC: Program not registredUpdating
```

Si después necesita actualizar sus archivos de servidor NIS, se cambia al directorio **/var/yp** y se envía el comando **make**.

```
cd /var/yp
make
```

Control del acceso

El archivo **/var/yp/securenets** habilita el acceso de los hosts en su servidor NIS. Se hace referencia a los hosts por su red o de forma individual. Las entradas constan de una máscara de subred y una dirección IP. Por ejemplo, se da acceso a todos los hosts en una red local con la siguiente entrada:

```
255.255.255.0 192.168.1.0
```

Para hosts individuales, se utiliza la máscara 255.255.255.255 o sólo el término “host”, como se muestra aquí:

```
host 192.168.1.4
```

En `/etc/ypserv.conf` se controla la manera en que diferentes hosts acceden a los datos comprimidos de NIS.

Grupos de red

Se utiliza NIS para configurar grupos de red, que le permiten crear grupos en el nivel red de usuarios. Mientras los grupos normales se crean de forma local en hosts separados, un grupo de red NIS se utiliza para servicios en toda la red. Por ejemplo, se utilizan los grupos de red NIS para controlar el acceso a sistemas de archivos NFS. Los grupos de red se definen en el archivo `/etc/netgroup`. Las entradas constan de un nombre de grupo seguido por identificadores de miembro que constan de tres segmentos: el host, el usuario y el dominio NIS:

```
grupo (host, usuario, dominio-NIS) (host, usuario, dominio-NIS) ...
```

Por ejemplo, en el dominio NIS **mistortugas.nis**, para definir un grupo llamado **misproyectos** que consta del usuario **carlos** en el host **conejo** y el usuario **jorge** en el host **lagarto.mipista.com**, se utiliza lo siguiente:

```
misproyectos (conejo, carlos, mistortugas.nis) \
              (lagarto.mipista.com, jorge, mistortugas.nis)
```

Un segmento en blanco coincidirá con cualquier valor. En la siguiente entrada se incluyen todos los usuarios del host **conejo**.

```
nuevojuego (conejo,,mistortugas.ni)
```

Si la manera en que se utiliza un grupo no requiere el usuario ni el segmento de host, se elimina uno u otro al utilizar un guión (-). En el siguiente ejemplo se genera un grupo de red que sólo contiene nombres de hosts, sin nombres de usuario:

```
misservidores (conejo,-,) (tortuga.mipista.com,-,)
```

Entonces se hace referencia a diferentes grupos de red en varios archivos de configuración al incluir un signo @ antes del nombre de grupo de red, como se muestra aquí:

```
@nuevojuego
```

Cientes NIS

Para que un host utilice NIS en su red, primero necesita especificar su nombre de dominio NIS en ese host. Además, sus clientes NIS necesitan saber el nombre de su servidor NIS. Si instala Linux en otra red que ya está ejecutando NIS, tal vez ya haya insertado esta información durante el proceso de instalación. Se especifica su nombre de dominio NIS y servidor en el archivo `/etc/yp.conf`.

Acceso al servidor

Entonces cada host cliente NIS de su red tiene que ejecutar el cliente NIS `ypbind` para acceder al servidor. En el archivo `/etc/yp.conf` del cliente, necesita especificar el servidor NIS que utilizará. En la siguiente entrada se hace referencia al servidor NIS en 192.168.1.1:

```
ypserver 192.168.1.1
```

Como opción, se especifica el nombre de dominio NIS y el servidor que usa:

```
domain midominio.nis server nombreservidor
```

La siguiente entrada para dominio y servidor estaría en `/etc/yp.conf` para el dominio NIS **mitortuga.nis** utilizando el servidor **tortuga.mipista.com**:

```
domain mistortugas.nis server tortuga.mipista.com
```

Se utiliza **ypcat** para mostrar cualquiera de los archivos de configuración NIS. El comando **ypwhich** desplegará el nombre del servidor NIS que usa su cliente. **ypmatch** se utiliza para encontrar una entrada particular en un archivo de configuración.

```
ypmatch cecilia passwd
```

Los usuarios cambian sus contraseñas en el archivo **passwd** de NIS con el comando **yppasswd**; trabaja igual que el comando **passwd**. También tendrá que tener en ejecución el daemon **yppasswdd**.

Especificación de los archivos de configuración con **nsswitch.conf**

Para asegurar que el cliente accede al servidor NIS para un archivo de configuración particular, debe especificar **nisplus** en la entrada de ese archivo en `/etc/nsswitch.conf`. La opción **nisplus** alude a la versión 3 de NIS. La opción **nis** alude a la antigua versión 2 de NIS. El archivo `/etc/nsswitch.conf` especifica dónde debe buscar un host para cierto tipo de información. Por ejemplo, la siguiente entrada indica que revise primero los archivos de configuración local (**files**) y luego el servidor NIS (**nisplus**) en busca de datos de contraseña:

```
passwd:    files nisplus
```

La designación **files** indica que se utilicen primero los archivos propios del sistema, los que están en el host local. **nis** indica que se busquen entradas en los archivos NIS, al acceder al servidor NIS. **nisplus** dice que se utilicen los archivos NIS+ que se mantienen por el servidor NIS+. **dns** indica que se realicen búsquedas DNS; sólo se utiliza en archivos como **hosts** que contienen nombres de host. Éstas son algunas entradas estándar:

```
passwd:    files nisplus
shadow:    files nisplus
group:     files nisplus

hosts:     files nisplus dns
bootparams: nisplus [NOTFOUND=return] files

ethers:    files
netmasks: files
networks:  files
protocols: files nisplus
rpc:       files
services:  files nisplus
netgroup:  files nisplus
publickey: nisplus
automount: files nisplus
aliases:   files nisplus
```


Sistemas de archivos de red distribuidos

En el caso de sistemas muy grandes distribuidos como los clústeres de Linux, Linux también da soporte a sistemas de archivos de red, como Coda, Intermezzo, Red Hat Global File System (GFS y GFS 2) y Parallel Virtual File System (PVFS2). Estos sistemas se basan en el concepto de NFS, además de técnicas RAID, para crear un sistema de archivos implementado en varios hosts de una red grande, distribuyendo el mismo sistema de archivos entre diferentes hosts en un nivel muy bajo (consulte la tabla 37-1). Puede considerar que se trata de un tipo de conjunto RAID implementado a través de hosts de red en lugar de un solo sistema. En vez de que cada sistema host dependa de sus propios sistemas de archivos en su propio disco duro, todos comparten el mismo sistema de archivos distribuido que utiliza discos duros agrupados en diferentes servidores distribuidos. Esto proporciona un uso mucho más eficiente de almacenamiento disponible para los hosts, además de brindar una administración mucho más centralizada del sistema de archivos que usa.

Parallel Virtual File System (PVFS)

Parallel Virtual File System (PVFS) implementa un sistema de archivos de red distribuido al utilizar un servidor que administra el sistema de archivos en diferentes servidores de E/S. Los servidores de administración mantienen la información de sistema de archivos, incluidos los permisos, la estructura de directorio y la información de metadatos. Las solicitudes para acceder a un archivo son remitidas por un cliente del servidor de administración. Éste establece entonces una conexión entre el cliente y los servidores E/S que almacenan los datos del archivo solicitado y la operación de acceso como las tareas de lectura y escritura que se llevan a cabo directamente entre el cliente y los servidores E/S. PVFS se implementa de forma transparente al utilizar un módulo de kernel para utilizar el sistema de archivos virtual del kernel. Entonces cualquier cliente podrá montar PVFS como cualquier sistema de archivos. En una implementación PVFS, el sistema de archivos se organiza en franjas de datos, de forma similar a un conjunto RAID, pero las franjas se distribuyen en hosts diferentes de la red. Se accede a los archivos a través de un conjunto de franjas que se distribuyen a través de esta red.

Sitio Web	Nombre
fedoraproject.org/wiki/Tools/GFS	Recursos y vínculos GFS de Fedora.
pvfs.org	Parallel Virtual File System, PVFS2 (fuente abierta)
sourceware.org/gfs	Global File System (Fedora y versiones comerciales)
coda.cs.cmu.edu ftp.coda.cs.cmu.edu/pub/coda/linux	Coda File system, acceso móvil desconectado (experimental)
inter-mezzo.org	Intermezzo (fuente abierta)
clusterfs.com	Lustre

TABLA 37-1 Sistemas de archivos distribuidos

Una nueva versión de PVFS, conocida como PVFS2, está disponible en pvfs.org. Puede descargar el código fuente de ahí. PVFS es un proyecto conjunto del Laboratorio de Investigación Paralela de Arquitectura de la Universidad de Clemson y la División de Matemáticas y Cómputo en el Argonne National Laboratory.

El servidor de administración PVFS utiliza dos archivos de configuración, **pvfs2-fs.conf** y **pvfs2-server.conf**. Se crean estos archivos al utilizar una secuencia de comandos de configuración, **pvfs2-genconfig**. La secuencia de comandos le pedirá su información de configuración como protocolo, puerto, archivo de registro y directorio de almacenamiento. Para configurar una red de clúster, asegúrese de insertar el host que operará como servidores E/S.

El servidor de administración es **pvfs2-server**. Para administrar el servidor se utiliza la secuencia de comandos de servicio **pvfs2-server**.

En servidores E/S, se ejecuta la secuencia de comandos **pvfs2-server** para crear el espacio de almacenamiento y después se inicia el servidor. Los servidores E/S utilizan el archivo de configuración **/etc/pvfs2-fs.conf**. Se utiliza la secuencia de comandos **pvfs2-server** para activar el servidor automáticamente.

En los clientes que utilizarán el sistema PVFS, necesita instalar herramientas de información de configuración y acceso de sistema de archivos. Cada cliente necesitará un daemon PVFS, **pvfs2-client** y su módulo de soporte y biblioteca. Los sistemas de archivos montan configuraciones que se almacenan en **/etc/pvf2stab**, aunque también se colocan entradas directamente en el archivo **/etc/fstab**. Para montar un sistema de archivos PVFS, también se utiliza el tipo **pvfs2** en el comando **mount** con la opción **-t**. PVFS2 proporciona su propio conjunto de comandos como **pvfs2-ping**, **pvfs2-cp** y **pvfs2-ls**, para desplegar y acceder a archivos en su sistema de archivos.

El cliente puede acceder a un sistema de archivos PVFS2 a través de un módulo de kernel para proporcionar acceso al sistema de archivos de Linux, o con la interfaz MPI-IO, que requiere recompilación con el uso de la biblioteca **libpvfs2** pero proporciona mejor rendimiento para aplicaciones paralelas. El soporte paralelo en PVFS2 se implementa con la interfaz de transición de mensajes (MPI) y tiene soporte en ROMIO (www-unix.mcs.anl.gov/romio) y MPICH2 (www-unix.mcs.anl.gov/mpi/mpich2) disponible en la División MCS del Argonne National Laboratory.

NOTA Para que las aplicaciones aprovechen por completo PVFS, deben ser compatibles con PVFS, que dividirán en franjas sus datos de archivos para mejor uso en sistemas PVFS.

Coda

Coda fue desarrollado por la Universidad Carnegie Mellon como un proyecto experimental, aunque está disponible de forma gratuita. Algunas de sus características incluyen soporte a computadoras móviles, controles de acceso y adaptación al ancho de banda. Obtenga más información acerca de Coda en coda.cs.cmu.edu y descargue paquetes de algunas versiones en ftp.coda.cs.cmu.edu/pud/coda/linux.

Al utilizar un módulo de kernel para interactuar con el sistema de archivos virtual, los archivos Coda distribuidos se acceden desde un directorio de Coda en un cliente, por lo general `/coda`. Coda mantendrá una caché de archivos a los que se accede de forma frecuente en el cliente para mejorar la eficiencia. Un administrador de caché llamado `venus` mantiene la caché y maneja todos los sistemas de archivos solicitados. El uso de una caché permite una operación desconectada en un archivo, dejando que los usuarios trabajen en un archivo de forma local y después lo actualicen con los servidores principales. Una operación desconectada funciona bien en equipos móviles, donde las laptops pueden desconectarse de la red por periodos. Las bases de datos correspondientes a los archivos usados con frecuencia por usuarios, conocidos como grupos, también se mantienen en el servidor para facilitar las actualizaciones.

Para configurar clientes, necesitará el paquete `coda-debug-client`. Utilice la secuencia de comandos `venus-setup` para configurar el cliente y después inicie Coda con el daemon `venus`. En el caso del servidor, debe instalar el paquete `coda-dbug-server` y ejecutar `venus-setup` para configurar su servidor.

Red Hat Global File System (GFS y GFS 2)

Red Hat ha lanzado Global File System (GFS) como un sistema de archivos de red de fuente abierta disponible de forma gratuita. La versión original de GFS se incluía con Fedora 4 y 5. A partir de Fedora Core 6, se incluye la nueva versión de GFS, GFS 2, que utiliza un conjunto de herramientas similares de configuración y administración, además del soporte a kernel nativo. En lugar de varios paquetes aparentemente inconexos, GFS 2 se implementa con sólo tres: `gfs2-utils`, `cman` y `lvm-cluster`. El soporte de kernel nativo para GFS 2 proporciona muchas operaciones más en el nivel de kernel.

Un sistema de archivos de red distribuido genera el concepto básico de NFS, además de técnicas RAID para crear un sistema de archivos implementado en varios hosts a través de una red grande, distribuyendo en realidad el mismo sistema de archivos entre diferentes hosts en un nivel muy bajo. Se puede considerar como un tipo de conjunto RAID implementado a través de hosts de red en lugar de un solo sistema. Es decir, en lugar de que cada sistema host dependa de sus propios sistemas de archivos en su propio disco duro, todos comparten el mismo sistema de archivos distribuido que utiliza discos duros agrupados en diferentes servidores distribuidos. Esto proporciona un uso mucho más eficiente de almacenamiento disponible para los hosts y proporciona una administración mucho más centralizada del sistema de archivos que usa. GFS se ejecuta directamente conectado a SAN (Storage Area Network, red de área de almacenamiento) o mediante almacenamiento de GNBD (Global Network Block Device, dispositivo global de bloque de red) conectado a través de una LAN. El mejor rendimiento se obtiene de una conexión SAN, mientras un formato GNBD se implementa de forma sencilla al utilizar el almacenamiento en sistemas conectados a LAN (Ethernet). Al igual que con dispositivos RAID, las opciones de espejo, recuperación de fallas y redundancia ayudan a proteger y recuperar datos.

GFS separa la implementación física del formato lógico. Un GFS aparece como un conjunto de volúmenes lógicos en un dispositivo lógico uniforme que se monta de forma sencilla en cualquier

Sitio Web	Nombre
redhat.com/software/rha/gfs	Global File System (versión comercial de Red Hat)
redhat.com/docs/manuals/csgfs/	Manuales de Global File System de Red Hat (implementación de Red Hat Enterprise)
sourceware.org/cluster	El sitio Web de Cluster Project, que incluye vínculos con documentación de GFS
<code>/etc/cluster.conf</code>	Archivo de configuración de cluster (css) GFS

directorio de su sistema de archivos Linux. Cluster Logical Volume Manager (CLVM) crea y administra los volúmenes lógicos; se trata de un LVM con clúster habilitado. Físicamente, el sistema de archivos se conforma con diferentes recursos de almacenamiento, conocidos como nodos de clúster, distribuidos a través de la red. El administrador maneja estos nodos, que proporcionan la función de espejo necesaria y expansión de almacenamiento. Si un nodo falla, GFS cierra un sistema hasta que se recupere éste. Es necesario planear la configuración de un GFS. Tiene que determinar con anticipación diferentes configuraciones como el número y los nombres de sus sistemas globales de archivo, los nodos que podrán montar los sistemas de archivos, los métodos de contención y las particiones y los discos que habrán de usarse.

Para conocer información más detallada, visite el sitio Cluster Project Page en sourceware.org/cluster. Se presentan los paquetes utilizados en GFS (Cluster Components–Old) y GFS 2 (Cluster Components–New). Aquí encontrará vínculos con documentación como las preguntas más frecuentes sobre división en clústeres. La guía Red Hat GFS Administrators Guide es útil pero depende de la fecha. La guía se encuentra en la página de documentación de Red Hat ubicada en redhat.com. (Tenga en cuenta que GFS utiliza ahora volúmenes lógicos en lugar del almacén para configurar volúmenes físicos.)

Paquetes de GFS 2 (Fedora Core 6 y posteriores)

El GFS original, GFS 1, utilizaba varios paquetes separados para servidores de clúster y herramientas de administración, que, al parecer, carecían de relación, a no ser por sus nombres. Con GFS 2, estos paquetes se combinaron en paquetes **cman** y **gfs2-tools**. Aquí encontrará herramientas como fence, cman cluster manager, dlm locking control y ccs cluster configuration. La configuración de cluster tiene soporte en Cluster Configuration System, ccs. Fencing se utiliza para aislar recursos fallidos. Tiene soporte en el servidor fence. El soporte a clústeres LVM se ubica en un paquete separado como **lvm2-cluster** (Fedora) o **clvm** (Debian).

Para ejecutar un clúster, necesita un administrador de clúster y un mecanismo de bloqueo. **cman** con Distributed Lock Manager (**dlm**) implementa administración de clúster y bloqueo. **cman** administra conexiones entre dispositivos y servicios clúster, al utilizar **dlm** para proporcionar bloqueo. El mecanismo de bloqueo **dlm** opera como un daemon con librerías de soporte.

Todos estos servicios se invocan con la secuencia de comandos **cman**, que revisa el archivo `/etc/clúster.conf` para configuración de clúster.

Secuencias de comandos de servicio GFS 2

Para iniciar el sistema de archivos GFS, se ejecuta la secuencia de comandos **cman** para iniciar el daemon necesario e implementar su configuración. La secuencia de comandos **cman** ejecutará **ccsd** para iniciar la detección de configuración, **fenced** para soporte de contención, **dlm_control** para administración de clúster de bloqueo **dlm** y **cman** para administración de clúster. La secuencia de comandos revisará cualquier configuración en el archivo `/etc/sysconfig/clúster`. Se utiliza la

secuencia de comandos **gfs2** para montar sus sistemas de archivos GFS 2. Para apagar el servicio del sistema de archivos GFS, se utiliza la secuencia de comandos **cman** con la opción **stop**.

```
service cman start
service gfs2 start
```

La secuencia de comandos del servicio **gfs2** montará los sistemas de archivos de GFS en las ubicaciones especificadas en el archivo **/etc/fstab**. Necesitará entradas para todos los sistemas de archivos GFS que quiera montar en **/etc/fstab**. La opción **stop** desmontará los sistemas de archivos. Se utiliza **cman_tool** para agregar un nodo a un clúster o para eliminarlo.

Implementación de un sistema de archivos GFS 2

Para configurar un sistema de archivos de GFS 2, primero necesita crear dispositivos de clúster al utilizar los volúmenes físicos y organizarlos en volúmenes lógicos. Se utiliza CLVM (Clustering Logical Volume Manager) para configurar volúmenes lógicos de particiones físicas (en el pasado se utilizaba un administrador de volúmenes llamado almacén para hacer esto). Luego se instalan directamente los sistemas de archivos GFS en estos volúmenes lógicos. CLVM opera como LVM, que utiliza los mismos comandos. Trabaja sobre una red distribuida y requiere que el servidor **clvmd** se ejecute.

Después se configura su sistema con Cluster Configuration System. Cree un archivo **/etc/cluster.conf** y establezca su configuración. La configuración incluirá información como nodos usados, los métodos de contención y el método de bloqueo utilizado. Consulte la página Man **cluster.conf** para conocer detalles de la configuración. Pruebe la configuración con la herramienta **ccs_test**.

```
ccs_test mygfs
```

Utilice entonces **ccs_tool** para crear archivos de configuración **cluster.ccs**, **fence.ccs** y **node.ccs**. Estos archivos se organizan en un archivo CCS que se coloca en cada dispositivo de nodo y de clúster.

En cada nodo, inicie la configuración **ccsd**, el servidor de contención **fenced** y el método de bloqueo que quiere utilizar, como **dlm**. Revise las páginas Man respectivas para conocer más detalles sobre los servidores de bloqueo. Se inician los servidores con las secuencias de comandos de servicio, como ya se indicó.

Para crear nuevos sistemas de archivos en los dispositivos de clúster, se utiliza el comando **gfs2_mkfs** y se montan con la opción **-t gfs2**. El siguiente comando crea un sistema de archivos GFS en **/dev/gv0/mgfs** y después se monta en el directorio **/mygfs**. En el caso de **gfs2_mkfs**, la opción **-t** indica la tabla de bloqueo y **-p** especifica el protocolo de bloqueo. La opción **-j** especifica el número de diarios.

```
gfs2_mkfs -t micluster:mygfs -p lock_dlm -j 2 /dev/vg0/mgfs
mount -t gfs /dev/vg0/mgfs /gfs1
```

Para que la secuencia de comandos de servicio **gfs** monte el sistema de archivos, necesita colocar una entrada para ésta en el archivo **/etc/fstab**. Si no quiere que el sistema de archivos se monte automáticamente, agregue la opción **noauto**.

```
/dev/vg0/mgfs /mygfs gfs2 noauto,defaults 0 0
```

Con las entradas **/etc/fstab** de GFS, puede utilizar la secuencia de comandos **gfs2** para montar el sistema de archivos GFS.

```
service gfs start
```

Comandos	Descripción
ccs	Secuencia de comandos de servicio CCS para iniciar el servidor Cluster Configuration Service
ccs_tool	Herramienta de actualización de configuración CCS
ccs_test	CHerramienta de diagnóstico CCS para probar archivos de configuración de CCS
ccsd	Daemon que se ejecuta en nodos para proporcionar datos de configuración CCS para software de clúster
clvmd	Daemon Cluster Logical Volume Manager, necesario para crear y administrar dispositivos de clúster LVM, además de una secuencia de comandos de servicio para iniciar clvmd
cman	La secuencia de comandos del Cluster Manager, cman , utiliza dlm para bloquear (cman se ejecuta directamente como un módulo de kernel)
cman_tool	Administra nodos de clúster, requiere cman
dlm	Distributed Lock Manager, implementado como un módulo de kernel, se invoca por la secuencia de comandos cman
fence	Revisión general de contención
fenced	Daemon de contención, además de una secuencia de comandos de servicio para iniciar el daemon fenced
fence_tool	Administra el daemon fenced
fence_node	Invoca un agente de contención
agentes de contención	Varios agentes de contención disponibles para diferentes tipos de conexiones; consulte la página fence
fence_manual	Agente de contención para interacción manual
fence_ack_manual	Interfaz de usuario fence_manual
gfs2	Secuencia de comandos de servicio GFS 2 para montar sistemas de archivos GFS 2, además de una revisión general de la página Man
gfs2_mount	Es invocado por mount; utiliza la opción -t gfs2
gfs2_fsck	Revisor de sistema de archivos GFS 2
gfs2_grow	Aumenta el sistema de archivos GFS 2
gfs2_jadd	Agrega un diario al sistema de archivos GFS 2
mkfs.gfs2	Crea un sistema de archivos GFS 2
gfs2_quota	Manipula las cuotas de disco de GFS 2
gfs2_tool	Administra un sistema de archivos GFS 2
getfacl	Obtiene permisos ACL para un archivo o directorio
setfacl	Establece control de acceso (ACL) para un directorio o archivo
rmanager	Resource Group Manager, administra servicios de usuario

TABLA 37-2 Herramientas, daemons y secuencias de comandos de servicio de GFS

Herramientas de GFS

GFS tiene varios comandos en diferentes categorías, como los que trabajan con contención, como **fence_tool**; **gulfm_tool** para administrar bloque **gulfm**; y los que se utilizan para configuración, como **cman_tool**. Los comandos de GFS para administrar sistemas de archivos GFS aparecen en la tabla 37-2. Revise sus páginas Man respectivas para conocer descripciones detalladas.

Operaciones de sistema de archivos GFS

Varios comandos GFS administran el sistema de archivos, como **gfs2_mount** para montar sistemas de archivos, **gfs2_mkfs** para hacer un sistema de archivos GFS, **gfs2_fsck** para revisar y reparar, y **gfs2_grow** para expandir un sistema de archivos. Revise las páginas Man respectivas para conocer descripciones detalladas.

NOTA Para GFS 1, se utilizan los mismos nombres para las herramientas de GFS sin el número 2; es decir, **gfs** en vez de **gfs2**.

Para montar un sistema de archivos GFS, se utiliza el comando **mount** al especificar **gfs2** como tipo de montaje, como en

```
mount -t gfs2 /dev/vg0/mgfs /migfs
```

Esto invocará la herramienta **gfs2_mount** para realizar la operación de montaje. También están disponibles varias opciones de montaje específicas de GFS, que se especifican con la opción **-o**, como **lockproto** para especificar un protocolo de bloqueo diferente y **acl** para habilitar el soporte ACL.

Para revisar el estado de un sistema de archivos, se utiliza **gfs2_fsck**. Esta herramienta opera de forma muy parecida a **fsck**, que revisa sistemas corruptos e intenta repararlos. Primero debe desmontar el sistema de archivos antes de usar **gfs2_fsck** en éste.

Si agrega espacio disponible al dispositivo en que reside un sistema de archivos GFS, se utiliza **gfs2_grow** para expandir el sistema de archivos al espacio disponible. Se ejecuta únicamente en un nodo para expandir un clúster completo. Si quiere crear diarios, primero tiene que agregar archivos de diario con la herramienta **gfs2_jadd**. **gfs2_grow** sólo se ejecuta en un sistema de archivos GFS montado.

Los archivos de diario de GFS se instalan en espacio fuera del sistema de archivos GFS, pero no en el mismo dispositivo. Después de crear un sistema de archivos GFS, se ejecuta **gfs2_add** para agregar archivos de diario para éste. Si está expandiendo un sistema de archivos GFS actual, primero necesita ejecutar **gfs2_add**. Como **gfs2_grow**, **gfs2_add** sólo se ejecuta en sistemas de archivos montados. Con el comando **setfacl** se establecen permisos para archivos y directorios.

Como se observó antes, para crear un sistema de archivos GFS se utiliza el comando **gfs2_mkfs**. La opción **-t** especifica la tabla de bloqueo, la opción **-j** indica el número de diarios que se creará y la opción **-p** especifica el protocolo de bloqueo que se usará.

Resource Group Manager, **rgmanager**, proporciona una interfaz de línea de comandos para administrar servicios de usuario y recursos en un sistema de archivos GFS, que le permite realizar tareas administrativas básicas como configurar cuotas de usuario, apagar el sistema (**clushutdown**) y obtener estadísticas del uso de GFS (**clustat**). La herramienta administrativa principal es **clusterfs**. Las opciones se establecen en el archivo **/etc/sysconfig/cluster**. Se inicia **rgmanager** con la secuencia de comandos **rgmanager**. Esto inicia el daemon **clurgmgrd**, que proporciona acceso al sistema GFS.

GFS también da soporte a controles de acceso. Se restringe el acceso por usuarios o grupos a ciertos archivos o directorios, al especificar permisos de lectura o escritura. Con el comando **setfacl** se establecen permisos para archivos y directorios. Se utiliza la opción **-m** para modificar

un permiso ACL y **-x** para eliminarlo. **getfacl** obtiene los permisos actuales para archivos o directorios. En el siguiente ejemplo se permite acceso de lectura del usuario **daniel** a **miarchivo**.

```
setfacl -m u:daniel:r miarchivo
```

GFS 1

GFS 1, la primera versión de GFS, implementa el sistema de archivos global con una serie de paquetes separados, al parecer no relacionados. Para utilizar GFS 1, tiene que instalar varios paquetes diferentes que incluyen las herramientas GFS 1, el método de bloqueo que quiere usar y las herramientas de configuración. Las herramientas GFS 1 y los métodos de bloqueo también tienen varios módulos de kernel y paquetes de encabezado correspondientes. Consulte **sourceware.org/cluster** bajo el encabezado “Cluster Components-Old” para conocer una lista de herramientas de GFS 1. Había diferentes paquetes de módulos kernel para diferentes tipos de kernel: i586, i686, SMP y Xen.

Dónde obtener distribuciones de Linux

Las distribuciones de Linux proporcionan sistemas de Linux muy estables y a nivel profesional junto con interfaces GUI KDE y GNOME, herramientas de configuración de sistema fáciles de usar y flexibles, un conjunto extenso de servidores de Internet, una variedad de diferentes aplicaciones multimedia y miles de aplicaciones de todo tipo de Linux. Se encuentra información reciente acerca de diferentes distribuciones y sus sitios Web respectivos. De ahí también se descargan imágenes de DVD o CD que se ejecutan y utilizan para instalar una distribución particular. Varias distribuciones también proporcionan Live-CDs que se utilizan para ejecutar Linux con una sola unidad de CD-ROM, sin tener que instalar Linux en su disco duro.

Casi todo el software de distribuciones principales está disponible para descargas de depósitos soportados por la distribución. Las imágenes de instalación de disco también están disponibles, ya sea instalaciones pequeñas sólo de escritorio o instalaciones de servidor más grandes. Por lo general se utiliza un Live-CD, si se proporciona, para instalar Linux. La estrategia de distribución recae en discos de instalación con una colección seleccionada de software que se actualiza y mejora después de una colección muy grande de software en el depósito de la distribución. Esto significa que la colección de software de una instalación inicial es relativamente pequeña. El software en el depósito también se actualiza continuamente, así que cualquier instalación tendrá que pasar por una extensa actualización del depósito.

Asegúrese primero de leer la guía de instalación para la distribución que está instalando. Ésta se proporciona en línea en el sitio Web de la distribución y se puede leer con cualquier explorador.

Varias distribuciones de Linux y sus sitios Web aparecen en la tabla A-1. En casi todos los casos sólo necesita buscar el nombre de un servicio como Google para encontrar el sitio. El sitio DistroWatch proporciona una lista detallada y una descripción de casi todas las distribuciones.

www.distrowatch.com

Nombre	Sitio
Debian	debian.org
Fedora	fedoraproject.org
openSUSE	opensuse.com
Red Hat	redhat.com
Ubuntu	ubuntu.com
Gentoo	gentoo.org
CentoOS	centos.org
MEPIS	mepis.org
Slackware	slackware.com
Turbolinux	turbolinux.com

TABLA A-1 Distribuciones populares de Linux

Símbolos

- ! (comando de historial)
 - editar en la shell C, 61
 - hacer referencia a sucesos con, 41-42, 44, 59-60
 - usar en la shell Z, 63
 - usar, 40-41
- !, operador, usar con el dispositivo eth0, 383
- !~, operador, usar con la shell TCSH, 82
- !=, operador, usar con la shell TCSH, 82
- # (signo de número), símbolo
 - como indicador de comandos, 21,36
 - usar con etiquetas de comandos de servicio, 413-414
 - usar con la directiva ServerName, 452
 - usar con servicios xinetd, 409-410
- \$ (signo de pesos)
 - anteceder el comando csh con, 57
 - aparece antes de las shell, 36
 - como indicador de comandos, 99
 - función de, 21,68
 - representar palabras con, 60
 - usar con la acción automática de completar de BASH, 39
- \$, operador, usar con las variables de shell, 68
- % (por ciento), símbolo
 - anteceder los números de trabajo con, 54
 - utilizar con archivos de registro de Apache, 457-458
- & (unión), operador
 - como símbolo de shell, 44
 - función de, 68
 - usar con trabajos en segundo plano, 53-54
- &&, comando
 - ejecutar comandos con, 37
 - usar con el kernel, 682
- * (asterisco)
 - buscar caracteres con, 45
 - como símbolo de shell, 44
 - función de, 68
 - usar con el archivo .Xresources, 159
 - usar con el historial de la shell C, 60
 - usar en el daemon syslogd, 540
- . (punto), volver a ejecutar la secuencia de comandos .bash_profile con el comando, 103
- . (punto), usar archivos, 116
- . (punto)
 - representar directorios con, 68,90
 - usar con extensiones, 116
- .. con el comando cd, usar el símbolo, 124
- / (directorio raíz)
 - contenido del, 118
 - función en FHS, 584-585
 - indicar, 120
- / (diagonal), usar con la línea de comandos BASH, 39
- : (dos puntos), usar en la shell Z, 63
- ; (punto y coma), separar comandos con, 37,44
- ? (signo de interrogación)
 - función de, 68
 - coincidir caracteres individuales con, 45-46
 - como símbolo de shell, 44
- @ (arroba), usar con la acción automática de completar de BASH, 39
- [] (llaves)
 - como símbolo de shell, ejecución de, 44
 - función de, 68
 - relacionar un rango de caracteres con, 46
 - usar con operaciones de prueba, 78-79
- \ (diagonal invertida)
 - despliegue, 100

ejecutar como símbolo de shell, 44
 usar con comandos, 36
 usar con indicadores de comandos, 99
 usar con signos de admiración (!), 60
 usar para citar caracteres, 46-47
 ^ (acento circunflejo)
 representar palabras con, 60
 usar en la shell C, 57
 ` (comillas invertidas)
 comparación con comillas sencillas (') y, 70
 usar con comandos, 70
 {} (corchetes), usar con patrones, 47
 | (operador de canalización)
 como símbolo de shell, 44
 conectar comandos con, 51
 usar con salida estándar, 49
 | &, ejecución del símbolo de shell, 44
 ~ (tilde), usar con la acción automática de completar de BASH, 39
 < (menor que), carácter
 como símbolo de shell, 44
 función de, 68
 = (operador de asignación)
 usar con servicios xinetd, 418
 usar con variables de shell, 68
 =~, operador, usar con la shell TCSH, 82
 ==, operador, usar con la shell TCSH, 82
 > (mayor que), carácter
 como símbolo de shell, 44
 función de, 68
 redireccionar la salida estándar con, 48-50
 >! (sobrescribir), símbolo de shell, ejecución, 44
 >& (redirección), símbolo de shell, ejecución, 44
 >> (operador de redirección)
 como símbolo de shell, ejecución de, 44
 usar con la salida estándar, 50,52
 2> símbolo de shell, ejecución, 44
 2>>, usar para adjuntar error estándar, 52
 20-storage-methods.fdi, contenido del archivo, 656
 50-udev.nodes, contenido del archivo, 657-658
 ' (comillas sencillas)
 comparar con comillas invertidas ('), 69
 usar con comandos, 69
 ./ antes de un comando, con el signo, 232
 " " (comillas), usar con caracteres, 47, 68-69

A

.a, extensión, significado, 233
 AbiWord, procesador de palabras, características, 243
 accept y reject, herramientas de la línea de comandos
 CUPS, características, 512
 ACCEPT, destino, usar con paquetes, 379-380, 383
 ACCEPT, directiva, incluir en la secuencia de comandos mifiltro, 393
 acceso PPP en la línea de comandos, implementación de, 737-738
 access.db, archivo, usar con Sendmail, 497
 acción automática de completar, ejemplos, 39
 acción de completar nombres de archivo, aplicar en TCSH, 62
 acento circunflejo (^)
 representar palabras con, 60
 usar en la shell C, 57
 ACL (lista de control de acceso)
 configurar con NFS4, 766
 crear para Squid, 470
 AddEncoding, directiva, usar con el servidor Web Apache, 455
 AddHandler, directiva
 usar con el servidor Web Apache, 455
 usar con SSI (Includes del lado del servidor), 462
 AddLanguage, directiva, usar con el servidor Web Apache, 455
 Addprincipal, comando, usar con la herramienta kadmin, 372
 AddType, directiva
 usar con el servidor Web Apache, 455
 usar con SSI (Includes del lado del servidor), 462
 Administrador de la red
 características, 733-734
 configuración de redes inalámbricas, 735-737
 Administrador de la red de GNOME. Véase Administrador de la red
 Administrador de máquina virtual, características, 686-687
 Administrador de tareas y monitor de rendimiento de KDE (KSysguard), características, 546
 administrador de volumen lógico (LVM). Véase LVM (Logical Volume Management, administrador de volumen lógico)
 Administrador de volúmenes de GNOME acceso a unidades DVD/CD-ROM, 177
 características, 176-178
 administrador del sistema, capacidades del, 523-524
 administradores de despliegue
 características, 19-20
 GDM (GNOME Display Manager), 164-166
 KDM (KDE Display Manager), 166-167

- tipos de, 160
- usar con X Window System, 160-162
- XDM (X Display Manager), 163-164
- y sesiones, 162-163
- administradores de ventana, usar el escritorio
 - GNOME con, 175-176
- Advanced Package Tool (APT), usar con Debian, 228
- agente de correo de usuario (MUA), definición, 477
- agentes de entrega de correo (MDA), características, 477
- agentes de transferencia de correo (MTA). *Véase* MTA (agentes de transferencia de correo)
- agentes de transporte de noticias, usar con noticias Usenet, 278-279
- agetty, usar el programa, 661
- AH (encabezado de autenticación), aplicación de instrucciones a, 351
- AIDE (entorno avanzado de detección de intrusión), implementación de detección de intrusos con, 325
- .aiff, descripción del tipo de archivos Web, 283
- AIM (AOL Instant Messenger), características de, 305
- alias
 - crear para comandos, 69
 - soporte en Sendmail, 485,487
- alias, comando
 - envío de, 91-93
 - usar con la shell TCSH, 107-108
- Alias, dispositivo, usar con el servidor Web Apache, 453
- alias IP, configurar, 742-743
- allow, directiva, usar con el servidor Web Apache, 453
- allow, palabra clave, usar en SELinux, 339-340
- AllowOverride, directiva, usar con el servidor Web Apache, 453
- ALSA (Advanced Linux Sound Architecture), sitio Web para, 260-261
- ALT, tecla. *Véase* también métodos abreviados de teclado
 - usar con listas de historial, 41
 - usar en la acción automática de completar de BASH, 39
- Amanda, respaldo de host con, 695-698
- amdump, comando, creación de copias de seguridad, 697-698
- anacron, usar cron con, 531
- AND, operador lógico, 78
- Anonymous, directiva, usar con el servidor Web Apache, 456
- AOL Instant Messenger (AIM), características de, 305
- Apache en Linux, características de las instalaciones, 446
- Apache Jakarta Project, características, 445-446
- Apache, servidor Web
 - archivos de registro en, 457-458
 - autenticación en, 456-457
 - características de, 444-445
 - configuración de archivos para, 448
 - configuración en el nivel directorio de, 452-453
 - configuración global de, 449-451
 - configuración MPM de, 450-451
 - directiva AddEncoding utilizada con, 455
 - directiva AddHandler utilizada con, 455
 - directiva AddLanguage utilizada con, 455
 - directiva AddType utilizada con, 455
 - directiva Alias utilizada con, 453
 - directiva allow utilizada con, 453
 - directiva AllowOverride utilizada con, 453
 - directiva Anonymous utilizada con, 456
 - directiva AuthConfig utilizada con, 453
 - directiva AuthDBGroupFile utilizada con, 457
 - directiva AuthDBMGroupFile utilizada con, 457
 - directiva AuthDBMGroupUserFile utilizada con, 457
 - directiva AuthDBUserFile utilizada con, 457
 - directiva AuthGroupFile utilizada con, 456-457
 - directiva AuthName utilizada con, 456
 - directiva AuthType utilizada con, 456
 - directiva AuthUserfile utilizada con, 456-457
 - directiva CustomLog utilizada con, 457-458
 - directiva DefaultType utilizada con, 455
 - directiva deny utilizada con, 453
 - directiva Directory utilizada con, 452
 - directiva DirectoryIndex utilizada con, 454
 - directiva FancyIndexing utilizada con, 455
 - directiva FileInfo utilizada con, 453
 - directiva FormatLog utilizada con, 457
 - directiva Keep Alive utilizada con, 449
 - directiva KeepAliveRequests utilizada con, 449-450
 - directiva Limit utilizada con, 453
 - directiva Listen utilizada con, 450,459
 - directiva LoadModule utilizada con, 450
 - directiva LogFormat utilizada con, 457
 - directiva MaxClients utilizada con, 451
 - directiva MaxRequestsPerChild utilizada con, 451
 - directiva Options utilizada con, 453
 - directiva Redirect utilizada con, 454
 - directiva ServerAdmin utilizada con, 451
 - directiva ServerLimit utilizada con, 451

- directiva ServerName utilizada con, 451-152
- directiva ServerRoot utilizada con, 449
- directiva ServerTokens utilizada con, 449
- directiva StartServer utilizada con, 451
- directiva ThreadsPerChild utilizada con, 451
- directiva Timeout utilizada con, 449
- directiva TransferLog utilizada con, 458
- directiva TypesConfig utilizada con, 455
- directiva UseCanonicalName utilizada con, 460-461
- directiva VirtualDocumentRoot utilizada con, 460
- directiva VirtualScriptAlias utilizada con, 460
- directivas usadas con, 448-149
- e indización automática de directorio, 455
- host virtual basado en nombre en, 459
- host virtual dinámico en, 459-462
- host virtual en, 458-462
- inicio y detención, 447-448
- módulos para, 450
- URL para, 12
- usar aplicación apxs con, 450
- usar MPM con, 447
- y archivos CGI (interfaz común de puerta de enlace), 455
- y host virtual basado en IP, 459,461
- y nombres de ruta URL, 453-454
- y tipos MIME, 454-155
- Apache-SSL, descripción de, 444
- apagado, comandos
 - descripción, 525
 - ubicación, 401
 - usar, 19-21, 534-535
- aplicaciones
 - de audio, soporte a, 260-261
 - de seguridad de red de sitios Web, 374
 - hacer que queden disponibles para los usuarios, 234
 - X remotas, usar, 148-149
- Aplicaciones preferidas, características de la herramienta, 184
- apol Policy Analysis, herramienta, usar en SELinux, 334
- applets. *Véase también* applets de GNOME
 - características de, 23-24
 - para el panel KDE, 206
- APT (Advanced Package Tool), usar con Debian, 228
- apxs, aplicación, usar con el servidor Web Apache, 450
- árbol, estructura, explicación, 117
- archivados de archivos tar, compresión, 139
- archivar en discos flexibles, 137
- archiveros. *Véase también* archivos
 - actualizar con la utilería tar, 136-137
 - comprimir, 137-138
 - crear con tar, 134-136
 - crear en cinta, 138
 - extraer con tar, 136
 - extraer contenidos, 139
 - revisar contenidos de, 230
- archiveros comprimidos. *Véase también* archivos
 - descargar, 229
 - identificar, 10
 - instalar software de, 228-233
 - seleccionar el directorio de instalación para, 230
- archivos. *Véase también* archiveros; archiveros comprimidos
 - archivar, 134-138
 - archivar y comprimir, 133-134
 - asignar varios nombres a, 130-132
 - borrar, 45,130
 - cambiar grupos, 565
 - cambiar nombre, 129
 - cambiar propietarios, 565
 - comprimir con la utilería bzip2, 140
 - comprimir con la utilería gzip, 138-140
 - comprimir con la utilería Zip, 140-141
 - copiar, 126-129
 - crear copias de seguridad, 81
 - crear copias de seguridad y restaurar, 693-694
 - descomprimir con la utilería Zip, 141
 - desplegar información detallada acerca de, 116
 - desplegar, 120
 - ejemplos de tipos, 117
 - enumerar, 119
 - estructura, 117-119
 - formato de flujo de bytes, 117
 - forzar la sobrescritura de, 44
 - imprimir, 121
 - mover, 129
 - nombrar, 116
 - nombre de archivos de dispositivo, 504, 643
 - operaciones de permiso, 564
 - probar, 78
 - propietario, 563-565
 - proteger de salida redirigida, 109
 - recuperar, 701-703
 - repaldar dispositivos, 137
- archivos compartidos
 - acceso en sistemas Linux, 31
 - configuración, 31

- archivos comprimidos
 - con la utilidad bzip2, 140
 - con la utilidad gzip, 138-140
- archivos de dispositivo
 - como vínculos simbólicos, 645-647
 - contenido, 639
 - crear con el comando MAKEDEV, 657-658
- archivos de encabezado de kernel, compilación de módulos, 670
- archivos de firma, usar con clientes de correo, 265
- archivos de información de dispositivo
 - directorios, 656-657
 - propiedades de HAL, 654
- archivos de inicialización
 - definir variables de parámetros shell, 96
 - del sistema, ubicación, 396
 - ubicación, 396
- archivos de inicialización en TCSH
 - .login, 112
 - .logout, 113-114
 - .tcshrc, 112-113
- archivos de inicio de sistema, rc.sysinit y rc.local, 401
- archivos de registro
 - generar y administrar en Apache, 458
 - personalizar en Apache, 457
 - revisar actividad sospechosa, 325
- archivos, estructura, función, 8
- archivos inittab, relacionar al archivo termcap, 661
- argumentos. *Véase también* argumentos de secuencia de comandos
 - alias, 92
 - analizar en la línea de comandos, 68
 - ejecutar comandos como, 37
 - utilizar en comandos, 28
- argumento de comandos, relación con variables de shell, 68
- argv, matriz, usar, 72-73
- Arquitectura Advanced Linux Sound Architecture (ALSA), sitio Web para, 260-261
- ARRAY, entrada, generación para RAID, 637
- arroba (signo @), usar con la acción automática de completar de BASH, 39
- AS (servidor de autenticación), función en Kerberos, 370
- ASCII predeterminado con clientes FTP, usar de, 295-296
- asociaciones de seguridad (sa), usar, 351-352
- asterisco (*)
 - buscar caracteres con, 45
 - ejecución del símbolo de shell, 44
 - función de, 68
 - usar con el archivo .Xresources, 159
 - usar con el historial de la shell C, 60
 - usar en el daemon syslogd, 540
- AT&T Unix, la shell Korn. *Véase* Korn, la shell
- ATTRS, clave, usar con el comando udevinfo, 651
- .au, descripción del tipo de archivo, 283
- audit2allow, comando, usar en SELinux, 334-335
- auditd, función del servidor, 541-542
- Auditing System, características de, 541-542
- autenticación
 - en el servidor Web Apache, 456-457
 - en Kerberos, 369-371
 - en OpenSSH, 361
 - en ProFTPD (Daemon FTP profesional), 436
 - función de, 373
- AUTH, comando, usar con SASL, 496
- AuthConfig, directiva, usar con el servidor Web Apache, 453
- AuthDBGroupFile, directiva, usar con el servidor Web Apache, 457
- AuthDBMGroupFile, directiva, usar con el servidor Web Apache, 457
- AuthDBMGroupUserFile, directiva, usar con el servidor Web Apache, 457
- AuthDBUserFile, directiva, usar con el servidor Web Apache, 457
- AuthGroupFile, directiva, usar con el servidor Web Apache, 456-457
- AuthName, directiva, usar con el servidor Web Apache, 456
- AuthType, directiva, usar con el servidor Web Apache, 456
- AuthUserfile, directiva, usar con el servidor Web Apache, 456-457
- autofs, servicio, montaje de NFS bajo pedido, 770
- .avi, descripción del tipo de archivo Web, 283
- ayuda sensible al contexto, disponibilidad, 29
- Ayuda, acceso a recursos, 28-30
- azap, sintonización de canales de TV con la herramienta, 263-264

B

- BackupPC, características de la herramienta, 694-695
- Balsa, cliente de correo, usar con GNOME, 270
- base de datos de seguridad (SAD), agregar
 - asociaciones de seguridad, 351
- base de datos, servidores
 - disponibilidad de, 515
 - MySQL, 517-520
 - PostgreSQL, 520

base de datos, sistemas de administración
 bases de datos SQL, 245-247
 bases de datos Xbase, 248
 y SQL, 516-517

bases de datos de directiva (SPD), agregar directivas, 351

BASH_ENV, contenidos de la variable, 99

BASH, especificación del entorno, 99

.bashjogout, características del archivo, 106-107

BASH, línea de comandos, capacidades de acción de completar, 38-39

.bash_profile, secuencia de comandos, 102-103
 colocar asignaciones de PATH en, 234-235
 contenido de, 101
 editar, 102-103
 función de, 554-555
 volver a ejecutar manualmente, 103

.bashrc, archivo
 configurar la shell BASH con, 105-106
 función, 554-555

BASH, shell. Véase también shell
 acceso al manual de referencia, 37
 archivos de configuración para, 91
 capacidades de secuencias de comandos de, 65-66
 características de, 89
 configuración, 105-106
 control de operaciones, 93-94
 edición en la línea de comandos, 37
 estructuras de control, 79-80
 estructuras de control condicional, 78-80
 lista de historial, 40
 operadores de prueba, 78
 sitio Web, 35
 utilería de historial en, 40-42

Bell Labs, popularidad de Unix en, 7

Berkeley Software Distribution (BSD), desarrollo, 7

Beryl, desarrollo, 176

bg, ejecución del comando, 53,55

bibliotecas
 de desarrollo, usar con programas compilados, 232
 diferencia entre bibliotecas compartidas y estáticas, 232-233
 disponibilidad de, 233

biff, utilería de notificación de correo, características, 273-274

/bin, directorio, contenido, 537,586

bin, directorio, usar con FTP anónimo, 426-427

/bin, significado de la extensión, 220

binaria, opción predeterminada, usar con clientes FTP, 295-296

bit adhesivo, configuración de permisos, 569-570

Blackdown, proyecto, obtención de puertos Linux de Java, 287

BlankTime, valor, usar en Xorg, 152

boleto que otorga un boleto (TGT), función en Kerberos, 369-370

boletos
 destruir, 371
 enumerar en Kerberos, 369

/boot, directorio, kernels en, 674

break, estructura, usar en la shell TCSH, 86

breaksw, palabra clave, usar en la shell TCSH, 82-83

BSD (Berkeley Software Distribution), desarrollo de, 7

bunzip2, comando, archivos de descompresión con, 140, 229

.bz2, archivos, descompresión, 229

.bz2, extensión, significado, 220

bzip2, utilería, compresión de archivos con, 140

bzimage, archivo para kernels, instalación manual, 682-683

C

C, shell. Véase también shell
 acceso, 57
 comandos de sustitución de patrón en, 57
 desarrollo de, 56-57
 edición de línea de comandos, 57
 hacer referencia a comandos en, 59
 variables de entorno en, 76

caché de proxy inversa, configuración para Squid, 475

cachés, configurar en Squid, 473-474

cadenas
 definir cadenas de usuario, 393
 en Netfilter, 377
 tipos de, 376

cadenas de comandos
 comparar, 77-78
 comparar en la shell TCSH, 83
 convertir en citas, 68-69
 probar en la shell TCSH, 82
 probar para igualdad y desigualdad, 82
 usar con variables shell, 68-69

capa de abstracción de hardware (HAL), relación con archivos de dispositivos de impresora, 504

capa de autenticación y seguridad simples (SASL), usar con Sendmail, 496

capa de conectores seguros (SSL), usar con servidores Web, 464-466

caracteres

- borrar, 21,28
- convertir en citas, 44, 46-47
- especificar rangos de, 46
- navegar, 37-38
- relacionar en nombres de archivo, 44
- relacionar, 45-46
- transportar, 37
- caracteres especiales en la shell de usuario,
 - deshabilitar, 109
- caracteres meta, convertir en citas, 68-69
- carga de seguridad de cifrado (ESP), usar, 351
- cargador de arranque, usar con paquetes de kernel, 675-676
- cargador de módulo de kernel (Kmod, Kernel Module Loader), funciones, 664-665
- case, efecto del comando, 78-80
- cat, comando
 - envío, 39,50,120-121
 - redirigir la salida estándar de, 52
- cbackup, secuencia de comandos, 81,87
- CD, archivos de imagen, creación, 611
- cd, comando
 - envío, 118,123
 - usar con el símbolo .., 124
- CD, iconos de dispositivo, despliegue en KDE, 201
- CD, lectores, vínculos simbólicos utilizados, 648
- CD/DVD, dispositivos, revisión de los vínculos HAL con, 261
- cddrecord, comando, 612
- cdpath, variable, usar en la shell TCSH, 110
- CD-R/RW, discos, copia de archivos en, 129
- cdrecord, aplicación, usar con DVD, 610
- CD-ROM, discos
 - acceso con el Administrador de volúmenes de GNOME, 177
 - acceso desde el escritorio KDE, 208
 - grabar con el Administrador de volúmenes de GNOME, 177
 - guardar datos, 610-613
 - montaje, 604-605
 - quemar datos, 27
- CD-ROM, unidades, nombres de dispositivo para, 592
- cedega, usar juegos de Windows en Linux, 31
- Centro de control de KDE, configurar KDE con, 214-216
- centro de distribución de claves (KDC), función en Kerberos, 369-370
- Centros Linux, URL para, 5
- certificados
 - usar con IPsec, 355
 - usar en SSL (capa de conectores seguros), 465
- CGI, archivos (interfaz común de puerta de enlace),
 - usar con el servidor Web Apache, 455
- chage, comando
 - envío, 556
 - opciones para, 557
- charla de retransmisión en Internet (IRC),
 - características de, 305
- chcon, comando, usar en SELinux, 334
- check-cdrom.sh, secuencia de comandos, función, 648
- checkmodule, comando, usar en SELinux, 341, 344
- checkpolicy, comando, usar en SELinux, 344
- chgrp, comando, usar, 565
- chkconfig, comando
 - configurar servicios xinetd para, 409-410
 - eliminar y agregar servicios con, 409
 - enumerar servicios con, 407-408
 - funcionalidad de, 410
 - habilitar para servicios xinetd, 418
 - habilitar y deshabilitar servicios xinetd con, 409
 - iniciar y detener servicios con, 408-409
 - usar con rsync, 428
 - usar la opción off con, 408
 - usar la opción reset con, 408
- chmod, comando
 - cambiar permisos con, 563
 - usar con directorios de grupo, 560
 - usar con FTP anónimo, 426-427
- chown, comando
 - usar, 565
 - usar con FTP anónimo, 426
- chroot, operación, usar con ProFTPD, 438
- CIDR (enrutamiento entre dominios y sin clase),
 - relación con direcciones TCP/IP, 714-717
- cifrado. *Véase también* GnuPG (GNU Privacy Guard)
 - clave utilizada en, 314
 - combinación con firmas digitales, 315
 - de sistemas de archivos, 326
 - en OpenSSH, 360-361
- cinta, archivado, 138
- claves
 - cargar para OpenSSH, 365
 - determinar tareas asociadas, 37
 - en OpenSSH, 360-361
- claves privadas, usar en OpenSSH, 360-361
- claves públicas
 - en OpenSSH, 360-361
 - función de cifrado, 314
 - importar para paquetes de software, 323
 - revisar en paquetes RPM, 324-325
 - validar para paquetes de software, 324

clientes de correo de línea de comandos

Mutt, 271

utilería Mail, 272

Cluster Project Page, sitio Web para, 780

cman, secuencia de comandos, ejecución para GFS 2, 780

Coda, características, 778-779

codecs, disponibilidad, 258, 263

CodeWeavers, soporte a Wine, 32

colores, lista en X Window System, 159

comandos

alias, 92-93

colocar en segundo plano, 54

como filtros, 51

convertir en cita, 69

crear bucles, 77

desplegar antes de la ejecución, 108

desplegar con la lista de historial, 40-43

desplegar los últimos comandos ejecutados, 28

ejecutar, 44

ejecutar argumentos, 37

ejecutar en la misma línea, 37

enumerar el uso reciente de, 40

enviar datos entre, 50-51

formato para, 28hacer referencia a la shell C, 59

hacer referencia a rangos de edición, 43

insertar en la línea de comandos, 36-37

interrumpir y detener, 36

obtener valores, 70

para editar, 38

para movimiento, 38

realizar operaciones de historial, 61

seguimiento en la shell BASH, 40-42

separar, 44

ubicación de, 233

comandos de acceso remoto, función, 307-308

comandos de completar, ejemplos, 40

comillas(" "), usar con caracteres, 47, 68-69

comillas invertidas (')

diferencia entre comillas sencillas (') y, 70

usar con comandos, 70

comillas sencillas (')

diferencia entre comillas invertidas (') y, 69

usar con comandos, 69

comillas

distinguir los tipos de, 70

usar con caracteres meta, 68-69

usar con comandos, 69

Common Unix Printing System (CUPS). *Véase también*

CUPS (Common Unix Printing System)

Compiz, usar con el escritorio GNOME, 176

Compiladores y herramientas de Linux (gcc), URL, 13

compress, comando, usar, 140

conexiones, configurar con setkey, 351-353

conexiones de red

configurar con iwconfig, 735

monitoreo con el programa netstat, 742

config, herramienta, inicio de, 678

Configuración de Windows, herramienta,

instalación, 32

configuración, archivos

archivos de configuración de usuario, 552

enumerar, 90

funciones de, 91

ubicación de, 90, 537, 587

usar Makefiles, 231

configuración del kernel, características, 679-681

configure, comando

con ./ antes, 232

opciones, 232

conjunto redundante de discos independientes

(RAID). *Véase también* RAID (conjunto redundante de discos independientes)

contenido de archivero, despliegue con tar, 134

contenido de archivo, desplegar en pantalla, 119

continue, estructura, usar en la shell TCSH, 86

contraseña, archivos de

campos, 553

ubicación, 553-554

contraseña cifrada, desactivar el soporte, 553

contraseña, usar herramientas, 554

contraseñas

administrar en Kerberos, 370

controlar contraseñas de usuario, 556

insertar y proteger, 21

control de acceso basado en funciones (RBAC),

método, usar en SELinux, 336

control de acceso basado en funciones (RBAC), modo,

usar en SELinux, 328

copiar archivos, 126-129

copias de seguridad

con la utilería dump, 698-701

de archivos, 81

de archivos y directorios, 693-694

de host con Amanda, 695-698

grabación con la utilería dump, 700

recuperación, 701-703

usar herramientas de archivador con,

693-694

corchetes ([])

buscar rangos de caracteres con, 46

como símbolo de shell, 44

- función de, 68
- usar con operaciones de prueba, 78-79
- correo, clientes
 - archivos de firma, 265
 - clientes de línea de comandos, 271-273
 - Emacs, 271
 - en GNOME, 269-270
 - Evolution, 267-268
 - KMail, 270
 - MIME (extensiones de correo de Internet de propósitos múltiples), 266
 - notificaciones de correo recibido, 273-274
 - protocolos, 267
 - SquirrelMail, 270-271
 - Thunderbird, 268-269
- correo recibido, notificaciones, 273-274
- correo, servidores. *Véase también* Sendmail, servidor de correo
 - IMAP (protocolo de acceso a correo de Internet), 498-499
 - POP (protocolo de oficina postal), 498-499
 - Postfix MTA (agente de transferencia de correo), 479-484, 501
 - Sendmail, servidor de correo, 487
- Courier MTA (agente de transferencia de correo), características, 477
- Courier-IMAP, sitio Web para el servidor, 500
- cp, comando
 - envío, 126
 - insertar en la línea de comandos, 36-37
 - significado de, 124
 - usar con directorios, 129-130
 - usar la opción -r con, 130
- CREATE TABLE, comando, usar, 517
- cron, daemon
 - edición del historial en la shell C, 529
 - nombres de directorio, 531
 - usar anacron, 531
 - variables de entorno, 528
- cron, secuencias de comandos de directorio, ejecución, 530-531
- cron, servicio, programar tareas con, 527
- crontab, comando, usar, 529
- crontab, entradas
 - campos para, 527-528
 - colocar en el directorio cron.d, 528-529
- CrossOver Office
 - disponibilidad de, 32
 - ejecutar, 238-239
 - sitio Web para, 238
- cryptsetup, comando, usar, 326
- csch, comando, acceso a la shell C, 57
- CTRL, tecla. *Véase también* métodos abreviados de teclado
 - usar con la lista de historial, 41
 - usar en la acción automática de completar en BASH, 39
- cualquier difusión, dirección, explicación, 721
- cuentas, acceso en otros sistemas, 309
- cuota, herramientas
 - edquota, 571-572
 - usar, 571
- cuotas de disco
 - establecer con edquotas, 571-572
 - y herramientas de cuotas, 571
- CUPS (Common Unix Printing System)
 - características de, 503-504
 - configurar, 508
 - configurar en GNOME, 505
 - configurar en KDE, 505
 - configurar impresoras remotas en, 507
 - habilitar en Samba, 507
- CUPS, clases de impresora, creación, 507
- CUPS, clientes de impresión de línea de comandos
 - lprm, 510
 - lpc, 510
 - lpq e lpstat, 510
 - lpr, 509
- CUPS, directivas, usar, 508
- CUPS, herramienta de configuración, usar, 506
- CUPS, herramientas administrativas de línea de comandos
 - accept y reject, 512
 - disponibilidad de, 510-511
 - enable y disable, 512
 - lpadmin, 511
 - lpinfo, 512
 - lpoptions, 511-512
- cupsd, servidor, función, 402
- cupsd.conf, archivo
 - contenido de, 507
 - ubicación de, 508
- curl, cliente de Internet, características, 293
- curl, cliente, descripción, 291
- CustomLog, directiva, usar en el servidor Web Apache, 457-458
- CVS (sistema de versión concurrente), función de, 235
- Cyrus IMAP, servidor, sitio Web para, 500

D

- daemons
 - apagado, 412

- inicio y apagado de, 403
 - secuencias de comandos relacionadas con, 402
 - y software de servidor FTP, 423
- daemon extendido de servicios de Internet (xinetd).
Véase también extensiones de xinetd (daemon extendido de servicios de Internet)
 - selección de archivos con, 45
 - usar con nombres de archivos, 116
- date, comando
 - descripción de, 525
 - ejecución, 37
 - envío, 526-527
- DB2, base de datos, características y sitio Web, 12, 246-247
- .deb, extensión, significado, 10, 220
- Debian Package tool (dpkg), herramienta, usar, 228
- Debian
 - características de, 227-228
 - niveles de ejecución, 405
 - URL, 5
- DefaultType, directiva, usar con el servidor Web Apache, 455
- definidas por el usuario, cadenas, crear para IPtables, 380-381
- deny, directiva, usar con el servidor Web Apache, 453
- depmod, comando, detección de dependencias, 666
- depósitos de terceros, descargar software, 11
- designadores de palabra en la shell Z, usar, 63
- detección de intrusos, implementar, 325
- /dev/mapper, directorio, contenido, 621
- /dev/pts, entrada, significado, 661
- /dev, contenido del directorio, 590
- device-mapper, usar con LVM, 621-622
- df, comando, usar, 593-594
- dhcp, habilitación para máquinas virtuales Xen, 691
 - ejecución de servidores, 759
 - función para IPv4, 750
 - host cliente IPv4, configuración, 750
 - implementación de actualizaciones DNS dinámicas, 755-757
 - servidor IPv4, configuración, 751, 753-754
 - subredes, revisión general, 757-758
 - usar direcciones fijas, 759-760
- DHCP (protocolo de configuración dinámica de host), direcciones IPv4, 754-755
- DHCPv6, proporcionar la configuración automática con estado, 748-749
- Dia, disponibilidad del programa de dibujo, 244
- diagonal (/), usar con la línea de comandos BASH, 39
- diagonal invertida (\)
 - despliegue, 100
 - ejecutar como símbolo de shell, 44
 - usar con ! (signo de admiración), 60
 - usar con comandos, 36
 - usar con peticiones, 99
 - usar para citar caracteres, 46-47
- digiKam, características del administrador de fotografías, 256
- dirección de correo, componentes, 478
- dirección de multidifusión, explicación, 721-723
- direccionamiento IP basado en clase, organización, 712-713
- direcciones de servidor de nombres, función, 719-720
- direcciones de transmisión, función de, 719
- direcciones IPv4 dinámicas, usar con DHCP, 754-755
- directivas de seguridad, usar, 352. *Véase también* directivas
 - directivas. *Véase también* directivas de seguridad
 - agregar a SPD (bases de datos de directiva), 351
 - configurar para recibir host, 353
 - diferencia entre propiedades para dispositivos, 655
 - para dispositivos de almacenamiento extraíbles, 656
- directorio compartido en KDE, contenido del, 216
- directorio raíz (/)
 - contenido del, 118
 - función en FHS, 584-585
 - indicar, 120
- directorios. *Véase también* principal, directorio
 - acceso, 118
 - administrar, 121-124
 - archivos de configuración, 90
 - borrar, 130
 - buscar, 124-126
 - cambiar, 118
 - comprimir con la utilería Zip, 140-141
 - contenidos, 117
 - copiar y mover, 129-130
 - crear y eliminar, 122-123
 - desplegar contenidos de, 123
 - desplegar nombres de ruta absolutos para, 122
 - directorios de sistema en FHS, 585-587
 - funciones, 120
 - localizar, 126
 - navegar, 123
 - operaciones de permiso, 564
 - organizar archivos, 8
 - permisos asignados a, 561-563
 - propietario, 563-565

- recuperar, 701-703
- respalda y restaura, 693-694
- Directory, directiva, usar con el servidor Web Apache, 452
- DirectoryIndex, directiva, usar con el servidor Web Apache, 454
- disable y enable, herramientas de línea de comandos de CUPS, características, 512
- disco, despliegue al usar, 544
- disco duro, particiones
 - crear, 637
 - crear LVM, 623-624
 - crear para dispositivos RAID, 631
 - montaje, 605-606
- discos de arranque, colocación de kernel en, 683
- discos duros formateados para MS-DOS, acceso a, 132-133
- discos flexibles
 - acceso del escritorio KDE, 201, 208
 - acceso para MS-DOS, 132-133
 - archivar en, 137
 - montar, 604
- disklist, archivo, usar con Amanda, 697
- disparcivo, secuencia de comandos, 75-76
- dispositivos
 - archivar, 134-138
 - crear con el comando mknode, 659-660
 - entradas de propiedades, 654-655
 - hacerse accesible vía HAL, 652
 - identificar, 651
 - obtener información acerca de, 639-641
 - respalda archivos, 137
 - tipos, 657
- dispositivo de bloqueo, explicación, 657
- dispositivo de carácter, explicación, 657
- dispositivos de almacenamiento extraíbles, directivas, 656
- dispositivos de copia de seguridad, restauración de, 703
- dispositivos de entrada
 - archivos de configuración, 160
 - detección, 662
- dispositivos del sistema, obtener información, 639-641
- dispositivos manuales, creación, 658-659
- dispositivos y medios extraíbles, usar, 27
- dispprimera, secuencia de comandos, 73
- disputima, secuencia de comandos, 74
- DivX para Linux
 - características, 264
 - sitio Web, 262
 - soporte, 27
- DLL, archivos, usar con Windows, 32
- dmraid, herramienta de RAID, características, 626-627
- DNAT, destino, usar, 396-397
- DNAT, reglas, explicación, 385
- DNS (servidor de nombres de dominio)
 - relacionar con servidores de correo, 478-479
 - revisión general, 725-726
 - y el archivo hosts.conf, 726-727
 - y NSS (intercambio de servicio de nombres), 727-729
- do, estructura de control de bucle, función, 80
- documentación de aplicación, acceso a, 29
- documentación, disponibilidad, 13-15
- dominios virtuales para Sendmail, definición, 496
- dominios, relación con servidores de correo, 477
- done, palabra clave, explicación, 81
- DontZap, valor, usar en Xorg, 152
- DontZoom, valor, usar en Xorg, 152
- dos puntos (:), usar en la shell Z, 63
- Dovecot, características, 499
- dpkg (herramienta Debian Package), usar, 228
- DROP, destino, usar con paquetes, 379-380
- DROP, directiva
 - especificar para la cadena FORWARD, 396
 - incluir en la secuencia de comandos IPtables, 389-390
- dummy, opción, usar con el comando cdrecord, 613
- dump, comando, relación con el archivo fstab, 598
- dump, utilería, crear copias de seguridad, 698-701
- DVB, soporte a recepción, 263-264
- DVD, discos
 - acceso con el Administrador de volúmenes de GNOME, 177
 - grabar con el Administrador de volúmenes de GNOME, 177
 - grabar datos en, 27
- DVD, iconos de dispositivo, despliegue en KDE, 201
- DVD, imágenes de distribución, descargar con BitTorrent, 220-221
- DVD, lectores, vínculos simbólicos utilizados, 648
- DVD, lista de reproductores, sitio Web, 262
- dvdrecord, comando, usar, 612
- DVD, reproductores, acceso, 262-263
- DVD::rip
 - características, 263
 - sitio Web, 262
- DVD+RW, herramientas, disponibilidad, 613
- DVD-R/RW, copia de archivos en discos, 129
- DVD-ROM, grabación de datos en discos, 610-613
- DVI, disponibilidad de visores, 244

E

- e2fsck, comando, usar, 594
- e2label, usar con sistemas de archivos, 600
- echo, comando
 - usar con variables shell, 68
 - usar en la shell TCSH, 108
- edición de línea de comandos
 - disponibilidad en la shell BASH, 37
 - en la shell C, 57
- editores
 - Emacs, 249-250
 - Gedit, 248
 - Gvim and Vim, 250-254
 - Kate, KEdit, and KJots, 248-249
- eject, comando, usar con CD-ROM, 604
- ejecución (x), permiso
 - explicación, 562
 - invocar, 566
 - usar máscaras binarias, 567-568
- Ekiga, aplicación VoIP, características, 304-305
- eliminar caracteres, 21
- ELinks, explorador de modo de línea, características, 286
- else, estructura de control condicional, función, 79
- else, palabra clave, usar en TCSH, 85
- elsels, secuencia de comandos, 80, 85
- Emacs, disponibilidad de clientes de correo, 271
- Emacs, editor
 - características, 249-250
 - usar teclados con, 251
- enable y disable, herramientas de línea de comandos de CUPS, características, 512
- encabezado de autenticación (AH), aplicación de instrucciones a, 351
- endif, palabra clave, usar en la shell TCSH, 82
- endsw, palabra clave, usar en la shell TCSH, 82-83
- engaño de IP, protección contra, 390
- enmascaramiento. *Véase también* IP, enmascaramiento
- enrutamiento entre dominios y sin clase (CIDR),
 - relación con direcciones TCP/IP, 714-717
- enteros, comparación, 77-78
- Entorno avanzado de detección de intrusión (AIDE),
 - implementación de detección de intrusos con, 325
- entorno, función, 8
- entrada estándar
 - explicación, 47
 - recibir datos, 50
- entunelamiento, implementar en OpenSSH, 367-368
- env, enumerar variables de shell con el comando, 96
- Epiphany, explorador Web de GNOME,
 - características, 286
- error estándar
 - adjuntar a archivos, 52
 - redirigir, 44
 - redirigir y canalizar, 51-52
- esac, efecto de la palabra clave, 78-80
- Escritorio, directorio en KDE, contenido, 215-216
- escritorios
 - configurar con KDE, 23
 - configurar sesiones en, 27
 - inicio, 24
- escritura (w), ejecución de permisos, 562, 566
- ESP (carga de seguridad de cifrado), usar, 351
- estados, especificar para rastreo de conexión, 383.
 - Véase también* niveles de ejecución
- estructuras de control
 - en las shell TCSH y C, 81-88
 - tipos de, 77
- estructuras de control condicionales
 - definición de, 77
 - usar, 78-80
- estructuras de control de bucle, tipos, 77
- /etc, directorio, contenido, 537
- /etc/exports, archivo
 - cambiar, 765
 - entradas, 762-766
 - ejemplo, 765
- /etc/fstab, archivo
 - impacto de HAL, 652-653
 - montaje de directorios NFS por entradas en, 768-769
- /etc/gdm, directorio, contenido, 166
- /etc/group, sistema de archivos, contenido, 559-560
- /etc/gshadow, archivo
 - contenido de, 559-560
 - funciones de, 554
- /etc/hosts.allow, archivo, funciones, 767
- /etc/hosts.deny, archivo, funciones, 767
- /etc/hosts, archivo, contenido, 724
- /etc/init.d, directorio, contenido, 402-403
- /etc/mdadm.conf, archivo, contenido, 631
- /etc/nsswitch.conf, archivo, ejemplo, 729
- /etc/profile, secuencia de comandos, características, 104-105, 234
- /etc/protocols, archivo, contenido, 725
- /etc/resolv.conf, archivo, contenido, 725
- /etc/services, archivo, contenido, 725
- /etc/shadow, archivo, función, 554
- /etc/X11/xorg.conf, archivo, secciones, 150
- eth0, dispositivo, usar el operador !, 383
- etiquetas de archivos de sistema, usar, 600
- Ettercap, programa olfateador, características, 739

Evolution, cliente de correo, usar en el escritorio
GNOME, 267-268
Exim MTA (agente de transferencia de correo),
características de, 478
EXINIT, variable, función, 96
exit, comando, usar, 21, 525
EXPN, opción, deshabilitar en Sendmail, 498
export, comando
 crear variables de entorno, 75,94
 usar con variables de parámetro, 101
ext3, registro por diario, 595
extensiones de archivo, selección de archivos con, 45
extensiones de correo de Internet de propósitos
 múltiples (MIME)
 asociaciones estándar, 267
 características de, 266

F

FancyIndexing, directiva, usar con el servidor Web
Apache, 455
.fc, archivos, usar en SELinux, 340-341
fc, comando, usar con eventos de historial, 42-43
fc-list, comando, enumerar fuentes con fuentes, 26
fd, tipo de partición, usar con dispositivos RAID, 629
fdi, archivos, usar con dispositivos, 654
fdisk, herramienta
 diferencia entre el comando df y, 593
 usar con sistemas de archivos, 606-608
FEATURE, macro, usar con Sendmail, 489-490
fecha, código de indicador de comandos, 100
Fedora Linux, URL, 5
Fetchmail, usar, 274-275
fg, comando, ejecución, 53-55
FHS (estándar de jerarquía de sistema de archivos).
 Véase también sistemas de archivos
 archivos de dispositivo en, 590
 directorio /home, 588
 directorio /media, 587
 directorio /mnt, 587
 directorio raíz (/), 584-585
 directorio /usr, 587
 directorio /var, 588
 directorios de programa en, 586
 directorios de sistema, 585-587
 y /sys, 589-590
 y dispositivos CD-ROM, 592
 y dispositivos de disco duro, 592
 y dispositivos de disco flexible, 592
 y el sistema de archivos /proc, 589
file, comando, usar, 117
File Roller, aplicación

archivar y comprimir archivos, 133-134
 iniciar, 139
Filelnfo, directiva, usar con el servidor Web Apache,
453
filtrado de paquetes
 agregar y cambiar reglas, 376-378
 ejecutar, 374-375
 implementación, 375
 relación con firewalls, 373
 y cadenas, 375-376
 y destinos, 376
filtrado de red, implementación de enmascaramiento
IP, 355
filtros, comandos como, 51
find, comando
 buscar directorios, 124-125
 localizar directorios, 126
 usar la opción -name, 124
 usar la opción -print, 124
 usar la opción -type, 126
finger, comando, obtener información de usuario,
301-303
Firefox, cliente FTP, 291
 características, 291-292
 descripción, 291
Firefox, explorador Web, características, 284-285
Firefox, habilitar JRE (Java Runtime Environment), 289
firewall, acceso, bloqueo en la secuencia de comandos
mifiltro, 391
firewall, tipos de cadenas, 376
firewalls
 configuración, 387-389
 e IPtables, 374-375
 funciones, 373
firmas digitales
 combinación de cifrado con, 315
 revisión de paquetes de software, 323-325
 usar en SSL (capa de conectores seguros), 465
 usar, 314
fixfiles, comando, usar en SELinux, 344-345
Flagship, base de datos, características y sitio Web,
246
Flask, arquitectura, organización, 327-328
Fluendo, sitio Web, 258
flujo de bytes, usar con archivos, 117
for, estructura de control de bucle, función, 80
foreach, estructura de control
 efecto, 81
 usar en la shell TCSH, 86-88
FormatLog, directiva, usar con el servidor Web
Apache, 457

formato de intercambio de LDAP (LDIF), archivo
 agregar registros, 578-579
 contenido, 577-578

FORWARD, cadena
 definición de la cadena de usuario de, 392-393
 especificar la directiva DROP, 396
 evaluación, 380
 usar, 376-377

fragmentación, definición, 764

free, efecto del comando, 544

Frysk, herramienta de monitoreo, características, 544

fsck, comando, usar, 594

F-Spot Photo Manager, características de, 256

fstab, archivo
 ejemplo, 599-600
 modificar, 596, 598
 valores enteros, 598

FTP (protocolo de transferencia de archivos), función, 423

FTP anónimo
 directorios de servidor para, 426-427
 explicación de, 425
 permitir, 297, 425-427

FTP, clientes
 administrador de archivos K Desktop, 292
 características de, 290
 cliente de Internet curl, 293
 Firefox, 291-292
 gFTP, 292-293
 herramienta wget, 293
 Konqueror, 292
 Nautilus, 292
 programa lftp, 298
 programa NcFTP, 299
 y transferencia de archivos de red, 290-291

ftp con FTP anónimo, usar el grupo, 425

FTP, cuenta de usuario, requisitos, 425

FTP muy seguro, URL para un servidor, 12

ftp, programa
 capacidad de inicio de sesión automático, 297-298
 características, 293-297
 descripción, 291
 soporte de macro, 297-298

FTP, servidores
 componentes, 423
 disponibilidad, 424
 ejecutar Tux, 444

FTP, sitios
 acceso con rsync, 427-428
 buscar archivos, 283

componentes, 425
 realizar búsquedas de documento, 520
 realizar un inicio de sesión anónimo, 297

FTP, usuarios
 acceso disponible a, 424
 crear, 426

fuentes
 acceso, 26
 administrar, 25
 agregar y eliminar, 25-26
 configurar, 26
 designar en X Window System, 151-152
 disponibilidad en X Window System, 158
 enumerar, 26
 establecer en X Window System, 156
 instalar para Windows, 32

fuentes del escritorio, cambiar el tamaño de las, 25

functions, secuencia de comandos, ejecución, 412-413

fuser, comando, usar, 603

fuse-smb, herramienta, características de, 31

G

GConf, editor de configuración
 características de, 194-196
 claves en, 196

gconfig, herramienta, usar, 678-679

GDM, archivos de configuración, ubicación, 165

GDM (GNOME Display Manager)
 cambio de la pantalla de inicio de sesión, 165
 características, 164-166
 disponibilidad de, 160
 función, 19-20

gdmsetup, comando, usar, 165

Gedit, editor, características, 248

Gentoo Linux, URL, 5

-geometry, argumento, usar con X Window System, 155

Gestor de energía de GNOME, características, 545

get, comando, usar con el programa ftp, 294, 296

getatsc, comando, usar, 56

getty, programa, usar, 661

GFS (Global File System), características de, 779-784

gFTP, cliente
 características del, 292-293
 descripción del, 291

.gif, tipo de archivo Web, descripción, 283

GIMP (GNU Image Manipulation Program),
 características, 257

gkc (configuración de kernel de GTK), herramienta,
 usar, 678

- GKrellM, monitores del sistema, características, 545-546
- Global File System (GFS), características, 779-784
- globales y exportadas, diferencia entre variables de entorno, 76
- GNOME (GNU Network Object Model Environment)
 - aplicación VoIP Ekiga, 304-305
 - capacidades de, 9
 - características, 169-170
 - configurar, 193
 - configurar CUPS, 505
 - configurar fuentes, 26
 - configurar redes, 710
 - configurar sesiones, 27
 - explorador Ayuda, 173-174
 - herramienta Preferencias de temas, 24-25
 - herramienta services-admin, 410
 - incluir directorios, 193
 - inicio de programas, 173
 - permisos, 562-563
 - preferencias de información personal, 26
 - salir, 173
 - seleccionar fotografías personales, 26
 - seleccionar temas, 193
 - usar el selector, 20
- GNOME, administrador de archivos. *Véase* Nautilus
- GNOME, aplicaciones
 - configuración, 195
 - descarga, 11
 - usar el widget GTK+, 171
- GNOME applets. *Véase también* applets
 - agregar, 191
 - características, 23-24, 191-192
 - Lista de ventana, 192
 - Selector de áreas de trabajo, 192
- GNOME, archivos binarios, ubicación, 193
- GNOME, bibliotecas, ubicación, 193
- GNOME, directorio DESKTOP, contenido, 194
- GNOME, directorio home, contenido, 194
- GNOME Display Manager (GDM). *Véase* GDM (GNOME Display Manager)
- GNOME, escritorio
 - advertencia acerca de eliminar iconos, 174
 - aplicaciones, 175
 - arrastrar y soltar archivos, 174-175
 - cliente de correo Evolution, 267-268
 - clientes de correo, 269-270
 - copiar archivos en, 174
 - crear vínculos, 174-175
 - crear y editar archivos MIME en, 267
 - desplegar con diferentes temas, 172
 - desplegar el contenido de archiveros tar, 134
 - herramienta Buscar, 125
 - menús, 22-23
 - mover y copiar archivos, 172
 - obtener documentación, 14
 - usar con administradores de ventana, 175-176
- GNOME, explorador de Ayuda, iniciar, 28
- GNOME, exploradores Web, disponibilidad, 286
- GNOME, interfaz
 - componentes de, 171-173
 - paneles en, 172
- GNOME Keyring Manager, sitio Web, 317
- gnome-luks-format, herramienta, cifrar sistemas de archivos, 326
- GNOME, menú de escritorio, desplegar, 175
- GNOME, objetos de panel
 - agregar, 189
 - mover, eliminar y bloquear, 189
 - objeto Bloquear, 191
 - objeto Lanzar, 191
 - objeto Salir, 191
 - usar lanzadores de aplicación con, 189-190
- GNOME Office
 - aplicaciones en, 242-243
 - sitio Web, 12, 238
- gnome-nettool, utilería, características, 301
- GNOME, paneles
 - agregar cajones, 190-191
 - agregar carpetas, 191
 - agregar carpetas y lanzadores de archivos, 190
 - agregar menús, 191
 - agregar paneles, 187
 - cambiar colores de fondo, 188
 - características, 187
 - configurar, 187-188
 - desplegar, 187
 - diferencia entre paneles expandidos y no expandidos, 188
 - mover y ocultar, 188
- GNOME, sitio Web de desarrolladores, URL, 13
- GNU, archivero, URL, 11
- GNU General Public License, significado de, 9
- GNU Image Manipulation Program (GIMP), características, 257
- GNU Java Compiler, sitio Web, 288
- GNU, Linux como software, 10
- GNU Network Object Model Environment (GNOME). *Véase también* GNOME (GNU Network Object Model Environment)
- GNU, páginas de información, acceso, 29

GNU SQL, base de datos, características y sitio Web, 12, 246-247

gnubiff, herramienta de notificación de correo, características, 273

GnuCash, aplicación de finanzas, disponibilidad, 244

GnuPG (GNU Privacy Guard). *Véase también* cifrado

- características de, 316-317
- cifrar mensajes en, 321
- descifrar firmas digitales en, 322
- descifrar mensajes en, 321-322
- firmar mensajes, 322
- generar claves privadas y públicas, 318
- hacer disponibles las claves públicas, 318-319
- obtener claves públicas, 319-320
- proteger claves, 318
- revisar huellas en, 320
- usar, 321
- validar claves en, 320

gpg, comando, usar, 321, 324

gPhoto, sitio Web para el proyecto, 257

Grand Unified Bootloader (GRUB). *Véase también* GRUB (Grand Unified Bootloader)

grep, comando

- usar con el comando ps, 543
- usar con la instalación de kernel, 673
- usar con lista de procesos, 405

groupadd, comando, usar, 561

groupdel, comando, usar, 561

groupmod, comando, usar, 561

growisofs DVD+RW, herramienta, características, 613

GRUB (Grand Unified Bootloader)

- características, 547-550
- configurar para kernel, 684
- reinstalar, 18
- usar con paquetes de kernel, 675-676

grupo, crear directorios, 560-561

grupos

- administración, 559-561
- cambio de propietario, 565

grupos de noticias

- disponibilidad, 15
- para Mutt, 271
- temas, 276

grupos de red, configurar con NIS, 774

gshadow, archivo de contraseña, función, 554

gStreamer

- configurar, 260
- descargar módulos y plug-in, 11, 259
- instalar soporte a MP3, 260
- plug-ins, 260

.gtckrc, archivo, descripción, 194

gThumb, visor de imágenes en miniatura, características, 257

GTK, selección de temas, 172

GTK+ widget, usar con aplicaciones de GNOME, 171

gtKam, herramienta gráfica, descargar, 257

GUI (interfaz gráfica de usuario)

- forzar salida de, 19
- inicio de la línea de comandos, 24

GUI, herramientas de administración, disponibilidad, 551-552

GUI, sesiones, administrar con GDM, 164-166

GView, herramienta gráfica, descargar, 257

gvim, editor, características, 253

.gz, extensión, significado, 137-138, 220

gzip, utilería

- comprimir archivos con, 138-140
- usar la opción -d con, 139

H

HAL (capa de abstracción de hardware)

- función de, 652-653
- implementación, 653
- relación con archivos de dispositivo de impresora, 504
- y fstab, 596
- y udev (dispositivos de usuario), 590-592

HAL, llamadas, función, 657

HAL, vínculos, revisar dispositivos CD/DVD, 261

hald, daemon, función, 653

halt, archivo, contenido, 401

halt, comando, usar, 21

HDTV, soporte a recepción, 263-264

Herramienta de configuración de Apache,

- características, 463-464

herramientas de administración de fotografías,

- disponibilidad, 256

herramientas de administrador de servidor,

- disponibilidad, 536

herramientas de monitoreo de red

- netstat, 742
- programa olfateador Ettercap, 739
- programa ping, 739
- tcpdump, 741
- Wireshark, 739-741

herramientas de notificación, usar para correo, 273-274

herramientas de red

- comando finger, 301-303
- comando host, 301-303
- comando ping, 301-303
- comando traceroute, 301-304
- herramientas de red Kerberizadas, 371

- utilería gnome-nettool, 301
 - herramientas gráficas, administración de fotografías, 256
 - hipertexto, base de datos, componentes, 281
 - historial (!), comando
 - edición en la shell C, 61
 - referencia de eventos con, 41-12, 44, 59-60
 - usar, 40-41
 - usar en la shell Z, 63
 - historial, editar en la shell TCSH, 62-63
 - historial, listas de
 - despliegue de comandos con, 40-43
 - usar metateclas, 41
 - historial, sucesos
 - configurar, 43
 - crear referencias en la shell BASH, 40-42
 - crear referencias en la shell Z, 63
 - editar, 42-43
 - seguimiento, 109
 - history, variable, usar en la shell TCSH, 111
 - HISTSIZE, variable, configuración predeterminada, 43
 - home, directorio
 - búsqueda, 102
 - en FHS, contenidos de, 588
 - enumerar archivos de configuración, 90
 - nombre, 118
 - ubicación, 118
 - HOME, variable, contenido, 96-97
 - \$HOME/*, archivos, descripciones, 364
 - host
 - configurar directivas, 353
 - respalda con Amanda, 695-698
 - host de recepción, configurar directivas, 353
 - host, forma de la dirección, 478
 - host, herramienta de red de, 301-303
 - hosts.conf, archivo, contenido, 726-727
 - host virtual
 - dinámico, usar con Apache, 459-462
 - en el servidor Web Apache, 458-462
 - relacionar a vsftpd, 434-435
 - ht://Dig, servidor de búsqueda e indización, características, 520
 - .htaccess, archivo
 - contenido de, 449
 - usar para configuración en nivel de directorio de Apache, 452-153
 - .html, tipo de archivo Web, descripción, 283
 - HTTP (protocolo de transferencia de hipertexto), función en URL, 282
 - http_access, opciones, usar en Squid, 472
 - httpd, proceso, revisión, 405
 - httpd, secuencia de comandos de servicio, ejemplo, 412, 414-415
- | —
- IBM, base de datos, características de y sitio Web para, 246-247
 - IBM DB2, software de la base de datos, URL, 12
 - ICMP (protocolo de mensajes de control de Internet), habilitación de paquetes, 381
 - ICMP, paquetes, controlar la secuencia de comandos mifiltro, 391-392
 - ICP (protocolo de caché de Internet), usar con Squid, 473-474
 - ICQ, características, 305
 - id, comando, determinar el contexto de seguridad, 330
 - identidades, función en SELinux, 329-330
 - identificador de terminal (TTY, terminal identifier), significado, 55-56
 - if, estructura de control condicional
 - efecto, 78-81
 - expresión de prueba utilizada, 82
 - ifconfig, comando, usar, 729-731
 - if-endif, estructura de control, usar TCSH, 84
 - ifls, secuencia de comandos, 84-85
 - if-then, efecto del comando, 78-80
 - if-then, estructura, usar en TCSH, 84-85
 - ignoreeof, característica
 - descripción, 92
 - usar con la shell TCSH, 108
 - IKE, configuración de Ipse con, 354-355
 - ImageMagick, programa, características, 257
 - imagen de espejo, nivel de RAID, función, 628
 - IMAP (protocolo de acceso a correo de Internet), función de, 498-499
 - IMAP, servidores, disponibilidad, 499-500
 - imposición de tipo (TE), usar en SELinux, 328
 - impresión
 - archivos, 53,121
 - nombres de archivo, 50-51
 - impresión, servicios, usar CUPS, 503-504
 - impresora, controladores, descargar, 504
 - impresoras
 - instalación con CUPS, 505-507
 - remotas, configuración en CUPS, 507
 - imprimirarchivo, secuencia de comandos, 75-76
 - In, comando
 - usar, 130-132
 - usar la opción -s con, 131
 - Includes del lado del servidor (SSI). Véase también SSI (Includes del lado del servidor)

- indicador de línea de comandos, creación, 109
- indicadores de comandos
 - códigos, 100
 - configurar, 99-100
 - crear, 109
 - usar con shell, 36
- InfiniBand, soporte, 743
- info, comando, usar, 29
- información
 - acceso a páginas de, 29-30
 - de acceso remoto, obtención, 308
 - de actividad del sistema, despliegue, 544
- información del sistema, desplegar con el comando `uptime`, 308
- información personal, configuración, 26-27
- Informix, base de datos, características y sitio Web, 246-247
- inicio de sesión, configuración de la shell, 101
- inicio de sesión anónimo para sitios FTP, 423
- inicio de sesión remoto, realización, 309
- inicio y salida de sesión, 20-21
- init, secuencia de comandos, funciones, 413
- init.d, archivos, 402
- init.d, secuencias de comandos de servicio, usar, 403
- initab, niveles de ejecución, 533
- Inkscape, aplicación gráfica de vectores, características, 257
- INN (InterNetNews), servidor de noticias
 - acceso al lector de noticias, 514
 - archivo `inn.conf`, 513-514
 - archivos de configuración, 513-514
 - características, 513
 - formatos de almacenamiento, 514
 - implementación, 515
 - soporte de resúmenes, 514-515
- INPUT, cadena
 - definir la cadena de usuario, 392-393
 - en la secuencia de comandos `mifiltro`, 394
 - evaluación, 380
 - usar, 376-377
- `.inputrc`, creación del archivo, 37
- inserción activa, explicación de dispositivos, 641
- INSERT INTO, comando, usar, 517
- insmod, comando, cargar módulos, 667
- instalación, preparación, 18
- instalaciones en el daemon `syslogd`, descripciones de, 539-540
- instances, atributo, usar en servicios `xinetd`, 417
- intercambio de correo (MX), registros, relacionar con servidores de correo, 478-479
- interfaces gráficas de usuario (GUI)
 - forzar la salida, 19
 - inicio desde la línea de comandos, 24
- interfaces y enrutadores de red, revisión general, 729-733
- interfaz común de puerta de enlace (CGI), archivos, usar con el servidor Web Apache, 455
- interfaz de bucle cerrado, relación con servidores de correo, 478
- Internet Software Consortium, URL, 12
- Internet, servidores, disponibilidad, 12-13
- iostat, efecto del comando, 544
- IP, activar el reenvío, 396
- IP, administración de reglas de tabla, 375
- IP Chains, paquete, características, 374
- IP, direcciones
 - advertencia con `radvd`, 749
 - determinar para sitios remotos, 303
 - obtener, 717-718
 - y host virtual dinámico, 461-462
- IP, enmascaramiento
 - implementar, 395
 - implementar en la secuencia de comandos `mifiltro`, 391
 - implementar con filtrado de red, 355
 - y reglas NAT, 396
 - de hosts seleccionados, 396-397
 - en Sendmail, 491-493
- IP Tables, URL para el firewall, 12
- ip6tables, paquete, características, 374-375
- IPcon, host virtual, usar con el servidor Web Apache, 459, 461
- IPsec
 - bases de datos de seguridad, 350
 - cifrado de puertas de enlace con, 356-357
 - configuración de conexiones, 351-353
 - configurar con la herramienta `racoon`, 354-355
 - configurar conexiones de dos vías seguras, 353
 - detención de paquetes, 355
 - habilitar en el kernel, 351
 - herramientas, 351
 - modo túnel, 356-357
 - modos de, 350
 - protocolos utilizados en, 349-350
 - recepción de transmisiones, 352-353
 - usar certificados, 355
- IPtables
 - control de acceso de puerto, 382
 - implementar enmascaramiento IP, 395
 - opciones, 379
 - y cadenas definidas por el usuario, 380-381
 - y enmascaramiento IP, 396

iptables, comando
 administración de reglas de tabla IP, 375
 agregar y modificar reglas de cadena, 376-378
 ejecutar, 374
 funciones, 377
 nombres de cadena, 375
 IPtables, efectos del filtrado de red, 355
 IPtables, expansibilidad del paquete, 387
 IPtables módulos, ubicación en Netfilter, 375
 IPv4, direccionamiento CIDR, función, 715-717
 IPv4, direcciones de red, partes de, 712
 IPv4, direcciones IPv4 dinámicas para DHCP, 754
 IPv4, direcciones reservadas, características, 717-718
 IPv6 e IPv4, categorías de métodos coexistentes, 723
 IPv6, configuración automática con estado, implementación, 748-749
 IPv6, configuración automática sin estado
 capacidades de, 745-746
 generar direcciones completas, 746
 generar direcciones locales, 746
 y reenumeración de enrutador, 746-747
 IPv6, direccionamiento CIDR, función, 717
 IPv6, direcciones
 formato de, 720-721
 identificadores de interfaz, 721
 tipos de, 721-723
 IPv6, enrutador, Linux como, 749
 IPv6, ventajas de, 711
 IRC (charla de retransmisión en Internet), características de, 305
 ISO, imágenes de distribución, descargar con BitTorrent, 220-221
 itpaccess, archivos, usar con ProFTPD, 436-438
 iwconfig, comando, usar, 735
 iwlist, herramienta, características, 736
 iwpriv, comando, usar, 735
 iwspy, herramienta, características, 736

J

Jakarta Project, sitio Web, 288
 Java, aplicaciones
 descarga, 289
 disponibilidad, 288
 Java, obtener puertos Linux, 287
 Java 2 Software Development Kit (SDK)
 descarga, 287
 herramientas, 289-290
 Java Runtime Environment (JRE)
 habilitar para Mozilla y Firefox, 289
 instalar, 289

jerarquía de sistema de archivos (FHS). *Véase también* FHS (jerarquía de sistema de archivos)
 jobs, comando, usar, 53-54
 JPackage Project, sitio Web, 287
 JPEG, cifrar y recuperar datos en imágenes, 323
 jpeg, tipo de archivos Web, descripción, 283
 J-Pilot, características, 245
 JRE (Java Runtime Environment)
 habilitar para Mozilla y Firefox, 289
 instalar, 289
 juegos, descargar, 11

K

K Desktop, administrador de archivos, capacidad FTP, 292
 K Desktop, ventana del administrador de archivo, usar como explorador Web, 285-286
 K Desktop Environment (KDE). *Véase también* KDE (K Desktop Environment)
 .k5login, acceso a cuentas con el archivo, 308
 kadmin, herramienta, usar, 372
 Kate, editor, características, 248-249
 KDC (centro de distribución de claves), función en Kerberos, 369-370
 KDE (K Desktop Environment)
 acceso a recursos del sistema en, 202-203
 cambiar escritorios virtuales, 205
 cambiar opciones, 201
 cambiar usuarios, 20
 capacidades, 9
 características, 22, 197-198
 configuración de métodos abreviados de teclado, 23
 configurar, 203
 configurar CUPS, 505
 configurar el ratón, 23
 configurar fuentes, 26
 configurar redes, 710
 configurar sesiones, 27
 configurar y administrar acceso, 199
 contenido del directorio compartido, 216
 copiar archivos, 23
 crear archivos de escritorio URL, 203-204
 crear directorios, 202
 crear y editar tipos MIME, 267
 despliegue de contenidos de archiveros tar, 134
 despliegue de directorios seleccionados, 208
 despliegue de medios de almacenamiento, 202
 directorios y archivos, 216
 entrada Vincular a aplicación, 203-204
 Gestor de temas, 25

- iniciar, 19-20
- iniciar aplicaciones, 208
- lanzar aplicaciones, 203
- montaje de dispositivos, 208
- mover ventanas, 23
- permisos, 563
- registrar tipos de archivos, 215
- salir, 201
- usar el administrador de archivos, 202-203
- usar el tipos MIME, 215
- y la biblioteca Qt, 197-198
- KDE, acceso a funciones de, 205-206
- KDE, administrador de archivos
 - abrir archivos, 210
 - acceso a Web y FTP, 213
 - archivos .directory, 212
 - buscar archivos, 211
 - características, 208-209
 - configurar, 213-214
 - configurar el panel Navegación, 210-211
 - copiar, mover, eliminar y cambiar nombres de archivos, 212
 - despliegue de la ventana Propiedades para archivos, 212
 - directorios de navegación, 211-212
 - extraer archiveros tar, 210
 - realizar operaciones de vinculación, 212
- KDE, aplicaciones
 - acceso, 201
 - asociar documentos, 207-208
 - configurar permisos, 207
 - crear archivos de escritorio, 207
 - iniciar, 207
- KDE, applets, características, 24
- KDE, bibliotecas de desarrolladores, URL, 13
- KDE, centro de ayuda, características, 206-207
- KDE, componentes, ubicar archivos de configuración, 215
- KDE, depósito de software, URL, 11
- .kde, directorio, contenido, 215
- KDE Display Manager (KDM). *Véase también* KDM (KDE Display Manager)
- KDE, iniciar el explorador de ayuda de, 28
- KDE, interfaz, configurar el comportamiento, 214
- KDE, opciones en menús de, 200-201
- KDE, operaciones del escritorio
 - administrar CD-ROM, DVD y discos flexibles, 201
 - desplegar el directorio de inicio en el panel, 201
 - eliminar archivos, 201
- KDE, paginador de escritorio, características, 205
- KDE, panel
 - agregar aplicaciones a, 206
 - applets y extensiones de panel, 206
 - botones desplegados en, 202
 - configurar la posición y el comportamiento de, 206
 - ubicación de, 205
- KDE, ventana del administrador de archivos, componentes, 209-210
- KDE, ventanas, usar, 204-205
- kdestroy, comando, usar con boletos de Kerberos, 371
- kdetv, sitio Web, 262
- KDM (KDE Display Manager)
 - características, 166-167
 - disponibilidad, 160
 - función, 19-20
- KEdit, editor, características, 249-250
- Keep Alive, directiva, usar con el servidor Web Apache, 449
- KeepAliveRequests, directiva, usar con el servidor Web Apache, 449-450
- kerberizados, configurar para utilizar los servicios, 371
- Kerberos
 - administración de contraseña, 370
 - descripción, 368-369
 - listas de boletos, 369
 - proceso de autenticación, 369-371
 - sitio Web, 369
- Kerberos, servidores
 - configurar, 371-372
 - función de, 369
- kernels
 - advertencia acerca de modificaciones, 675-676
 - colocar en discos de arranque, 683
 - compilar del código fuente, 676-679
 - compilar e instalar, 681-683
 - configurar, 677
 - configurar GRUB, 684
 - configurar herramientas, 677-679
 - definición, 8, 671
 - determinar la versión, 672
 - distribuir, 4
 - habilitar IPsec, 351
 - instalar módulos, 670
 - instalar versiones nuevas, 673-674
 - números menores y mayores, 671
 - parámetros modificables en tiempo de ejecución, 673
 - referencias, 672

- retener copias, 675
 - ubicación, 674
 - URL, 5
 - versiones de, 671-672
 - Kernel-based Virtualization Machine (KVM),
 - características, 687-688
 - kernel, descarga de paquetes RPM, 673
 - kernel, descargar e instalar fuentes, 677
 - kernel, entradas para sistemas de archivos de
 - interfaz, 601
 - kernel, instalación manual de la imagen, 682-683
 - kernel, módulos
 - herramientas, 664-665
 - ubicación, 665
 - kernel, paquetes
 - instalar, 674
 - utilizar el cargador de arranque GRUB, 675-676
 - Kicker. *Véase también* KDE, panel
 - kill, comando
 - cancelar trabajos, 55
 - referencia de números de proceso de sistema, 56
 - usar, 53
 - killall, comando, usar con xinetd, 407
 - killproc, función, efecto, 412
 - kinit, comando, usar, 370-371
 - KJots, características del editor, 248-249
 - klist, comando, usar, 369
 - KMail, características del cliente de correo, 270
 - Kmod (cargador de módulo de kernel), función de, 664-665
 - Knemo, configuración del monitor de dispositivo de
 - red, 200
 - Knoppix Linux, URL, 5
 - KOffice, disponibilidad de aplicaciones, 241-243
 - KOffice, paquete para KDE, sitio Web, 238
 - KOffice, software, URL, 12
 - Konqueror, cliente FTP, descripción, 291
 - Konqueror. *Véase también* KDE, administrador de
 - archivos
 - Korn, shell
 - archivos de configuración, 91
 - características, 89
 - sitio Web, 36
 - KParts, implementación, 242-243
 - kpasswd, comando, usar, 370
 - krb5.conf, archivo, contenido, 371
 - KSysguard (Administrador de tareas y monitor de
 - rendimiento de KDE), características de, 546
 - KUser, herramienta, características, 552
 - KView, visor de imágenes, características, 257
 - KVM (Kernel-based Virtualization Machine),
 - características, 687-688
- L**
- l, opción, usar con comandos, 35
 - LAME, evolución, 261
 - lanzadores de aplicación
 - usar con Nautilus, 184
 - usar con objetos de panel de GNOME, 189-190
 - LDAP (protocolo ligero de acceso a directo)
 - buscar, 579
 - características, 573
 - habilitar en Thunderbird, 580
 - soporte en Sendmail, 485, 487
 - y módulos PAM, 580
 - y NSS (Name Service Switch), servicio, 580
 - LDAP, archivos de configuración, ubicación, 574
 - LDAP, base de datos de directorio
 - atributos y clases de esquema, 575-576
 - distinguir nombres, 576-577
 - entradas LDIF, 577-578
 - LDAP, configuración del servidor, 574-575
 - LDAP, directorio, implementar, 573
 - LDAP, herramientas, disponibilidad, 579
 - LDIF (formato de intercambio de LDAP), archivo
 - agregar registros, 578-579
 - contenido, 577-578
 - LDP (Linux Documentation Project), documentos
 - disponibles en, 13-14
 - Leafnode, características del servidor de noticias, 515
 - lectores de noticias, lectura de artículos Usenet con, 277
 - lectura (r), permisos
 - ejecución, 566
 - explicación, 562
 - lenguaje de consulta SQL, características del, 516-517
 - less, comando
 - usar, 120-121
 - usar con consultas RPM, 225
 - lftp, características del programa, 298
 - lftp, descripción del programa, 291
 - LGPL (licencia pública general menor), significado, 9
 - /lib/modules, contenido del directorio, 665
 - licencia pública general menor (LGPL), significado, 9
 - licencia pública Qt (QPL), significado, 10
 - licencias públicas, protección de software de fuente
 - abierta por, 9-10
 - licencias, protección de software de fuente abierta
 - por, 9-10
 - lighthttpd, descripción de, 444

Limit, directiva, usar en el servidor Web Apache, 453
línea de comandos

- acceso, 35
- acceso a Linux, 20-22
- analizar argumentos en, 68
- apagar Linux de, 21-22
- cambiar a, 19,160
- ejecutar, 44
- iniciar GUI de, 24
- iniciar Linux de, 167-168
- usar, 27-28

linux-wlan, propósito del proyecto, 737

Linux

- acceso a archivos compartidos, 31
- acceso desde la interfaz de línea de comandos, 20-22
- apagar desde la línea de comandos, 21-22
- comparación con Unix, 4
- componentes, 3
- distribuciones, 4-5
- historia, 7-8
- inicio desde la línea de comandos, 167-168
- paquetes de oficina disponibles, 11
- revisión, 8-9
- software de base de datos disponible, 11
- terminar sesiones, 21

Linux de modo de usuario (UML), función, 674

Linux Documentation Project (LDP), documentos disponibles, 13-14

Linux Foundation, acceso al sitio Web, 8

Linux Game Tome, URL, 11

Linux, sistemas

- acceso, 19-22
- apagar y reiniciar, 19-20
- configuración de archivos compartidos, 31

Linux, software. *Véase también* software

- aplicar niveles RAID, 627-628
- descarga, 11
- disponibilidad, 10-13

LinuxTV.org, vínculos, 262

listas de correo, suscripción, 275

listas grises, usar en Postfix MTA (agente de transferencia de correo), 482-483

Listen, directiva, usar con el servidor Web Apache, 450,459

llamados, relación con HAL, 657

llaves ({}), usar con patrones, 47

LoadModule, directiva, usar con el servidor Web Apache, 450

localizador universal de recursos (URL), componentes, 282

LogFormat, directiva, usar con el servidor Web Apache, 457

.login en TCSH, archivo de inicialización
descripción, 111
función, 112

logout, comando
usar, 21, 94
usar en la shell TCSH, 108

.logout en TCSH, archivo de inicialización
descripción, 111
función, 113-114

logresolve, utilería, usar con los archivos de registro de Apache, 458

lpadmin, herramienta de línea de comandos CUPS, características, 506, 511

lpinfo, herramienta de línea de comandos CUPS, características, 512

lpnum, archivo ejecutable, 71-72

lptions, herramienta de línea de comandos CUPS, características, 511-512

lpq, comando, usar, 121

lpq y lpstat, clientes de impresión, usar con CUPS, 510

lpr, cliente de impresión, usar con CUPS, 509

lpr, comando
aplicar, 119,121
usar, 51, 53

lprm, cliente de impresión, usar con CUPS, 510

lprm, comando, usar, 121

ls, comando
renombrar, 92
usar, 37, 119, 121, 123

ls -l, comando, usar, 116

lsuf, comando, usar, 603

Lugares, menú en GNOME, capacidades, 22-23

Luks, cifrado de sistemas de archivos, 326

lvcreate, comando, usar, 620

LVM (administrador de volumen lógico)
device-mapper utilizado por, 621-622
diferencia entre RAID, 616
ejemplo, 623-624
estructura, 616-617
función, 615
herramientas de configuración de distribución, 617
reemplazar unidades con, 622-623

LVM, dispositivos, nombres, 621-622

LVM durante la instalación, creación, 617

LVM, grupos de volumen, activación de grupos de volumen, 620

LVM, grupos, administración, 619-620

LVM, herramientas, usar comandos, 617-622
 LVM, información, despliegue, 619
 LVM, instantáneas, creación, 625
 LVM, volúmenes físicos, administración, 619
 LVM, volúmenes lógicos, administración, 620-621
 Lynx, explorador de modo de línea, características, 286

M

mail, acceso en servidores de correo POP remotos, 274-275
 Mail Notification, características de la herramienta, 273
 Mail, programa, método abreviado de teclado, 94
 Mail, utilidad de cliente de línea de comandos, características, 272
 mail, variable, usar en la shell TCSH, 111
 MAILER, macro, usar con Sendmail, 488,490
 mailer, tabla, usar con Sendmail, 495
 Mailman, administración de listas de correo, 275
 .mailrc, archivo de inicialización, 273
 Majordomo, sitio Web, 275
 Make, comando, usar con kernels, 681-682
 make policy, comando, usar en SELinux, 344
 MAKEDEV, creación de archivos de dispositivo con el comando, 657-658
 Makefiles
 editar, 233
 utilizar con secuencias de comandos de configuración, 231
 Man, acceso a páginas, 29
 man, comando, usar, 29
 mangle, tabla
 destinos, 386
 identificar, 386
 relacionar con filtrado de paquete, 375
 máscaras binarias, usar con permisos absolutos, 566-568
 máscaras de red, relación con direcciones de red TCP/IP, 713-714
 MASQ, compuerta, explicación, 395
 MaxClients, directiva, usar con el servidor Web Apache, 451
 MaxDB, base de datos, características y sitio Web, 246-247
 MaxRequestsPerChild, directiva, usar con el servidor Web Apache, 451
 mayor que (>), carácter
 como símbolo de shell, 44
 función, 68
 redirigir la salida estándar, 48-50
 MCS (seguridad de varias categorías), extensión de SELinux con, 331-332, 336
 MD, controlador, usar con dispositivos RAID, 629
 MDA (agentes de entrega de correo), características de, 477
 mdadm.conf, opciones, 631
 mdadm con Multipath, usar el daemon, 628
 mdadm
 opciones -create, 633
 realizar administración RAID con, 629
 Mdir, comando, usar con discos de MS-DOS, 133
 /media en FHS, contenido del directorio, 587
 medios extraíbles, administrar con el Administrador de volúmenes de GNOME, 176-178
 menor que (<), carácter
 como símbolo de shell, 44
 función, 68
 mensajes de error, colocación en error estándar, 51-52
 menuconfig, herramienta, usar con kernels, 677-678
 Mepis Linux, URL, 5
 meta, teclas, usar con listas de historial, 41
 método binario, usar con permisos de propietario, 569
 métodos abreviados de teclado. *Véase también* ALT, tecla; CTRL, tecla
 comando exit, 525
 configurar en KDE, 23
 editar historial de eventos, 42
 eliminar líneas de comandos, 28
 eliminar palabras, 57
 interfaz de línea de comandos, 19,160, 524
 interrumpir comandos, 36
 la shell Z, 63
 menú principal de KDE, 200
 mover y editar comandos, 38
 navegar por el escritorio KDE, 205
 navegar y editar caracteres, 37
 programa Mail, 94
 reiniciar, 22
 suspender trabajos, 55
 trasponer caracteres, 37
 mget, comando, usar con el programa ftp, 296
 Microsoft Office, ejecución en Linux, 238-239
 mienumerarg, secuencia de comandos, 87
 mifiltro, secuencia de comandos IPtables
 acceso al servidor, 390-391
 acceso externo al firewall, 391
 bloqueo del acceso iniciado en el exterior, 391
 configurar la directiva DROP, 389-390
 configurar una LAN, 394

- control de paquetes ICMP, 391-392
- crear, 387-389
- implementar enmascaramiento, 391
- lista de reglas, 392
- permitir acceso a redes locales, 391
- proteger contra engaño de IP, 390
- reglas definidas por el usuario, 392-393
- milista, secuencia de comandos, 86
- MIME (extensiones de correo de Internet de propósitos múltiples)
 - asociaciones estándar, 267
 - características, 266
- MIME, protocolos, usar con clientes de correo, 267
- MIME, tipos
 - crear y editar en GNOME y KDE, 267
 - y el servidor Web Apache, 454-455
- mime.types, entradas en el archivo, 266-267
- mingetty, usar el programa, 661
- Minix, significado del programa, 7
- miprogarch, secuencia de comandos, 138
- misargs, secuencia de comandos, 73
- mkdir, comando
 - usar, 122-123
 - usar con el programa ftp, 294
- mkfs, herramienta, usar con sistemas de archivos, 609-610
- mkinitrd, comando, crear archivos de imagen de disco RAM, 685
- mkisofs, comando, usar con CD y DVD, 611-612
- mknod, comando, crear dispositivos, 659-660
- mkswap, herramienta, usar con sistemas de archivos, 610
- MLS (seguridad de varios niveles), extensión de SELinux con, 331-332, 336
- /mnt, directorio, contenido en FHS, 587
- .mod, archivos de módulo, crear en SELinux, 341
- módems
 - archivo de vínculo simbólico, 646
 - instalar y administrar, 660-661
- modinfo, comando, parámetros de módulo de descubrimiento, 666
- modo de túnel, usar con IPsec, 356-357
- modprobe, comando
 - instalar módulos, 666-667
 - usar con el comando depmod, 666
 - usar en el rastreo de conexión, 384
- modprobe, configuración, acciones asociadas, 667-668
- mod_ssl, implementación de SSL
 - características de, 465
 - configurar, 466
- módulo, dependencias, detección, 666
- módulos de multiprocesamiento (MPM),
 - configuración para Apache, 450-451
- módulo RAM, discos, usar, 684-685. *Véase también* RAM (memoria de acceso directo)
 - módulos
 - administrar con modprobe, 666
 - cargar, 667
 - compilar con archivos de encabezado de kernel, 670
 - descargar, 667
 - descargar controladores para la instalación, 669
 - herramientas de módulos kernel, 664-665
 - instalar el kernel, 670
 - usar de, 664
- Monitor del sistema de GNOME, características, 543
- Mono, soporte de .NET proporcionado por, 613-614
- more, comando, usar, 51, 120-121
- mount, comando
 - usar con NFS (sistemas de archivos de red), 769
 - usar con sistemas de archivos, 596, 602-603
- movimiento de archivos, 129
- Mozilla, cliente de correo Thunderbird ofrecido por el proyecto, 268-269
- Mozilla, habilitar JRE (Java Runtime Environment), 289
- Mozilla, marco estructural, usar, 283-284
- MP3, opción alterna, 261
- MP3 gstreamer, descarga del plug-in, 11
- .mpeg, tipo de archivo Web, descripción, 283
- MPlayer, características, 262-263
- MPM (módulos de multiprocesamiento)
 - configurar para Apache, 450-451
 - usar con Apache, 447
- mput, comando, usar con el programa ftp, 296
- MS-DOS, acceso a discos, 132-133
- MTA (agentes de transferencia de correo)
 - características, 477-478
 - definición, 477
- mttools, usar, 132-133
- mtr, características de la herramienta de red, 302
- MUA (agente de correo de usuario), definición, 477
- MULTICS (servicio multiplexado de información y computación), significado, 6-7
- multimedia, aplicaciones, 258
 - gStreamer, 258-260
 - quemadores y extractores de CD, 261
 - sonido, 260-261
 - video, 262-264
- Multirruta, relacionar con niveles RAID, 628
- música, archivos, compresión, 261

Mutt, acceso al grupo de noticias, 271
 Mutt, características del cliente de correo de línea de comandos, 271
 mv, comando
 significado, 124
 usar, 129
 usar con directorios, 129-130
 MX (intercambio de correo), registros, relación con servidores de correo, 478-479
 .my.cnf, archivo, usar, 518-519
 MySQL, base de datos, características y sitio Web, 246
 MySQL, herramientas, disponibilidad, 519
 MySQL, servidor de base de datos
 administrar con mysql y mysqladmin, 519-520
 configuración de usuario, 518-519
 configuración global de, 518
 configurar, 517-518
 estructura, 517
 MySQL, software de base de datos, URL, 12

N

-name, opción, usar con el comando find, 124
 Name Service Switch (NSS). Véase NSS (Name Service Switch)
 NAT (traducción de direcciones de red), explicación de, 384
 NAT, ejecución de tareas de, 374
 NAT, implementar redirección, 386
 NAT, reglas
 agregar, 384-385
 y enmascaramiento IP, 396
 NAT, tabla
 cadenas en, 385
 destinos válidos, 386
 NAT, tipos de cadenas, 376
 NAT, tipos de operaciones, 385
 Nautilus
 agregar emblemas, 181
 agrupar archivos, 183
 cambiar el nombre de archivos, 182-183
 características, 178
 como explorador FTP, 186
 desplegar de archivos y carpetas, 180-181
 directorio pixmaps, 185
 eliminar archivos, 182
 establecer preferencias, 186
 establecer propiedades de despliegue, 186
 explorar en las vistas Spatial y Navegador, 182
 iniciar aplicaciones y archivos, 183-184
 menú Archivo, 182

 usar la opción Abrir con, 183-184
 usar lanzadores de aplicación, 184
 vista Navegador, 180
 vista Spatial, 180
 Nautilus, barra lateral
 panel Información, 180
 vista Árbol, 180
 vista Histórico, 180
 vista Lugares, 180
 vista Notas, 180
 Nautilus, cliente FTP
 características, 292
 descripción, 291
 Nautilus, opciones en el menú, 181
 Nautilus, paneles de propiedades
 panel Abrir con, 185
 panel Básico, 184-185
 panel Emblemas, 185
 panel Notas, 186
 panel Permisos, 185
 Nautilus, ventanas
 acercar o alejar las vistas, 179
 vista Navegador, 178-179
 vista Spatial, 178-179
 NcFTP, cliente, descripción, 291
 NcFTP, programa, características, 299
 NCSA, servidor Web, descripción, 444
 .NET, disponibilidad del soporte, 613-614
 net_traversal, opción, agregar con racoon, 355
 Netfilter, paquete de software
 características, 374
 módulos, 375
 Netfilter, reglas, usar con Servidores NFS, 768
 .netrc, archivo de configuración ftp, usar, 297-298
 Netscape Enterprise Server, descripción, 444
 Netscape, código fuente, disponibilidad, 284
 netstat, programa, características, 742
 newrole, comando, usar en SELinux, 330
 newsbin, programa, instalación, 32
 NFS (sistemas de archivos de red)
 acceso a nivel de usuario, 764
 configurar, 762-766
 iniciar y detener, 762
 montar automáticamente, 768-770
 montar bajo pedido, 770
 montar de forma manual, 769
 NFSv4, 761
 opciones, 764
 seguridad de archivo y directorio, 766
 y daemon NFS, 762
 NFS, control de acceso a servidores, 766-768

- NFS, entradas de host, tipos, 762, 764
- NFS4, configurar ACL con, 766
- NIS (sistema de información de red), desarrollo, 771
 - configurar grupos de red con, 774
 - relación con NFS, 770
- NIS, configuración de clientes, 774-775
- NIS, configurar servidores, 771-773
- NIS, creación de bases de datos, 773
- NIS, definición de dominios, 771-772
- NIS, servidores
 - controlar el acceso, 773
 - especificar archivos compartidos, 772-773
 - establecer opciones, 772
- nivel de ejecución, vínculos, usar con la herramienta update-rc.d, 411
- nivel de franjas de RAID, función, 628
- nivel de paridad de RAID, función, 628
- nivel de paridad distribuida de RAID, función, 628
- nivel lineal de RAID, función, 627
- niveles de ejecución. *Véase también* estados
 - activar o desactivar servicios, 536
 - cambiar con telinit, 149, 162, 533-534
 - detección, 403
 - en initab, 533
 - habilidad, 531-532
 - para Red Hat y Debian, 405
- NNTPSERVER, variable
 - usar, 100-101
 - usar con lectores de noticias, 277-278
- noauto, opción, usar con sistemas de archivos, 601
- noclobber, característica
 - activar, 108
 - descripción, 92
 - usar con la shell TCSH, 109
- noglob, característica
 - descripción, 92
 - usar con la shell TCSH, 109
- nombre de archivo, caracteres especiales para expandir, 49
- nombre de ruta absoluto, explicación del, 119
- nombre de ruta relativo, explicación, 119
- nombres de archivos
 - distinguir de nombres de directorios, 123
 - expandir, 43-47
 - imprimir, 50-51
 - relacionar caracteres en, 44
 - usar expansiones en, 116
- nombres de directorios, distinción de nombres de archivos, 123
- nombres de dispositivo, relación con reglas udev, 643-645
- nombres de host
 - código de indicador de comandos, 100
 - en URL, 282
 - identificar, 723-725
 - representar en la shell BASH, 39
- nombres de ruta. *Véase también* URL, nombres de ruta de directorios, 118-119
 - de variables, 110
 - en URL, 282
 - tipos, 119
- nombres de usuario
 - código de indicador de comandos, 100
 - insertar, 21
 - representar en la shell BASH, 39
- NOT, operadores lógicos, 78
- noticias, servidores
 - características, 512
 - especificar, 100-101
 - Leafnode, 515
 - servidor de noticias INN (InterNetNews), 513-515
- notify, comando, usar con trabajos, 54
- NSS (Name Service Switch)
 - relación con LDAP, 580
 - y DNS (servicio de nombres de dominio), 727-729
- nsswitch.conf, especificar archivos de configuración, 775
- NTFS, montaje de particiones, 239
- ntfs para particiones de Windows, especificar el tipo, 600-601
- nueva línea, código de indicador de comandos, 100
- número (#), símbolo de
 - como indicador de comandos, 21, 36
 - usar con la directiva ServerName directive, 452
 - usar con las etiquetas de secuencia de comandos de servicio, 413-414
 - usar con servicios xinetd, 409-410
- número de historial, código de indicador de comandos, 100
- número de proceso del sistema, usar con trabajos, 53
- número octal, calcular para permisos absolutos, 567
- números de líneas, salida del contenido de archivo, 72

0

- .o, extensión, significado, 665
- o, opción, usar con el comando set, 92
- off, opción, usar con chkconfig, 408
- Office, ejecución en Linux, 238-239
- Ogg Vorbis, sitio Web, 261

- opciones, usar en comandos, 28
- Open Secure Shell, URL, 12
- OpenOffice, software
 - sitio Web, 238
 - URL, 12
- OpenOffice.org, aplicaciones, 239
 - acceso a base de datos, 241
 - herramienta Draw, 240
 - herramienta Math, 240
 - hoja de cálculo Calc, 240
 - procesador de palabras Writer, 240
- OpenPGP Public Keyserver, sitio Web para el proyecto, 319
- OpenPGP, protocolo, usar con clientes de correo, 267
- OpenPrinting, sitio Web para la base de datos, 504
- OpenSSH. *Véase también* SSH (Secure Shell)
 - autenticación, 361
 - cargar claves, 365
 - cifrado, 360-361
 - claves autorizadas, 365
 - entunelamiento, 367-368
 - herramientas, 361-362
 - reenvío de puerto, 367-368
 - sitio Web, 360
- OpenSSL, disponibilidad, 465
- openSUSE Linux, URL, 5
- operaciones lógicas
 - en la shell TCSH, 83
 - realizar con el comando test, 77-78
- operador de asignación (=)
 - usar con servicios xinetd, 418
 - usar con variables shell, 68
- operador de canalización (|)
 - como símbolo de shell, 44
 - conectar comandos con, 51
 - usar con la salida estándar, 49
- operadores en el daemon syslogd, descripción, 539-540
- operadores relacionales, usar en la shell TCSH, 83
- Options, directiva, usar con el servidor Web Apache, 453
- Options, directiva, usar con SSI (Includes del lado del servidor), 462
- OR, operador lógico, 78
- Oracle, base de datos, características y sitio Web, 246-247
- Oracle, software de base de datos, URL, 12
- OSTYPE, macro, usar con Sendmail, 490-491
- OUTPUT, cadena
 - en la secuencia de comandos mifiltro, 394
 - usar, 376-377, 385

P

- palabras
 - cambiar partes, 57
 - eliminar, 57
 - hacer referencia en eventos, 60
- PAM (Pluggable Authentication Modules), usar con LDAP, 580-582
- PAM, archivos de configuración, contenido, 581
- paneles. *Véase* GNOME, paneles
- paquetes
 - aceptar y negar, 379-380
 - cambiar direcciones, 385
 - configurar servicios, 385
 - detectar información de seguimiento, 383
 - habilitar paquetes ICMP, 381
 - rutas, 731
 - seguimiento, 384
- paquetes de controladores, descargar para
 - instalación de módulos, 669
- paquetes de oficina, disponibilidad, 11-12
- paquetes del kernel de CPU, disponibilidad de, 674
- Parallel Virtual File System (PVFS), características, 777-778
- parted, herramienta
 - interfaz GUI disponible, 631
 - usar con sistemas de archivos, 608-609
- particiones
 - crear en diferentes discos duros, 623-624
 - crear para dispositivos RAID, 631
- passwd, comando
 - controlar contraseñas de usuario con, 556
 - descripción, 525
 - uso por el usuario root, 524
- PATH, variables
 - agregar directorios, 102
 - asignación, 101
 - contenido, 98-99, 234
 - crear una entrada de asignación, 234
 - definir, 96
 - personalizar, 234-235
- pathmunge, función, usar con la variable PATH, 234
- patrones, generación, 47
- PCHDTV, tarjeta de video, disponibilidad, 263
- PCMCIA, dispositivos, soporte, 664
- PDA, acceso, disponibilidad, 245
- PDF, acceso, disponibilidad, 244
- Perl, URL, 13
- permisos
 - cambio con el comando chmod, 563
 - categorías, 562
 - en GNOME, 562-563

- en KDE, 563
 - establecer en directorios, 568
 - establecer opciones predeterminadas, 570-571
 - establecer permisos de propietario, 569
 - método absoluto, 566-568
 - permisos de bit adhesivo, 569-570
 - representar permisos vacíos, 562
- permisos absolutos, usar, 566-568
- permisos de acceso remoto, obtención, 308
- permisos de directorio, establecer y desplegar, 568
- PGP (muy buena privacidad), sitios Web, 314
- PGP, administración de claves secretas, 317
- PHP (PHP: Hypertext Preprocessor), características, 463
- PID (Id de proceso), significado, 55-56
- pilot-link, contenido del paquete, 245
- ping, comando
 - características, 739
 - revisar conexiones de sistema con, 301-303
- ping, operaciones, control en relación con paquetes ICMP, 381
- planchado de paquetes, 386
- Pluggable Authentication Modules (PAM), usar con LDAP, 580-582
- .png, tipo de archivo Web, descripción, 283
- POP (protocolo de oficina postal), función, 498-499
- POP, servidores de correo, acceso a correo, 274-275
- POP, servidores, disponibilidad, 499-500
- porcentaje (%), símbolo
 - números de trabajo antes de, 54
 - usar con archivos de registro de Apache, 457-458
- portmapper, servicio, usar con servidores NFS, 767
- Postfix MTA (agente de transferencia de correo)
 - archivo main.cf, 480
 - características, 479
 - comandos, 479-480
 - conexiones directas, 481
 - configurar, 480
 - configurar para SpamAssassin, 501
 - configurar parámetros de red, 480-481
 - control de acceso a usuario y host, 483-484
 - control de clientes, emisores y destinatarios, 483-484
 - dominios y cuentas virtuales, 482
 - en redes locales, 481
 - parámetro masquerade_domains, 482
 - revisiones de encabezado y contenido, 483
 - soporte de listas grises, 482-483
- Postfix, servidor de correo, URL, 12
- PostgreSQL, base de datos, características y sitio Web, 12, 246, 520
- POSTROUTING, cadena, usar, 376, 385
- PostScript, visores, disponibilidad, 244
- PowerDVD, sitio Web, 262
- PPP, acceso, implementar desde la línea de comandos, 737-738
- Preferencias, menú en GNOME, capacidades, 23
- PREROUTING, cadena, usar, 376, 385
- PreSession, directorio, contenido, 166
- principal, directorio. *Véase también* directorios
 - establecer permisos, 568
 - hacer referencia a, 124
 - identificar, 125
- print, opción, usar con el comando find, 124
- prioridades en el daemon syslogd, descripciones, 539-540
- /proc, sistema de archivos, función, 641
- /proc, sistema de archivos, relacionar con FHS, 589
- procesos
 - lista con el comando ps, 543
 - lista, 405
 - terminar, 55-56
- procmail, configuración para utilizar SpamAssassin, 500-501
- profile, secuencias de comandos
 - /etc/login.access, 555-556
 - /etc/login.defs, 555
 - /etc/skel, 555
 - función, 554-555
 - ubicación y contenido, 103
- ProFTPD (daemon FTP profesional)
 - archivo proftpd.conf, 436-438
 - autenticación, 436
 - directivas utilizadas, 437
 - e inicio de sesión como invitado, 439-440
 - instalación e inicio, 436
 - sitio Web, 12
 - usar archivos .ftppass con, 436-438
 - usar la directiva RequireValidShell con, 439
 - y acceso anónimo, 438-440
 - y servidores FTP virtuales, 440-442
- programa, directorios, contenido, 537, 586
- programas, ubicación, 233
- prompt, comando, usar con el programa ftp, 296
- prompt, variables
 - crear indicadores de línea de comando con, 109
 - usar en la shell TCSH, 109-110
- property, entradas, usar con dispositivos, 654-655
- propietario, permisos, establecer, 569
- protocolo de acceso a correo de Internet (IMAP), función de 498-199

protocolo de autenticación de red Kerberos, URL, 12

protocolo de caché de Internet (ICP), usar con Squid, 473-474

protocolo de configuración dinámica de host (DHCP), direcciones dinámicas IPv4, 754-755

protocolo de control de transmisión/protocolo de Internet (TCP/IP), función, 707-709

protocolo de mensajes de control de Internet (ICMP), habilitar paquetes, 381

protocolo de oficina postal (POP), función, 498-499

protocolo de transferencia de archivos (FTP), función de, 423

protocolo de transferencia de hipertexto (HTTP), función en URL, 282

protocolo de transferencia, función en URL, 282

protocolo ligero de acceso directo (LDAP). *Véase también* LDAP (protocolo ligero de acceso directo)

proxies, funciones, 374

proxy, servidores, características, 467-468

ps, comando

- cancelar trabajos con, 55-56
- lista de procesos con, 543

PS1 y PS2, variables, contenido, 99

pub, directorio, contenido, 427

puerta de enlace, direcciones, función, 719

puertas de enlace, cifrar con IPsec, 356-357

puertos

- control de acceso, 382
- especificar para paquetes, 385
- reciclaje, implementar en OpenSSH, 367-368
- seriales, tipos, 660

punto (.)

- archivos, usar, 116
- representar directorios con, 68, 90
- usar con extensiones, 116
- volver a ejecutar la secuencia de comandos . bash_profile con el comando, 103.

punto de montaje, definición, 602

punto y coma (;), separar comandos con, 37, 44

put, comando, usar con el programa ftp, 294, 296

PVFS (Parallel Virtual File System), características, 777-778

pwd, comando, usar, 121-122

Q

-q y -qa, opciones, usar con paquetes RPM, 224-225

qconf, herramientas, usar, 679

Qmail MTA (agente de transferencia de correo), características, 478

QPL (licencia pública Qt), significado, 10

Qpopper, servidor POP, sitio Web, 500

Qt, biblioteca, ventajas, 197-198

.QT, tipo de archivo Web, descripción, 283

quemadores y extractores de CD, disponibilidad de, 261

quota, comando, usar, 572-573

quotacheck, comando, usar, 572

quotaoff, comando, usar, 572

quotaon, comando, usar, 572

R

r (lectura), permiso

- ejecución, 566
- explicación, 562

-r, opción, usar con el comando cp, 130

racoon, herramienta

- configurar conexiones de puerta de enlace con, 357
- configurar IPsec con, 351, 354-355

radvd, advertencia de direcciones IP con, 749

RAID (conjunto redundante de discos independientes)

- configurar, 631-632
- configurar RAID de arranque, 635
- ejemplo, 636-637
- en comparación con LVM, 616
- función, 615, 625-626
- y dmraid, 626-627

RAID, administración, realizar con mdadm, 629

RAID, agregar sistemas de archivos, 630

RAID, conjuntos

- administrar, 634
- crear, 632
- iniciar y detener, 634
- monitorear, 634-635

RAID, dispositivos

- arrancar, 629
- controlador MD utilizado con, 629
- crear e instalar, 630
- crear grupos de repuesto, 633
- crear particiones de disco duro, 631
- crear sistemas de archivos, 633-634
- tipo de partición fd utilizada con, 629
- ubicación de archivos de configuración, 631
- y particiones de disco duro correspondientes, 635-636

RAID, niveles

- aplicar a software Linux, 627-628
- y Multipath, 628

RAID para arranque, configuración de, 635

raíz, ventana en X Window System

- controlar el despliegue, 156
 - establecer características, 160
- RAM (memoria de acceso directo), lista de cantidad, 544. *Véase también* módulo RAM, discos de
- RAM, creación de archivos de imagen de disco, 685
- range, declaración, usar con direcciones IPv4 dinámicas para DHCP, 754
- rar, archivadores, comprar archivador, 133
- rastreo de conexión
 - especialización, 384
 - usar, 383
- ratón, configuración con KDE, 23
- RBAC (control de acceso basado en funciones), modo, usar en SELinux, 328, 336
- rc, secuencia de comandos, función, 403
- rc.sysinit, archivo, contenido, 401
- rcconf, herramienta
 - administración de servicios con, 410-412
 - usar con niveles de ejecución, 536
- RDBMS (sistemas de administración de base de datos relacionales)
 - bases de datos SQL como, 245-247
 - estructura, 516
- re.local, archivo, contenido, 401
- Readline, capacidades de edición proporcionadas por, 37
- recursos de desarrollo, disponibilidad, 13
- recursos en línea, disponibilidad, 13
- red con tcpdump, captura de paquetes, 741
- Red Hat, clave pública, descargar, 323
- Red Hat Global File System, características, 779-784
- Red Hat Linux
 - niveles de ejecución, 405
 - usar la secuencia de comandos virt-install, 690-691
 - sitio Web, 5
- Red Hat Package Manager (RPM)
 - características, 221-222
 - software de empaquetamiento con, 235-236
- Red Hat virt-manager, características, 686-687
- redes, configuración en GNOME y KDE, 710
- redes inalámbricas, con el Administrador de la red, 733-737
- redirección, combinar operaciones, 50
- redirección (>>), operador
 - ejecutar como símbolo de shell, 44
 - usar con la salida estándar, 48-50, 52
- redirección de la salida, proteger archivos, 109
- REDIRECT, destino, usar, 386
- Redirect, directiva, usar con el servidor Web Apache, 454
- refpolicy, archivo fuente de SELinux, contenido, 342
- registro por diario
 - con ext3, 595
 - soporte, 583, 594-595
- registros
 - usar para el acceso de Squid, 474
 - ver, 537
- reglas de cadena, agregar y modificar, 376-378
- reinicio, forzar el, 22
- ReiserFS, registro por diario con, 595
- reject y accept, herramientas de línea de comandos de CUPS, características, 512
- RELAY, regla, usar con Sendmail, 497-498
- rename, comando, usar con el programa ftp, 294
- rendimiento, herramientas y procesos de análisis de comando ps, 543
 - Frysk, 544
 - Gestor de energía de GNOME, 545
 - GKrellM, 545-546
 - KSysguard, 546
 - Monitoreo del sistema de GNOME, 543
 - System Tap, 544
 - vmstat, top, free, Xload, iostat y sar, 544
- rep, insertar el comando, 307-309
- repeat, estructura, usar en la shell TCSH, 86
- repquota, comando, usar, 572-573
- reproductores de medios, acceso a, 262-263
- RequireValidShell, directiva, usar con ProFTPD, 439
- reset, comando, usar con el programa ftp, 295
- Resource Group Manager, usar con GFS, 782
- Restart, opción, usar, 19-20
- restore, comando, usar, 701-703
- restore, comando, usar con SELinux, 345
- rgmanager, usar con GFS, 782
- Ritchie, Dennis, 7
- rlogin, comando, usar, 309
- rm, comando
 - eliminar archivos y directorios con, 130
 - usar, 45, 123
 - usar con vínculos hard, 132
- rmdir, comando
 - usar, 122-123
 - usar con el programa ftp, 294
- rmmod, comando, descargar módulos, 667
- root, control del acceso a nivel usuario, 525-526
- root, inicio de sesión en la cuenta de usuario, 524
- root, usuario
 - iniciar sesión como, 524
 - usar el comando passwd con, 524

rootnoverify, opción, usar con GRUB, 548
 rotatelog, utilidad, usar con archivos de registro
 Apache, 458
 route, comando, usar, 732-733
 RPM (Red Hat Package Manager)
 características, 221-222
 empaquetamiento de software con, 235-236
 RPM, base de datos, reconstruir, 227
 rpm, comando
 enviar, 222
 instalar y actualizar paquetes con, 226
 usar con la instalación del kernel, 673
 RPM, depósitos de paquetes, URL, 11
 .rpm, extensión, significado, 10, 220
 RPM, instalación, verificación, 226-227
 RPM, paquetes
 con el término noarch, 222
 consulta de información con, 224-225
 desplegar información, 225
 eliminar, 226
 instalar y desinstalar, 222
 nombrar convenciones, 222
 revisar claves públicas, 324-325
 rsh, inserción del comando, 307, 309-310
 RSSOwl, herramienta, usar con GNOME, 286
 rsync, comando
 acceso a sitios FTP con, 427-428
 usar, 694
 rsync, imagen de espejo, implementación, 429
 rsync, servidores, configurar, 428-429
 rules.d, directorio, contenido, 643-645
 runlevel, comando, usar, 534
 ruptime, comando, usar, 308
 rwho, comando, usar, 308

S

-s, opción, usar con el comando In, 131
 sa (asociaciones de seguridad), usar, 351-352
 SAD (base de datos de seguridad), agregar
 asociaciones de seguridad a, 351
 salida de comandos, canalización, 44
 salida estándar
 explicación, 47
 redirigir, 48-50
 saludoarg, secuencia de comandos, 72
 Samba
 explorar sistemas de Windows con, 177
 habilitar CUPS, 507
 usar para acceder a sistemas Windows,
 30-31
 sitio Web, 12

SAP, base de datos, características y sitio Web, 246-247
 sar, efecto del comando, 544
 SASL (capa de autenticación y seguridad simples),
 usar con Sendmail, 496
 savhistory, variable, usar en la shell TCSH, 111
 scandvb, herramienta, características, 263-264
 scp, cliente, características, 366-367
 scp, comando, usar, 309
 Scribus, herramienta de edición de publicaciones de
 escritorio, sitio Web, 238
 Seahorse Encryption Key Manager, acceso a, 170-171
 seaudit, herramienta, revisar mensajes SELinux, 334
 secuencia de comandos, archivos, establecer
 permisos de ejecución, 71
 secuencia de comandos, argumentos, usar, 71-73.
 Véase también argumentos
 Secure Shell (SSH), implementaciones, 359. *Véase también* OpenSSH
 Security-Enhanced Linux (SELinux). *Véase también* SELinux (Security-Enhanced Linux)
 segundo plano
 colocar comandos en, 53-54
 ejecutar trabajos en, 53-54
 seguridad
 de servidor de correo Sendmail, 496-498
 en Squid, 470-473
 seguridad, contexto de
 determinar para identidad de seguridad, 330
 en SELinux, 331
 relacionar a Flask, 327-328
 revisar en SELinux, 333
 usar, 345
 seguridad de varias categorías (MCS), extensión de SELinux con, 331-332, 336
 seguridad de varios niveles (MLS), extensión de SELinux con, 331-332, 336
 Seinfo, comando, usar con SELinux, 332-333
 SELECT, comando, usar, 517
 selector, usar con GNOME, 20
 Selector de usuarios, herramienta, usar, 20
 SELinux (Security-Enhanced Linux)
 acceso a la administración del sistema, 328-329
 administrar usuarios, 346
 agregar usuarios, 345-346
 archivo default_context, 346
 archivo default_types, 346
 archivo initrc_context, 346
 cambiar funciones, 329
 comando checkmodule, 341
 configurar, 337

- contextos de seguridad, 331
- contextos y tipos de seguridad en tiempo de ejecución, 346-347
- desactivar, 332
- descripción, 327
- deshabilitar, 332
- directorio contexts/files, 347
- directorio file_context.homedirs, 347
- dominios, 330
- etiquetado, 331
- funciones, 330
- identidades, 329-330
- implementación de políticas, 337
- MLS (seguridad de varios niveles), 331-332
- modelos de seguridad, 328
- permitir palabras clave, 339-340
- políticas, 331
- revisar el contexto de seguridad, 333
- revisar el estado, 332-333
- revisar estadísticas, 332-333
- tipos, 330
- transiciones, 331
- usar contextos de seguridad, 345
- SELinux, archivos de configuración de directivas
 - archivos .fc y .te, 340-341
 - archivos de contexto de seguridad, 343
 - archivos de interfaz, 342
 - archivos de módulo, 343
 - cambiar, 340-341
 - configurar aplicación, 344
 - funciones, 343
 - herramientas de módulo de directiva, 343-344
 - tipos de archivos, 343
- SELinux, configuración de fuente, usar, 341-342
- SELinux, directiva de referencia, usar, 335-336
- SELinux, directivas, creación, 344-345
- SELinux, herramientas de administración
 - audit2allow, 334-335
 - chcon, 334
 - herramienta Policy Analysis apol, 334
 - semanage, 334
- SELinux, mensajes, revisión, 334
- SELinux, métodos de directiva
 - imposición de tipo, 336
 - RBAC (control de acceso basado en funciones), 336
- SELinux, módulos, compilación, 341
- SELinux, permisos retenidos por usuarios, 336
- SELinux, reglas de directivas
 - composición, 337
 - contextos de archivo, 339
 - declaraciones de tipo y función, 338
 - funciones de usuario, 339
 - macros de transición y regla de vector, 340
 - reglas de permisos de función, 340
 - reglas de restricción, 340
 - vectores de acceso, 339-340
- semanage, comando, aplicar a usuarios, 346
- semanage, herramienta en SELinux, características, 334
- semanage_module, comando, usar, 344
- semodule_package, comando, usar, 341
- Sendmail, servidor de correo
 - característica redirect, 489-490
 - características, 402, 484-485
 - como servidor de correo o cliente de correo, 493
 - configurar, 487-491
 - configurar estaciones de trabajo con conexiones ISP directas, 495
 - configurar para configuración de red simple, 494
 - configurar para servidor de correo centralizado, 494-495
 - deshabilitar la opción EXPN, 498
 - deshabilitar la opción VRFY, 498
 - enmascaramiento, 491-493
 - seguridad, 496-498
 - soporte a alias y LDAP, 485, 487
 - URL, 12
 - usar el archivo access.db con, 497
 - usar el comando define con, 490
 - usar el comando divert con, 491
 - usar el comando dnl con, 491
 - usar la regla RELAY con, 497-498
 - usar la tabla mailer con, 495
 - y dominios virtuales, 496
- sendmail.cf, archivo
 - definiciones inf, 488
 - ubicación, 487
- ServerAdmin, directiva, usar con el servidor Web Apache, 451
- ServerLimit, directiva, usar con el servidor Web Apache, 451
- ServerName, directiva, usar con el servidor Web Apache, 451-452
- ServerRoot, directiva, usar con el servidor Web Apache, 449
- ServerTokens, directiva, usar con el servidor Web Apache, 449
- service, comando, usar, 403, 405
- services-admin, herramienta, características, 536

- servicio multiplexado de información y computación (MULTICS), significado de, 6-7
- servicio, secuencias de comando de
 - ejemplo, 414-415
 - etiquetas, 413-414
 - función, 412-413
 - instalación, 415
- servicios
 - acceso con Kerberos, 370
 - activar o desactivar, 410
 - configurar inicio automático, 402
 - controlar el acceso, 374
 - definición, 404
 - desplegar información de inicio, 409
 - eliminar y agregar con chkconfig, 409
 - enumerar con chkconfig, 407-408
 - establecer para paquetes, 385
 - iniciar automáticamente, 406-407
 - iniciar directamente, 405
 - iniciar y detener con chkconfig, 408-409
 - iniciar y detener con secuencias de comandos de servicio, 406
 - iniciar y detener, 403
 - números de puerto y etiquetas, 382
 - restaurar a chkconfig predeterminado, 408
- servidor de autenticación (AS), función en Kerberos, 369-370
- servidor de claves GnuPG, acceso al, 319
- servidor de nombre de dominio, explicación, 719
- servidor de nombres de dominio (DNS). *Véase también* DNS (servidor de nombres de dominio)
- servidor de otorgamiento de boletos (TGS), función en Kerberos, 369-371
- servidor ftp muy seguro (vsftpd). *Véase también* vsftpd (servidor ftp muy seguro)
- servidor independiente, función, 406
- servidores de correo remotos POP, acceso, 274-275
- servidores, iniciar con init.d, 403
- sesiones, configurar en escritorio, 27
- sestatus, comando, usar con SELinux, 332
- set, comando
 - argumentos, 93
 - definir el historial de la shell C con, 58
 - usar con las variables de shell, 68
 - usar con la shell TCSH, 108
- setenv, comando, definir la variable de entorno, 76
- setfiles, herramientas, usar en SELinux, 345
- setkey, herramienta
 - configurar conexiones con, 351-353
 - usar con IPsec, 351
- seusers, archivo, entrada root, 347
- sftp y sftp, características de clientes servidor, 367
- shell. *Véase también* BASH, la shell; C, la shell; TCSH, la shell; Z, la shell
 - acceso, 35
 - archivos de inicialización y configuración, 90
 - definición, 65
 - definir variables de entorno, 94
 - diferencia entre shell de usuario y subshell, 73-74
 - tipos, 35, 89
 - usar indicadores de comando con, 36
- shell, comandos, insertar en archivos de secuencia de comandos, 70-71
- shell, configuración del indicador de comandos, 99-100
- shell, control de operaciones, 93-94
- shell, secuencias de comandos de
 - capacidades, 65
 - definición, 66, 70
 - definir variables, 73
 - ejecutar, 71
 - usar permisos de ejecución con, 567
- shell, símbolos
 - ejecutar, 44
 - relacionar, 46-47
- shell, variables
 - asignar valores a, 67-68
 - definición, 66
 - definir y evaluar, 66-67
 - disponibilidad, 66
 - para sistemas de configuración, 109-111
 - y argumentos de comando, 68
- shell, variables de parámetros
 - características, 95-97
 - enumerar, 96
 - y archivos de inicialización, 96
- shell, versión, código de indicador de comandos, 100
- showrgb, comando, usar con X Window System, 159
- signo de interrogación (?)
 - como símbolo de shell, 44
 - función, 68
 - relacionar caracteres individuales con, 45-46
- signo de número (#), significado del indicador de comandos, 21
- signo de pesos (\$)
 - antes del comando csh, 57
 - aparición antes de shell, 36
 - como indicador de comandos, 21, 99
 - función, 21, 68
 - representar palabras, 60

- usar con la acción automática de completar de BASH, 39
- símbolos de permisos, usar, 566
- sistema de archivos, herramientas
 - fdisk, 606-608
 - mkfs, 609-610
 - mkswap, 610
 - parted, 608-609
- sistema de información de red (NIS). *Véase también* NIS (sistema de información de red)
- sistema de versión concurrente (CVS) función de, 235
- sistema, directorios
 - contenido, 119
 - en FHS (jerarquía de sistema de archivos), 585-587
 - funciones, 120, 536-537
- Sistema, menú en GNOME, capacidades, 23
- sistema, niveles de ejecución. *Véase también* niveles de ejecución
- sistema, ubicación de registros, 537
- sistemas de administración de base de datos relacionales (RDBMS)
 - base de datos SQL como, 245-247
 - estructura, 516
- sistemas de archivos. *Véase también* sistemas de archivos de red distribuidos; FHS (jerarquía de sistema de archivos)
 - acceso con el Administrador de volúmenes de GNOME, 177
 - arranque y revisión de disco, 598
 - cifrado, 326
 - crear para dispositivos RAID, 633-634
 - exportar de forma manual, 765-766
 - montaje, 583, 593
 - montar de forma automática, 596-601
 - montar de forma manual, 601-606
 - obtener información acerca, 593-594
 - organización, 583
 - reparar, 594
 - restaurar, 703
 - revisar la consistencia, 594
 - sistema de archivos proc, 641
 - sistema de archivos sysfs, 639-641
- sistemas de archivos de red (NFS). *Véase también* NFS (sistemas de archivos de red)
- sistemas de archivos de red distribuidos. *Véase también* sistemas de archivos
 - Coda, 778-779
 - GFS (Global File System), 779-784
 - PVFS (Parallel Virtual File System), 777-778
- sistemas operativos, función, 6
- sitios de información, sitios Web, 14
- sitios de noticias, sitios Web, 14
- sitios remotos, determinar direcciones IP, 303
- slogin, comando, usar, 309
- slrn, lector de noticias, características, 278
- smb, comando, usar, 31
- smb.conf, archivo, contenido, 466
- SNAT, destinos, usar, 385, 396
- .so, extensión, significado, 233
- sobrescritura (>!), símbolo de shell, ejecución, 44
- software. *Véase también* Linux, software
 - compilar, 231-232
 - descomprimir de forma separada, 229-230
 - descomprimir y extraer, 228-229
 - empaquetar con RPM, 235-236
 - extraer, 230-231
 - instalar de archiveros comprimidos, 228-233
 - opciones de consulta, 224-225
- software de base de datos, disponibilidad, 11-12
- software de fuente abierta, Linux como, 9-10
- software, depósitos
 - actualizar software, 10-11
 - contenido, 30
- software, paquetes
 - instalar y actualizar con rpm, 226
 - revisar firmas digitales, 323-325
 - tipos, 219-220
- sonido, instalación de dispositivos, 662-663
- sonido, soporte a aplicaciones, 260-261
- soporte multimedia, instalar, 27
- sort, comando, usar, 51
- SourceForge, sitio Web, 9, 11
- SpamAssassin, características, 500-501
- SPD (bases de datos de directivas), agregar directivas, 351
- spdadd, instrucción, usar con directivas de seguridad, 352
- spool, directorios, relacionar para impresión, 505
- SQL, bases de datos, como RDBMS, 245-247
- Squid, servidor proxy
 - cachés soportadas, 473-474
 - características, 467-468
 - configurar exploradores de cliente, 468-469
 - configurar memoria caché, 474
 - crear ACL (lista de control de acceso), 470
 - registros mantenidos por, 474
 - seguridad, 470-473
 - sitio Web, 12
 - usar ICP (protocolo de caché de Internet) con, 473-474
 - y aceleración de servidor Web, 474-475

- y caché de proxy inversa, 474-475
- squid.conf, archivo
 - entradas predeterminadas, 471
 - opciones http_access, 472
 - ubicación y contenido, 469-470
- SquirrelMail, características del cliente de correo, 270-271
- .src.rpm, significado de la expresión, 220
- SSH (Secure Shell), implementaciones, 359. *Véase también* OpenSSH
- SSH, archivos de configuración
 - descripciones, 364
 - ubicación, 368
- ssh, características de cliente, 365-366
- SSH, clientes
 - scp, 366-367
 - sftp y sftp-server, 367
 - ssh, 365-366
- ssh, comando, usar, 309-310
- SSH con ssh-keygen, creación de claves, 363-364
- SSI (Includes del lado del servidor)
 - características, 462-463
 - directiva AddHandler con, 462
 - directiva AddType utilizada con, 462
 - directiva Options utilizada con, 462
 - directiva XBitHack utilizada con, 462
- SSL (capa de conectores seguros)
 - usar con Sendmail, 496
 - usar con servidor Web, 464-466
- StarOffice, sitio Web para software, 12, 238, 240
- StartServer, directiva, usar con el servidor Web Apache, 451
- startx, comando, usar, 24, 156, 167
- state, extensión, usar con paquetes, 383
- steganography, definición, 323
- storage.fdi, contenido del directorio, 656-657
- Stronghold Enterprise Server, descripción, 444
- su, comando
 - descripción, 525
 - editar, 526
 - iniciar sesión al root desde, 524
 - usar en SELinux, 329
- subredes, explicación, 716
- subshell
 - generación, 73
 - referencia de variables, 74-75
- Subversion, método, función, 235
- sucesos
 - editar, 43
 - editar en el historial de la shell C, 61
 - realizar una sustitución global, 61
 - referencia con la shell Z, 63
 - referencia con la utilería de historial, 40-42
 - referencia de palabras, 60
 - sustituir en el historial de la shell C, 59-60
 - volver a ejecutar en la shell C, 59
- sudo, comando, descripción, 525
- sudo, comando, usar, 525-526
- Sun Java, sitio Web, URL, 13
- Sun Java System, descripción, 444
- Sun, sitio Web, 289
- superusuarios, capacidades, 523-524
- sustitution, comando, usar en el historial de la shell C, 61
- swat, archivo de configuración, usar con xinetd, 409
- swat, archivo, despliegue en el directorio xinetd.d, 417
- switch, estructura de control
 - efecto, 81
 - usar en TCSH, 84-86
- Sybase, base de datos, características y sitio Web, 246-247
- Sybase, software de base de datos, URL, 12
- SYMLINK, creación de reglas, 649-650
- /sys, sistema de archivos, consultar, 650-652
- sysfs, sistema de archivos, función, 589-590, 639-641
- syslog.conf, archivo
 - efecto de cambios, 537
 - ejemplo, 541
 - entradas, 539-540
- syslogd, daemon
 - acciones y usuarios, 540-541
 - función, 537
 - instalaciones, propiedades y operadores, 539-540
- System Tap, herramienta de diagnóstico, características, 544
- System V, secuencias de comandos init, etiquetas, 413
- SysV Init, iniciar servidores, 403
- sysv-rc-conf, administración de servicios, 410-412

T

Tablas

- acciones de acceso de Sendmail, 496
- aplicaciones de OpenOffice.org, 239
- aplicaciones de seguridad de red, 374
- aplicaciones KOffice, 242
- aplicaciones multimedia y de sonido, 259
- archiveros de archivos tar, 135
- archiveros, depósitos y vínculos de software de terceros de Linux, 11
- archivos de configuración CUPS, 508

- archivos de configuración de directivas SELinux, 342
- archivos de configuración de usuario, 552
- archivos de configuración INN, 513
- archivos de configuración shell, 91
- archivos de configuración SSH, 364
- archivos de configuración X Window System, 161
- archivos de información de dispositivo /proc, 641
- archivos soportados por NSS, 728
- archivos vsftpd, 433
- archivos y direcciones de configuración TCP/IP, 724
- archivos y directorios de configuración de sistema, 538
- archivos y directorios de configuración GDM, 166
- archivos y directorios de configuración XDM, 164
- archivos y directorios de inicio de sistema, 402
- archivos y directorios de Sendmail, 486
- archivos y directorios del servidor Web Apache, 446
- atributos xinetd, 419-420
- cadenas integradas Netfilter, 377
- caracteres de expansión de archivo, 44
- características de Sendmail, 489-490
- características especiales de la shell BASH, 92
- claves de reglas udev, 645
- clientes de correo GNOME, 269
- clientes de correo, 266
- clientes de impresión de CUPS, 509
- clientes FTP, 291
- clientes para charla en red y mensajería, 303
- códigos de discrepancia RPM, 227
- códigos de indicador de comandos, 100
- códigos de sustitución udev, 646
- comando find, 127
- comando mount, 602
- comandos de acceso remoto, 307
- comandos de Amanda, 696
- comandos de directorio, 122
- comandos de historial y eventos de historial, 41
- comandos de IPtables, 377
- comandos de módulo kernel, 665
- comandos del editor Vi, 251-252
- comandos del historial de la shell C, 58
- comandos del programa ftp, 295
- comandos fdisk, 608
- comandos, herramientas, archivos y directorios cron, 530
- comandos LVM, 618
- comandos MySQL, 519
- comandos para completar texto en la línea de comandos, 40
- comandos SQL, 516
- comandos X Window System, 157
- comandos y opciones de GnuPG, 316-317
- comodines de envoltura TCP, 421
- controladores de dispositivo de video y TV, 663
- declaraciones, parámetros y opciones DHCP, 752-753
- destinos de IPtables, 376
- direcciones IP de red local IPv4, 718
- directivas de almacenamiento HAL, 655
- directorios /usr en FHS, 587
- directorios de configuración de GNOME, 194
- directorios de instalación de KDE, 216
- directorios de sistema de archivo, 585
- directorios del sistema, 120, 536, 586
- directorios Xorg, 147
- dispositivos de sonido, 662
- distribuciones y sitios del kernel Linux, 5
- editores, 249
- editores de escritorio, 249
- enrutar entradas de tabla, 732
- enumerar, desplegar e imprimir archivos, 121
- estructuras de control condicional TCSH, 84
- estructuras de control de bucle TCSH, 86
- estructuras de control de la shell BASH, 79
- etiquetas de secuencia de comandos System V init, 413
- extensión de archivos de software de paquete, 220
- funciones de la secuencia de comandos init, 413
- GNOME Office, 243
- grupos de desarrollo de protocolo TCP/IP, 708
- grupos de noticias Usenet, 15
- herramientas administrativas CUPS, 511
- herramientas de administración de grupos y usuarios, 557
- herramientas de administración de SELinux, 333
- herramientas de administración de sistema, 525
- herramientas de configuración X Window System, 148
- herramientas de creación de particiones y sistemas de archivos, 607
- herramientas de IPsec, 351
- herramientas de red, 302

- herramientas de rendimiento, 543
- herramientas GFS, 782
- herramientas gráficas para Linux, 258
- herramientas SSH, 362
- historial de la shell Z, 64
- lectores de noticias, 277
- Linux Documentation Project (LDP), 14
- máscaras de red CIDR IPv4, 715
- menú de escritorio de GNOME, 176
- menú del administrador de archivos
 - Nautilus, 181
- menú emergente de archivos de Nautilus, 183
- métodos abreviados de teclado del
 - administrador de archivos KDE, 211
- modos de mdadm, 630
- MTA (agentes de transferencia de correo), 478
- niveles de ejecución para Red Hat y Debian, 405
- niveles de ejecución, 532
- niveles RAID para software de Linux, 627
- nombres de comandos para invocar shell, 90
- opciones /etc/exports, 763
- opciones ACL de Squid, 471
- opciones chkconfig, 408
- opciones de apagado, 535
- opciones de compilación del comando make del
 - kernel, 681
- opciones de configuración de aplicación X
 - Window System, 156
- opciones de configuración vsftpd.conf, 431
- opciones de consulta de paquete RPM, 225
- opciones de consulta para software instalado, 225
- opciones de cuota, 573
- opciones de gzip, 139
- opciones de ifconfig, 730
- opciones de IPtables, 378-379
- opciones de la utilería dump, 699
- opciones de mdadm-create, 633
- opciones de mdadm.conf, 631
- opciones de mkfs, 609
- opciones de montaje NFS, 769
- opciones de montaje para sistemas de archivos, 599
- opciones de RPM, 223-224
- opciones del comando chage, 557
- opciones del sistema de resolución host.conf, 727
- opciones edquota, 572
- opciones useradd y usermod, 558
- operación de administración de trabajos, 53
- operaciones de edición de línea de comando, 38
- operaciones y opciones de restauración, 702
- operadores de expresión de prueba TCSH, 83
- operadores de prueba de la shell BASH, 78
- paquete de protocolo TCP/IP, 709
- paquetes de oficina, 238
- paquetes ICMP, 381
- paquetes Java y aplicaciones Web de Java, 288
- parámetros del Administrador de la red, 736
- permisos para archivos y directorios, 564
- prefijos de formato IPv6, 722
- prefijos de nombre de dispositivo, 591
- programación de Linux, 13
- protocolos de Squid, 468
- protocolos Web, 282
- proyectos y aplicaciones de video y DVD, 262
- recursos de base de datos, 515
- recursos de dispositivo, 640
- recursos de GNOME, 170
- recursos de impresión, 504
- recursos de información, 29
- recursos de IPsec, 350
- recursos de Mozilla, 284
- recursos de respaldo, 694
- recursos de SELinux, 328
- recursos SSH y Kerberos, 360
- redirección de operaciones de shell, 48-50
- restaurar comandos shell del modo interactivo, 703
- secuencias de comandos de servicio
 - /etc/init/d, 404
- servicios de configuración NSS, 728
- servidores FTP, 424
- shell, 36
- símbolos de shell, 44
- sistemas de administración de base de datos, 246
- sistemas distribuidos de archivos, 778
- sitios de información y noticias de Linux, 14
- sitios de PGP (Pretty Good Privacy, muy buena privacidad), 314
- sitios Web de KDE, 198
- sitios Web relacionados con Apache, 445
- software de servidor de red y seguridad, 12
- subdirectorios /var, 588
- subdirectorios y archivos /proc, 589
- subredes y máscaras, 717
- tipos de archivos Web, 283
- tipos de sistema de archivos, 597

- variables de entorno de sistema utilizadas por shell, 96
- variables de shell, 95
- variables wvdial, 738
- vínculos simbólicos de dispositivo, 647
- visores PostScript, PDF y DVI viewers, 244
- Talk, utilería, características, 305
- TANGO, relacionar con GNOME, 170
- .tar, extensión, significado, 220
- tar, archiveros, extraer con el administrador de archivos de KDE, 210
- tar, comando
 - usar, 134
 - usar la extensión x, 230
 - usar la opción c, 135
 - usar la opción f, 134
 - usar la opción u, 136
 - usar la opción z, 137, 140
- tar, comprimir los miembros del archivo, 140
- tar, utilería
 - actualizar archiveros con, 136-137
 - archivar archivos y dispositivos con, 134-138
 - archivar en cinta con, 138
 - archivar en discos flexibles con, 137
 - comprimir archivos con, 137-138
 - crear archivos con, 134-136
 - descomprimir y extraer software con, 228-229
 - desplegar contenidos de archiveros con, 134
 - extraer archiveros con, 136
 - .tar.bz2, significado de la extensión, 220
 - .tar.gz, significado de la extensión, 137, 139, 220
- tareas
 - organización de tareas programadas, 529
 - programar con cron, 527-528
- tareas programadas, organizar, 529
- tarjeta madre, disponibilidad del soporte RAID, 626-627
- TCP, envolturas, usar con servicios xinetd, 421-422
- TCP/IP (protocolo de control de transmisión/protocolo de Internet), función, 707-709
- TCP/IP, archivos de configuración
 - /etc/hosts, 723-725
 - /etc/protocols, 725
 - /etc/resolv.conf, 725
 - /etc/services, 725
- TCP/IP, direcciones de red
 - direccionamiento IP basado en clase, 712-713
 - direcciones de difusión, 719
 - direcciones de puerta de enlace, 719
 - direcciones de red IPv4, 712
 - direcciones de servidor de nombres, 719-720
 - y CIDR (enrutamiento entre dominios y sin clase), 714-717
 - y máscaras de red, 713-714
- tcpd, invocar el daemon, 422
- tcpdump, herramienta, características, 741
- TCSH, shell. *Véase también* shell
 - archivos de configuración, 91
 - archivos de inicialización, 111-114
 - bucles, 86-88
 - cadena de secuencias de comandos, 82
 - capacidades de secuencia de comandos, 66
 - características, 62, 89
 - caso de variables, 109
 - completar en la línea de comandos, 62
 - conjunto argv, 72-73
 - editar el historial, 62-63
 - estructura if-then, 84
 - estructura switch, 85-86
 - estructuras de control condicional, 82
 - estructuras de control, 81-88
 - estructuras de control if, 83
 - operadores de expresión de prueba, 83
 - relacionar a la shell C, 57
 - sitio Web, 36
 - usar comando echo, 108
 - usar el comando alias con, 107-108
 - usar el comando ignoreeof con, 108
 - variable cdpath, 110
 - variable history, 111
 - variable mail, 111
 - variable savhistory, 111
 - variables de definición, 108
 - variables de entorno, 76
 - variables de indicador de comandos, 109-110
- .tcshrc, archivo de inicialización en TCSH
 - descripción, 111
 - función, 112-113
- .te, archivos, usar en SELinux, 340-341
- TE (imposición de tipo), usar en SELinux, 328
- teclado con KDE, configuración, 23
- teclas de unión, configuración de, 37
- telinit, comando
 - apagado de sistemas con, 534
 - cambiar niveles de ejecución con, 149, 533-534
 - descripción, 525
- Telnet, sesiones, prohibir el comando su en sesiones, 525
- Telnet, utilería, invocar, 306
- temas
 - administrar en KDE, 25
 - personalizar en GNOME, 24-25

- seleccionar en GNOME, 172
- termcap, archivo, contenido, 661
- terminal, acceso a la ventana, 27-28
- terminal, dispositivos
 - inicializar con el comando tset, 661
 - instalar y administrar, 660-661
- test, comando, usar, 77-78
- TeX, herramienta de composición tipográfica,
 - disponibilidad, 244
- texto, agregar en la shell BASH, 37
- TGS (servidor de otorgamiento de boletos), función
 - en Kerberos, 369-371
- TGT (boleto que otorga un boleto), función en
 - Kerberos, 369-370
- The Open Group (TOG), acceso al sitio Web, 146
- Thompson, Ken, 7
- ThreadsPerChild, directiva, usar con el servidor Web
 - Apache, 451
- Thunderbird, cliente de correo
 - características, 268-269
 - habilitar LDAP, 580
- tilde (~), usar con la acción de completar automática
 - de BASH, 39
- Timeout, directiva, usar con el servidor Web Apache,
 - 449
- .title, argumento, usar con X Window System, 156
- TOG (The Open Group), acceso al sitio Web, 146
- .torrent, extensión, significado, 220
- torrents, inicio, 221
- Torvalds, Linus, 7
- TOS, usar destinos, 385
- Totem
 - características, 263
 - sitio Web, 262
- trabajos
 - cancelar, 55-56
 - cancelar e interrumpir, 53
 - ejecutar en segundo plano, 53-54
 - llevar al frente, 54-55
 - obtención de números, 54
 - referencia, 53-54
 - suspender y detener, 55
 - usar el comando notify con, 54
- traceroute, herramienta de red, 301-304
- traducción de direcciones de red (NAT), explicación,
 - 384
- TransferLog, directiva, usar con el servidor Web
 - Apache, 458
- Tripwire, implementación de detección de intrusos
 - con, 325
- Trolltech, sitio Web, acceso, 198

- tset, colocación del comando, 661
- TSIG, clave, uso con actualizaciones de DNS de DHCP
 - dinámica, usar, 756
- TTY (identificador de terminal), significado, 55-56
- Turbo Linux, URL, 5
- Tux, servidor Web, características, 443-444
- TV, dispositivos, instalación, 663
- TV, reproductores, disponibilidad, 263
- tvtime, reproductor de TV, características, 263
- tvtime, sitio Web, 262
- twm, administrador de ventanas, iniciar, 167-168
- type, opción, usar con el comando type, 126
- TypesConfig, directiva, usar con el servidor Web
 - Apache, 455
- .tz, significado de la extensión, 220

U

- Ubuntu Linux, URL, 5
- udev (dispositivos de usuario), herramienta, usar,
 - 590
- udev, herramienta de inserción activa
 - configurar, 642-643
 - función, 641-642
- udev, reglas
 - crear, 648-649
 - relacionar con nombres de dispositivo, 643-645
- udevinfo, comando, usar, 650-652
- umask, configurar permisos predeterminados,
 - 570-571
- UML (user-mode Linux), función, 674
- Umount, comando, usar con sistemas de archivo, 603
- Unalias, comando, usar con la shell TCSH, 108
- uncompress, usar el comando, 140
- unicast, dirección, explicación, 721-722
- unidades, remplazo con LVM, 622-623
- unión (&), operador
 - como símbolo de shell, 44
 - función de, 68
 - usar con trabajos en segundo plano, 53-54
- Universidad de Washington, servidores POP e IMAP,
 - sitio Web, 499
- Unix
 - en comparación con Linux y, 4
 - historia, 7
- unrar, herramienta, descargar, 133
- unset, comando
 - usar con la shell TCSH, 108
 - usar con variables de shell, 68
- unzip, comando, usar, 141
- update-rc.d, herramienta, usar con vínculos de nivel
 - de ejecución, 411-412

upgrade, comando, usar con Debian, 228
 URL (localizador universal de recursos), componentes, 282
 URL, nombres de ruta, directivas utilizadas con, 453-454. *Véase también* nombres de ruta
 URL. *Véase también* sitios Web
 USB desde el escritorio KDE, acceso a unidades, 208
 UseCanonicalName, directiva, usar con el servidor Web Apache, 460-461
 Usenet, artículos, leer con lectores de noticias, 277
 Usenet, grupos de noticias, disponibilidad, 15
 Usenet, noticias
 acceso, 276
 alimentación de noticias, 278
 descripción, 275
 usar agentes de transporte de noticias con, 278-279
 useradd, comando, usar, 558-559
 userod, comando, usar, 559
 users-admin, herramienta, características, 551-552
 uso de espacio en disco, determinación para sistema de archivos, 593-594
 -usr, directorio, contenido en FHS, 587
 usuario, archivos de configuración, rutas, 552
 usuario, cadenas, definición, 393
 usuario, entornos, administración, 554-557
 usuario, inicio de sesión, administrar con KDM, 166-167
 usuario, interfaz, relacionar con el sistema operativo, 6
 usuario, sesiones de inicio, administrar con GDM, 164-166
 usuario, shell de
 deshabilitar caracteres especiales, 109
 generación, 73,94
 usuarios
 agregar en SELinux, 345-346
 agregar o eliminar, 557-559
 control de contraseñas, 556
 creación de grupos privados, 560
 identificar cuando se desmontan sistemas de archivos, 603
 lista con el comando who, 303
 seleccionar, 524
 usuarios que han iniciado sesión, determinar, 553
 utilería de historial de la shell C
 comandos en, 57-58
 definir con el comando set, 58
 edición de sucesos, 61
 referencias de patrón utilizadas con, 58-59
 sustituciones de sucesos, 59-60

usar comandos de sustitución en, 61

V

valores de variable, usar en shell y subshell, 74
 /var, directorio, contenido en FHS, 588
 variables
 asignar resultados de Linux caso, 109
 comandos a, 70
 definir en la shell TCSH, 108
 definir en secuencias de comandos shell, 73
 referenciar en subshells, 74-75
 representar en la shell BASH, 39
 variables de entorno
 capacidades, 75
 definir en shell, 94
 en las shell Bourne, BASH y Korn, 75-76
 en las shell TCSH y C, 76
 variables de parámetro, exportar, 101
 /var/log, directorio, contenido, 537
 verificación de integridad, realizar con firmas digitales, 314
 Vgcreate, comando, usar con grupos LVM, 619
 Vgextend, comando, usar con grupos LVM, 620
 Vi, editor, usar con eventos de historial, 42
 video y reproductores de DVD, aplicaciones, 262-263
 video, dispositivos, instalación, 663
 VideoLAN, sitio Web, 262
 vim, comando, ejecución, 253
 Vim, editor, características, 250-254
 vínculos
 duros, 131-132
 simbólicos, 131
 vínculos duros, función, 130-132
 vínculos simbólicos
 archivos de dispositivo, 645-647
 función, 130-131
 para lectores CD/DVD, 648
 virt-install, secuencia de comandos, usar la, 690-691
 virt-manager en Red Hat, características, 686-687
 VirtualDocumentRoot, directiva, usar con el servidor Web Apache, 460
 VirtualHost, directiva, usar con ProFTPD, 440-442
 virtualización de hardware, usar con KVM, 687-688
 virtualización, métodos de
 disponibilidad, 685-686
 KVM (máquina de virtualización basada en kernel), 687-688
 virt-manager en Red Hat, 686-687
 virtualización Xen, 688-692

- VirtualScriptAlias, directiva, usar con el servidor Web Apache, 460
- virtusertable.db para Sendmail, generación del archivo, 496
- visor de imágenes en miniatura, disponibilidad del, 257
- visores de documentos, disponibilidad, 244
- visudo, comando, usar, 526
- vmstat, comando, efecto, 544
- VMware, sitio Web, 239
- VoIP Ekiga, aplicaciones, 304-305
- VRFY, opción, deshabilitar en Sendmail, 498
- vsftpd (servidor ftp muy seguro)
 - acceso de usuario, 433
 - autenticación de usuario, 434
 - comando access, 434
 - configurar, 430
 - ejecutar, 429-430
 - habilitar el acceso independiente y de inicio de sesión, 430
 - habilitar permisos de usuario local, 430-432
 - host virtual, 434-435
 - implementar usuarios virtuales, 435
 - inicio de sesión, 433
 - límites de tiempo de conexión, 432
 - negar el acceso, 433
 - permitir mensajes, 432
 - permitir usuario anónimo, 432
 - restricciones de usuario, 434

W

- w (escritura), ejecución del permiso, 562, 566
- .wav, tipo de archivo Web, descripción, 283
- Web, creación de páginas, 286-287
- Web, exploradores
 - características, 282-283
 - ELinks, 286
 - en GNOME, 286
 - Firefox, 284-285
 - Lynx, 286
 - marco estructural de Mozilla, 283-284
 - ventana del administrador de archivos K Desktop, 285-286
- Web, FTP basado en explorador, usar Firefox, 291-292
- Web, servidores
 - acceso, 286
 - Apache, 444-448
 - Apache-SSL, 444
 - lighttpd, 444
 - Netscape Enterprise Server, 444

- PHP (PHP: Hypertext Preprocessor), 463
- servidor de aplicación Zope, 444
- servidor Web NCSA, 444
- SSI (Includes del lado del servidor), 462-463
- Stronghold Enterprise Server, 444
- Sun Java System, 444
- Tux, 443-444
- y SSL (capa de conectores seguros), 464-466
- Web, sitios
 - Apache Jakarta Project, 445
 - Apache-SSL, 444
 - aplicación de administrador de finanzas GnuCash, 244
 - aplicaciones de GNOME, 11
 - aplicaciones de Java, 288
 - aplicaciones de seguridad de red, 374
 - archivero GNU, 11
 - biblioteca de desarrolladores KDE, 13
 - clientes de correo Emacs, 271
 - clientes de correo para GNOME, 269
 - codecs y plugins multimedia de gstreamer, 11
 - Compiladores y herramientas
 - de Linux (gcc), 13
 - compresión Ogg Vorbis para archivos de música, 261
 - controladores comerciales cedega, 31
 - creación, 286-287
 - Crossover Office, 238
 - depósito de Java, 11
 - depósito de paquetes RPM, 11
 - depósito de software KDE, 11
 - distribuciones y kernel de Linux, 5
 - firewall IP Tables, 12
 - Fluendo, 258
 - GFS (sistema de archivos global), 780
 - GNOME (entorno de modelo de objeto de red GNU), 170
 - GNOME Keyring Manager, 317
 - GNOME Office, 238
 - GNU Java Compiler, 288
 - GnuPG (GNU Privacy Guard), 322
 - gStreamer, 259
 - herramienta de edición de publicaciones de escritorio Scribus, 238
 - herramientas de red Kerberizadas, 371
 - Internet Software Consortium, 12
 - Jakarta Project, 288
 - JPackage Project, 287
 - juegos de Linux, 11
 - la shell BASH, 36
 - lighttpd, 444

- Linux Foundation, 8
- Linux Game Tome, 11
- Majordomo, 275
- Netscape Enterprise Server, 444
- Open Secure Shell, 12
- OpenOffice, 238
- OpenSSL, 465
- paquete KOffice para KDE, 238
- paquete StarOffice, 238
- Perl, 13
- PGP (Pretty Good Privacy, muy buena privacidad), 314
- protocolo de autenticación de Kerberos, descripción del, 368-369
- protocolo de autenticación de red Kerberos, 12
- proyecto AbiSource, 243
- proyecto Blackdown, 287
- proyecto gPhoto, 257
- proyecto Mozilla, 268-269
- proyecto OpenPGP Public Keyserver, 319
- proyecto XFree86, 145
- realizar búsquedas de documentos, 520
- recursos de Kerberos, 360
- recursos de respaldo, 694
- recursos SELinux, 328
- referencias de kernel, 672
- servidor Courier-IMAP, 500
- servidor Cyrus IMAP, 500
- servidor de aplicación Zope, 444
- servidor de correo Postfix, 12
- servidor de correo Sendmail, 12
- servidor de noticias Leafnode, 515
- servidor FTP muy seguro, 12
- servidor FTP ProFTPD, 12
- servidor proxy Squid, 12, 467
- servidor Qpopper, 500
- servidor SMB Samba, 12
- servidor Web Apache, 12
- servidor Web NCSA, 444
- servidores IMAP y POP, 499-500
- servidores POP e IMAP de University of Washington, 499
- servidores POP e IMAP, 499-500
- shell, 36
- sistemas de administración de base de datos, 246
- sitio Cluster Project Page, 780
- sitio Web de desarrolladores GNOME, 13
- sitio Web de Sun Java, 13
- sitios Web KDE, 198
- software de base de datos DB2, 12
- software de base de datos IBM DB2, 12
- software de base de datos MySQL, 12
- software de base de datos Oracle, 12
- software de base de datos PostgreSQL, 12
- software de base de datos SQL de GNU, 12
- software de base de datos Sybase, 12
- software de GNOME Office, 12
- software KOffice, 12
- software OpenOffice, 12
- software StarOffice, 12
- SourceForge, 9, 11
- SSH Communications Security, 359
- StarOffice, 240
- Stronghold Enterprise Server, 444
- Sun Java System, 444
- Sun, 289
- tarjeta de video PCHDTV, 263
- temas de GNOME, 172
- Trolltech, 198
- VideoLAN, 262
- VMware, 239
- Wine, 238
- X.org Foundation, 145
- Xen Virtualization Kernel, 688
- Xorg, 146
- Web, tipos de archivos, 283
- Web y direcciones URL, clientes, 282
- Webalizer, herramienta, generar informes web-log, 457
- Welsh, Matt, 13
- wget, herramienta, acceso a sitios Web y FTP, 293
- while, estructura de control
 - efecto, 81
 - expresiones de prueba utilizadas con, 82
 - usar con la shell TCSH, 86-87
- who, comando, enumerar usuarios en línea con, 302-303
- whois, herramienta de red, características, 302
- Windows, acceso de red, configurar, 30-31
- Windows, aplicaciones, instalar con Wine, 32
- Windows, juegos en Linux, 31
- Windows, particiones, montar, 600-601
- Windows, sistemas, explorar en GNOME, 177
- Windows, software, ejecutar en Linux, 31-32
- Wine, capa de compatibilidad de Windows
 - configurar, 32
 - instalar, 31
- wine, comando, usar, 32
- Wine, sitio Web, 238
- Wireless Tools, características, 735

Wireshark, analizador de protocolo de red,
características, 739-741
wireshark, herramienta de red, características, 302
wvdial, programa, implementar el acceso de línea de
comandos PPP, 737-738

X

x (ejecución), permiso
explicación, 562
involucrar, 566
usar máscaras binarias con, 567-568
X, aplicaciones, invocar, 149
X, comandos, ejemplos, 160
X, configuración, obtención de descripción, 158
X, despliegues, administración con XDM, 163-164
X, interfaz, configurar, 146
X, protocolo, desarrollo, 146
X, servidores
usar con X Window System, 148
y puerto de seguridad, 382
X, utilerías, descargar, 146
X. Véase X Window System
XII. Véase X Window System
X Display Manager (XDM)
características, 163-164
disponibilidad, 160
X.org Foundation, sitio Web, acceso, 145
X Session manager (xsm), características, 163
X Window, documentos HOWTO, consulta, 149
X Window System
argumento -geometry, 155
argumentos -title, 156
comandos, 157
configuración, 145
configuraciones gráficas, 158
controlar el despliegue de ventanas root, 156
ejecutar archivos de configuración, 156
establecer el color de primer plano, 156
establecer el color de segundo plano, 156
establecer fuentes, 156
herramientas de configuración, 148
iniciar, 167
programas gráficos, 257
soporte de fuentes, 158
X Window System, inicio de aplicaciones, 155
X Window System, inicio de sesiones, 166-167
X Window System, instalación del servidor, 145-146
x264, características del codec, 263
.Xauthority, archivo, contenido, 161
Xbase, base de datos, características de y sitio Web,
246, 248

XBitHack, directiva, usar con SSI (Includes del lado
del servidor), 462
.Xclients, archivo, contenido, 156
xconfig, herramienta, usar con kernels, 678-679
.Xdefaults, archivo, acceso, 158
XDM (X Display Manager)
características, 163-164
disponibilidad de, 160
Xen, virtualización, implementación, 688-692
Xerm, ventana, inicio, 167-168
XFce4, escritorio, características, 22
Xfig, programa, características, 257
XFree86, sitio Web del proyecto, acceso, 145
XFS, servidor de fuentes, configuración, 158
xine
características, 263
sitio Web, 262
xinetd (daemon extendido de servicios de Internet),
función, 415-416
xinetd, atributos, usar, 418-420
xinetd, servicios
archivo de configuración swat, 409
archivos de configuración, 417
configuración, 418
configurar para utilizar por chkconfig, 409-410
deshabilitar y habilitar, 418, 420-421
habilitar y deshabilitar, 409
iniciar y detener, 416
usar envolturas TCP con, 421-422
usar, 406-407
utilizar el atributo disable con, 420-421
y seguridad de red, 417
xinetd.conf, archivo, contenido, 416
xinetd.d, directorio, contenido, 417
xinit, comando, iniciar X Window System con, 167
.xinitrc, secuencias de comandos, crear, 167
Xload, comando, efecto, 544
xm, administración de máquinas
virtuales Xen con, 692
xmkmf, usar, 232
Xmodmap, archivo, contenido, 160
Xmorph, programa, características, 257
xmtr, herramienta de red, características, 302
Xorg, configuración
de características, 152
de la sección Device, 154
de la sección Files, 151
de la sección Input Device, 152-153
de la sección Module, 152
de la sección Monitor, 153-154
de la sección Mouse, 153

- de la sección Screen, 150-151
- de la sección ServerLayout, 154-155
- de varios monitores, 155
- Xorg, instalar el servidor, 146
- xorgcfg, herramienta, usar, 149
- xorgconfig, herramienta, usar, 149
- Xorg
 - actualizar, 147
 - bibliotecas disponibles, 147
 - directorios, 147
 - páginas Man, 147, 149
 - servidor utilizado por, 147
 - ubicación de aplicaciones y servidores, 147-148
 - ubicación de archivos de configuración, 147
- Xpaint, programa, características, 257
- xrdb, comando, efecto, 158
- .Xresources, archivo, entradas, 158-159
- xrog, archivos de configuración, probar, 149
- Xsession, secuencia de comandos
 - contenido, 162
 - invocar, 162
 - lista de entornos, 163

- xset, comando, efecto, 160
- xsetroot, comando, efecto, 160
- xsm (X Session manager), características, 163
- Xterm, archivo, contenido, 158
- XviD
 - características, 264
 - sitio Web, 262

Z

- .Z, extensión, significado, 220
- Z, shell. *Véase también* shell
 - archivos de configuración, 91
 - archivos de inicialización y configuración, 90
 - características, 63, 89
 - comandos de historial, 64
 - designadores de palabras, 64
 - sitio Web, 36
- Zero Configuration Networking (zeroconf),
 - capacidades, 710-711
- zip, comando, usar la opción -r con, 140-141
- Zip, utilería, comprimir archivos, 140-141
- Zope, servidor de aplicaciones, descripción, 444