

# 1. Historia de Linux

Tanto la creación de Linux como su desarrollo y popularización son acontecimientos relativamente recientes. En consecuencia, cabe preguntarse si merece la pena dedicar tiempo y espacio a una historia que, temporalmente, apenas abarca una década. La principal de estas razones es que las características básicas de Linux se encuentran directamente vinculadas tanto con los orígenes del sistema como con el ámbito en el que ha tenido lugar su desarrollo. Linux puede considerarse el resultado de la evolución constante de ciertas ideas básicas. Así pues, el conocimiento de dichas ideas y del proceso que las ha hecho evolucionar hasta dar lugar a Linux (a sus distintas versiones) se convierte en fundamental para la comprensión plena de las particularidades que presenta el sistema, y, en definitiva, de sus virtudes y defectos. Este planteamiento se refuerza si recordamos que, actualmente, Linux sigue evolucionando y perfeccionándose (y también ganando adeptos) día a día. El seguimiento de esta evolución permanente también se beneficiará del conocimiento de los pasos dados hasta la fecha.

Por otra parte, la vida de Linux hasta el momento, aunque breve, ha sido decididamente intensa. No podía ser de otro modo para un sistema que, contra viento y marea y contra todo pronóstico, ha conseguido desarrollarse, sobrevivir y ganar aceptación hasta convertirse en el fenómeno que es hoy en día. No en vano, muy pocos podían pensar, a comienzos de la década de los noventa, que Linux llegaría hasta donde ha llegado. Estas circunstancias han provocado que, en ocasiones, se hable o escriba sobre Linux en términos casi épicos, dotando al sistema de cierto halo revolucionario, rebelde y hasta heroico. En cualquier caso, y sin necesidad de exagerar, es innegable que la historia de Linux contiene ingredientes suficientes como para interesar a cualquiera, independientemente de si se es o no conocedor o partidario de este u otros sistemas, o incluso de la informática en general.

Además, el desarrollo de Linux está asociado a determinados movimientos y tendencias que en los últimos tiempos han abierto nuevos horizontes en el mundo de la informática, planteando tanto la necesidad de nuevos enfoques como dudas respecto a la validez de antiguos conceptos. No nos referimos únicamente a aquellas tendencias meramente tecnológicas, sino también a ciertas tendencias sociales, como podría ser la popularización mundial del uso de Internet. En definitiva, la historia de Linux es un buen ejemplo para reflejar los profundos cambios experimentados en el ámbito informático.

## 1.1 Los antepasados directos: Unix y Minix

Sin duda, muchos de los lectores habrán escuchado o leído en alguna ocasión frases que definen a Linux como «un Unix para PC». Este tipo de afirmaciones no son totalmente correctas ni definen plenamente al sistema, pero sí que permiten formar una idea aproximada sobre cuál es el origen de Linux y sobre cuáles son sus características más elementales. Esto es debido a que Linux, sin ser una mere versión

de Unix para PC, sí puede considerarse un descendiente directo de este sistema operativo.

Efectivamente, la relación de «parentesco» entre ambos sistemas es tan estrecha que es ineludible referirse a Unix a la hora de abordar el tema del origen de Linux. Actualmente, Unix es un sistema operativo sobradamente conocido y empleado ampliamente (tanto en PC como en máquinas de mayor potencia, como estaciones de trabajo -*workstations* o supercomputadores), por lo que no entraremos en describirlo al detalle y nos limitaremos a destacar los aspectos más relevantes de su filosofía básica. Así, Unix es un sistema robusto, estable, multiusuario, multitarea, multiplataforma, y con gran capacidad para la gestión de redes. Nuestro interés en estas características es debido a que definen también la filosofía básica de diseño de Linux, ya que este último fue creado siguiendo estas premisas ya establecidas. En definitiva, puede decirse que Linux, como descendiente de Unix, ha heredado de este «antepasado » sus características fundamentales, aquellas que marcan su comportamiento como sistema operativo. Las primeras versiones de Unix fueron desarrolladas en la década de los setenta por AT&T y sus filiales, y fueron (y son todavía) explotadas comercialmente de forma tradicional por sus propietarios. A finales de la década de los ochenta, el pago de una licencia de estas versiones podía ser asumido más o menos sin problemas por una empresa, pero, para la mayoría de usuarios de ordenadores personales, la cifra requerida suponía un desembolso excesivo.

Ahora bien, esta situación mejoró para los usuarios con la aparición de un precedente de Linux todavía más cercano a éste en el tiempo. Se trataba de Minix, un sistema operativo desarrollado por Andrew S. Tanenbaum, conocido como el prolífico autor de libros empleados en todo el mundo para el estudio y difusión de la informática en sus diversas facetas. Minix pretendía ser una implementación básica y reducida de Unix: una especie de Unix en miniatura o «de bolsillo», como su nombre indica. Pero su autor no estaba guiado por intereses comerciales, sino que sus motivaciones eran, básicamente, académicas y docentes. Tanenbaum pretendía que sus alumnos pudieran aproximarse a Unix y practicar el manejo del mismo (o al menos de algo similar) en sus ordenadores personales sin tener que afrontar el pago de una costosa licencia. Así, Tanenbaum desarrolló él mismo en su totalidad el código de Minix (en C y en ensamblador), y lo incluyó como ejemplo de implementación y diseño de un sistema operativo en uno de sus libros: «*Operating Systems: Design and Implementation*», publicado en 1988. De esta forma, muchos estudiosos de los sistemas operativos tuvieron acceso a un código que les permitía entrar en el mundo de Unix de forma asequible.

Aunque Tanenbaum acabó vendiendo los derechos de su sistema, sembró con Minix la semilla de la que florecería Linux, ya que la primera versión de este sistema no pretendía ser otra cosa que un Minix mejorado. Esto nos lleva directamente a abordar el momento crucial de la génesis de Linux, en el que, naturalmente, hemos de referirnos a su creador. De todo ello nos ocupamos en el siguiente apartado.

## 1.2 El padre de Linux: Linus Torvalds

Linus Torvalds nació en Helsinki, Finlandia, el 29 de julio de 1970. A principios de los noventa, Linus Torvalds era un estudiante de informática en la Universidad de Helsinki que practicaba con Minix. A pesar de su innegable talento, y casi con toda seguridad, este finlandés no podía imaginar en esa época que una década después sería conocido (y respetado) en el mundo entero como una de las figuras clave en el mundo de la informática, ni que sus intervenciones en conferencias y otros actos públicos despertarían la máxima expectación. Pero así ha sido, debido a la serie de acontecimientos que el señor Torvalds estaba a punto de desencadenar a mediados del año 1991.

En esos días, Linus se encontraba trabajando en un proyecto propio para mejorar Minix (al que encontraba ciertas deficiencias), o para desarrollar, de algún modo, algún sistema similar a Minix que funcionara mejor que Minix. Linus comenzó desarrollando este proyecto en su 386 utilizando ensamblador, pero pronto recurrió a C, lo que, según el propio autor, aceleró el proceso. De esta forma, Linus Torvalds creó lo que puede considerarse la primera versión de Linux (contracción de Linus y Unix), que posteriormente el mismo autor numeraría como versión 0.01. Téngase en cuenta que a la primera versión de un programa que resulta completamente operativa y funcional se la considera, por convenio, versión 1.0 de dicho programa. Esto permite apreciar claramente que la versión 0.01 de Linux era poco menos que el juguete de un programador, y así lo entendió el propio Linus. Efectivamente, esta versión primigenia sólo contenía en realidad un kernel muy rudimentario, y para poder realizar cualquier operación (de las escasas posibles) era imprescindible que la máquina también tuviera instalado Minix.

Sin embargo, la versión 0.01 de Linux no llegaría a difundirse. Linus Torvalds continuó modificando el nuevo sistema operativo, y no lo hizo público hasta que consiguió ejecutar ciertos programas del proyecto GNU, como el *bash* (*Bourne Again Shell*) y el *gcc* (*GNU C Compiler*). Esta nueva versión de Linux sería numerada como versión 0.02, y fue la primera versión de Linux que su autor consideró mínimamente utilizable, a pesar de sus muchas limitaciones, y de suficiente nivel como para merecer ser publicada.

La publicación de la versión 0.02 de Linux tuvo lugar el 5 de octubre de 1991, en un foro de debate sobre Minix (en concreto en *comp.os.minix*). Linus distribuyó de forma gratuita en este foro el código fuente de Linux, invitando a la participación active en el proyecto Linux de cualquier persona con interés en el tema y, por supuesto, con los suficientes conocimientos de programación como para examinar el código y aportar mejoras o añadidos al sistema, así como para apuntar posibles errores o nuevas ideas. Esta llamada a la colaboración tuvo un eco tal que, a partir de este punto, podemos decir que Linux dejó de ser obra exclusiva de Linus Torvalds para pasar a ser el fruto del trabajo colectivo de un equipo formado por un buen número de colaboradores voluntarios.

### 1.3 El esfuerzo común

Es un hecho innegable que el desarrollo de Linux no hubiera sido posible sin el trabajo y el tiempo dedicados al sistema por parte de los miles de programadores que, antes o después, aceptaron la invitación para colaborar en el proyecto Linux a través de la red. Todos ellos, en mayor o menor medida, pero siempre de forma altruista, han aportado al sistema sus conocimientos, sus ideas y grandes dosis de entusiasmo. El trabajo en paralelo de toda esta comunidad permitió completar, depurar y mejorar las distintas versiones de Linux a un ritmo imposible de conseguir para un solo programador.

Un dato que revela la importancia de esta colaboración es que se estima que Linus Torvalds no ha escrito más allá del 5 por 100 del código total de Linux (aunque esta estimación no es sencilla, dadas las múltiples versiones y correcciones del sistema). Este hecho no significa que Linus Torvalds perdiera protagonismo en el proceso, sino al contrario: Linus siguió asumiendo el papel principal en la dirección y coordinación del proyecto. También comenzó a adquirir (quizá en contra de su voluntad) cierta fama de rebelde antisistema.

Evidentemente, desarrollar un proyecto en estas circunstancias exige no sólo voluntad e ideas, si no también que las distintas aportaciones al sistema se realicen de forma ordenada y coordinada. Es posible que las mejoras realizadas por un colaborador no sean realmente acertadas o necesarias (o ni siquiera puedan ser consideradas mejoras). También es posible que varios colaboradores trabajen simultáneamente en solucionar un mismo problema, solapándose así su trabajo. Para evitar estos problemas, un grupo de colaboradores (Linus Torvalds incluido), considerados de suficiente nivel y prestigio, asumió las tareas de seleccionar las aportaciones realmente relevantes y de coordinar mínimamente el trabajo de los distintos colaboradores (sin llegar nunca a vetar cualquier tipo de colaboración).

Gracias a este esfuerzo común, Linux evolucionó de forma exponencial, sucediéndose rápidamente las versiones 0.03, 0.10, 0.11 y 0.12. En marzo de 1992 se produjo un avance considerable, alcanzándose la versión 0.95. En la misma época comenzaron a desarrollarse aplicaciones diseñadas específicamente para Linux. El sistema ya podía usarse con el entorno gráfico XWindow, facilitándose así el manejo y empleo del sistema desde el punto de vista del usuario final. Poco después, llegaba la versión de Linux 0.96. Esto indicaba que Linux se aproximaba a una versión verdaderamente complete, o, al menos, así lo consideraba la comunidad de programadores implicados en el proyecto.

Sin embargo, esto no sucedería hasta pasado cierto tiempo, ya que la primera versión complete y sin errores de Linux (la versión 1.0) no vio la luz hasta octubre de 1994. En aquel momento, algunos colaboradores enfocaron sus esfuerzos hacia la creación de documentación de calidad (y también gratuita) sobre el sistema, lo que sin dudar propició la proliferación de usuarios. A su vez, las necesidades de estos usuarios renovaron el interés en el desarrollo de aplicaciones finales y nuevos controladores de Hardware para Linux.

Ahora bien, la evolución de Linux no terminó con la publicación de la versión 1.0, sino que continuó, y continúa hoy en día, siempre mediante el trabajo en común, con la intención de perfeccionar cada vez más el sistema y de disponer de mejores y más diversas aplicaciones. La siguiente versión completa (versión 2.0) tardaría en llegar dos años más, fue anunciada el 9 de junio de 1996, e introducía considerables innovaciones respecto a sus predecesoras. Como merecedora, diremos que la máquina donde se escribieron estas líneas funciona con la versión 2.2.14 del núcleo de Linux, pero ésta no es la última versión disponible actualmente. Como vemos, la evolución del sistema ha sido y sigue siendo constante.

A lo largo de esta evolución, la cantidad de colaboradores del proyecto Linux ha crecido de forma considerable. Paralelamente, se ha producido un espectacular incremento del número de usuarios del sistema, sobre todo tras la publicación de las versiones completas. El sistema, que en un principio parecía reservado a iniciados en informática o al ámbito universitario, ha acabado llegando al gran público (bien mediante Internet, bien mediante las distribuciones comerciales). Este éxito en la evolución y difusión de Linux no hubiera sido posible sin la existencia de dos fenómenos sociales de nuestro tiempo: El auge de la filosofía del software libre (o software de libre distribución, o de código abierto) y la popularidad alcanzada a nivel mundial por Internet. A estos dos fenómenos, que se han convertido en factores fundamentales para el desarrollo de Linux, están dedicados los siguientes apartados.

#### **1.4 La filosofía de Software libre**

Tal como se explica en apartados anteriores, Linus Torvalds desarrolló Linux sin ningún ánimo de lucro, ofreciendo siempre el código fuente del programa de forma gratuita. El principal beneficio obtenido por Linus fueron las mejoras realizadas en el sistema por todos los colaboradores del proyecto, que también desempeñaban (y desempeñan) esta labor sin recibir ninguna compensación económica directa. El código fuente desarrollado por estos colaboradores para mejorar el sistema también se distribuye abiertamente. Esta forma de actuación se ajusta plenamente a los postulados de la filosofía del software libre, a la que Linus era (y sigue siendo) ideológicamente afín.

El software libre, también conocido como software de libre distribución o software de código abierto (aunque esta última denominación sólo debe ser admitida con ciertas reservas), es un concepto que ha arraigado con fuerza en la mentalidad de numerosos programadores, originando un movimiento social (y casi político) de consecuencias impredecibles. Sin olvidar que se trata de un fenómeno complejo y rico en matices, trataremos de exportar sus planteamientos más esenciales. Ante todo, es necesario aclarar que la denominación inglesa de esta idea («*Free Software*») ha originado ciertas confusiones y mal interpretaciones a causa de la ambigüedad de la palabra «*free*», que en inglés significa tanto «libre» como «gratis». Realmente, un programa se considera software libre para un usuario si éste, además de poder ejecutarlo, puede modificarlo para adaptarlo a sus necesidades (o sea: el usuario tiene acceso al código fuente del programa) y puede distribuir copias tanto de la versión original del programa como de las versiones modificadas del mismo. Esta distribución puede

realizarse gratis o no, pero siempre con el objetivo de que el programa acabe mejorando y evolucionando mediante las distintas modificaciones realizadas. Evidentemente, estos planteamientos chocan de frente con la mentalidad clásica de comercialización del software, en la que siempre se cobra una cantidad al usuario por el uso del programa y nunca se le permite el acceso al código fuente del mismo, considerándose además delito de piratería informática la realización y distribución de copias del programa.

Siguiendo esta revolucionaria filosofía de forma un tanto radical, cuando Linus hizo público el código de las primeras versiones de Linux, lo hizo gratuitamente y según los términos de una licencia restrictiva que impedía cualquier tipo de distribución comercial del programa. A principios de 1992, Linus suavizó su postura y puso el sistema bajo los términos de una licencia ligeramente menos restrictiva conocida como licencia GPL (*General Public License*) para programas. Esta licencia sí permite la distribución comercial del programa, pero siempre de manera que éste, haya sido o no modificado, debe seguir sujeto a las normas del software libre. Es decir, cuando el autor de un programa lo distribuye bajo los términos de la licencia GPL, permite que el programa sea ejecutado, modificado, copiado y distribuido (incluyendo su venta), pero no permite que el programa original o las versiones modificadas del mismo dejen de ser software libre. Conviene no olvidar que existe otra licencia GPL para librerías, llamada LGPL (*Library General Public License*) que establece normas ligeramente distintas a la licencia de programas.

Con esto queda claro que las distribuciones comerciales de Linux, que actualmente gozan de cierto éxito, no contradicen las ideas propugnadas por el software libre (esta acusación ha sido realizada contra ellas en alguna ocasión). Además, la cantidad que hay que pagar actualmente para adquirirlas es relativamente modesta: el distribuidor realmente cobra el soporte físico de los datos y el trabajo de recopilación de software. Y, por supuesto, el comprador obtendrá software libre, teniendo acceso al código y pudiéndolo modificar, copiar, etc.

La licencia GPL es el reflejo de la idea conocida como «copyleft». A diferencia del «copyright» tradicional, el «copyleft» permite al autor del programa ser reconocido como tal, pero no para obtener beneficio o imponer restricciones que impidan que el programa sea modificado, copiado o distribuido, sino para impedir que otra persona se apropie del programa (total o parcialmente) e imponga aquellas restricciones sobre su uso que el mismo autor no ha impuesto. Puesto que la licencia GPL establece que las versiones modificadas de los programas de software libre deben seguir siendo software libre, la comunidad de programadores que colabora en un proyecto bajo licencia GPL se beneficia inmediatamente de cualquier mejora que se realice en el programa, lo que favorece su evolución y depuración. El mejor ejemplo de que este planteamiento no es sólo una teoría es sin duda el caso del propio Linux.

Las licencias GPL (para programas y para librerías) son unos de los muchos frutos del proyecto GNU, que ya hemos mencionado en el apartado anterior. El objetivo concreto de este proyecto es desarrollar un sistema operativo completo, similar a Unix, pero sin ser Unix (GNU viene de *GNU's Not Unix*), y que sería distribuido como software libre.

No se trata del único proyecto con este objetivo: en la Universidad de Berkeley se ha venido desarrollando, paralelamente al sistema GNU y con idénticos planteamientos, el sistema operativo BSD. El proyecto GNU está promovido desde 1984 por Richard Stallmann, que es otro de los adalides del movimiento del software libre (e incluso más radical que Linus en sus planteamientos). No en vano, Stallmann es uno de los fundadores de la FSF (*Free Software Foundation*), una fundación destinada a promover el desarrollo y uso de software libre y recabar ayudas (incluyendo aportaciones económicas y donaciones de material informático) para continuar adelante con el propio proyecto GNU.

El proyecto GNU y Linux están relacionados más allá de lo meramente ideológico. Ya hemos mencionado que Linus Torvalds comprobó el funcionamiento de sus primeras versiones de Linux compilando y ejecutando determinados programas GNU como el *bash* o el *gcc*. En aquella época, el proyecto GNU no había conseguido desarrollar un núcleo estable para su sistema operativo, pero sí una buena cantidad de aplicaciones de gran utilidad para el sistema (compiladores, *shells*, etc.), que habían adquirido cierta popularidad. Dado que Linux pronto demostró su funcionalidad, la elección de éste como núcleo del sistema GNU era evidente. Desde entonces, se estableció una asociación natural entre Linux y el software desarrollado por el proyecto GNU que ha continuado hasta la fecha. Obviamente, esta asociación se ha visto favorecida por el hecho de que tanto Linux como los programas GNU son software libre. Hoy en día, prácticamente la totalidad de usuarios de Linux emplea también aplicaciones GNU en sus sistemas. De hecho dichos sistemas han sido denominados en ocasiones «sistemas GNU/Linux», para destacar que el software GNU empleado habitualmente, aunque no forme parte del núcleo, sí resulta fundamental para el funcionamiento del sistema global. Aunque esta denominación ha sido empleada en ciertos círculos, actualmente existe una cierta tendencia a referirse simplemente a «sistemas Linux». Esta tendencia se justifica por el hecho de que no todo el software empleado en sistemas Linux es software GNU, pero también por cierto desconocimiento de la realidad del sistema por parte de muchos usuarios, que cotidianamente emplean software GNU sin ser conscientes de ello.

## **1.5 La expansión en Internet**

A estas alturas, y dada su popularidad mundial, existen pocos aspectos de Internet que no hayan sido explicados, comentados y valorados en la multitud de libros, estudios, artículos y reportajes realizados sobre el tema. Teniendo esto en cuenta, el objetivo de este apartado no debe ser explicar los fundamentos técnicos de la «red de redes» ni analizar su innegable impacto social, sino resaltar su importancia en el proceso de evolución y popularización de Linux.

La importancia de Internet en este proceso es evidente si recordamos que Linux, al menos desde cierto punto de su existencia, ha evolucionado como una obra colectiva. La filosofía de software libre propició este tipo de desarrollo y la licencia GPL ofreció incluso una base legal para el mismo, pero es evidente que nada de esto hubiera servido si no existiera un medio para que todas las personas involucradas, en mayor o menor grado, en el proyecto Linux (comenzando por el propio Linus Torvalds)

podieran compartir sus mejoras, opiniones y correcciones al sistema. Internet, y, más concretamente, el correo electrónico y los foros de debate, constituyeron los medios idóneos para la comunicación entre los colaboradores del proyecto.

Ya mencionamos que el «nacimiento» de Linux en la red se produjo en un foro de debate sobre Minix. El interés despertado por Linux fue tal que, en poco tiempo, el sistema ya contaba con varios foros de debate propios, en los que tuvieron lugar buena parte de las deliberaciones que hicieron evolucionar el sistema. Al mismo tiempo, la red permitía que el código fuente de cualquier contribución a Linux estuviera disponible inmediatamente para ser examinado o probado por cualquiera, o para ser incorporado al sistema una vez aprobada su validez.

Por otra parte, una vez que el sistema estuvo mínimamente completo, comenzaron a aparecer las primeras páginas Web dedicadas a Linux. La proliferación de estas páginas y la popularidad alcanzada por Internet en todo el mundo han permitido que el número de personas que entran en contacto con Linux mediante la red crezca día a día. Este hecho, unido a la disponibilidad gratuita de Linux, ha originado que el número de usuarios del sistema aumente hasta alcanzar, hoy en día, las decenas de millones. Es decir, Internet no sólo ha sido el medio que ha permitido (y permite) la construcción y evolución de Linux, sino que también constituye actualmente el principal medio para la difusión del sistema. Sin duda, esta expansión ha contribuido de forma decisiva a afianzar definitivamente la posición de Linux como sistema operativo completamente funcional. No importa que los recién llegados al sistema aporten o no algún tipo de mejora concreta al proyecto global: cualquier usuario que instale Linux en su máquina y opere con él ya estará comprobando implícitamente la funcionalidad del sistema, efectuando así cierto «control de calidad».

Hoy en día, los usuarios de Linux contemplan la red no sólo como una fuente para obtener el código del núcleo del sistema y del software asociado, sino también como el mejor lugar donde buscar una solución a las dudas o problemas planteados por el sistema. Precisamente uno de los defectos más frecuentemente achacados a Linux es la carencia de un servicio técnico «oficial» que responda de cualquier fallo del sistema (al modo de las grandes compañías de software comercial), pero este hecho pierde importancia si se considera que el soporte brindado espontáneamente, a través de la red, por la comunidad Linux mundial puede ser. Muy probablemente, mejor y más rápido que el ofrecido por cualquier gran compañía comercial. Por último, recordar que aunque las distribuciones comerciales de Linux permiten acceder al código fuente del sistema sin necesidad de obtenerlo a través de la red, también recurren a ella para ofrecer a sus usuarios actualizaciones, correcciones y documentación.



## 2. Distribuciones de Linux

Como tal, Linux no es un único Sistema Operativo. Lo único fijo, es el kernel. El resto de las aplicaciones, variantes, programas de correo, de edición (como el StarOffice 5.0, el Wordperfect 8, klyx/lyx, o L<sup>A</sup>T<sub>E</sub>X, con el que está escrito este documento), forman parte de la grandísima cantidad de programas disponibles para Linux, incluyendo de empresas comerciales, como Caldera, Oracle, Stardivision, etc. que ya tienen versiones Linux de sus programas más importantes o están en proceso de sacarlas al mercado, con una cláusula que las hace gratuitas para uso personal (similar a la licencia adoptada por Netscape). Si bien el ámbito de donde provienen la mayoría de las aplicaciones distribuidas para Linux es el proyecto GNU, ya explicado, por lo que suele llamársele con propiedad el Sistema Operativo GNU/Linux. Es por ello que existen distintos grupos de gente, ahora ya empresas con unos ingresos algo sustanciales, que se dedican a compilar los programas que ellos consideran más interesantes, y ofrecerlo conjuntamente con Linux en lo que se conoce como una distribución de Linux. Cabe destacar un par de ellas.

La desconcertante elección entre un número siempre creciente de distribuciones de Linux puede crear confusión entre aquellos nuevos en Linux. Se listan a continuación las 10 distribuciones que generalmente se consideran las más extendidas entre los usuarios de Linux de todo el mundo. Hay otras muchas que probar, y algunas pueden ser mas adecuadas para tus necesidades, pero como norma general, las presentes son populares por lo que tienen foros y listas de correo muy activas donde preguntar dudas si existiesen.

Lycoris, Xandros y Lindows son consideradas las mejores para aquellos usuarios nuevos en Linux, que quieren empezar a ser productivos con Linux lo antes posible sin tener que aprender todas sus complejidades. En el lado opuesto del espectro se encuentran Gentoo, Debian y Slackware, que son distribuciones más avanzadas que requieren un completo aprendizaje antes de poder ser usadas eficientemente. Mandrake, Red Hat y SuSE se encuentran a medio camino entre ambas. Knoppix es un caso aparte, es genial para probar Linux sin tener que hacer nada, ya que funciona directamente del CD, si ninguna instalación. Estas distribuciones no están listadas de acuerdo a su cuota de mercado o calidad.

- Debian GNU/Linux
- Gentoo Linux
- LindowsOS
- Lycoris Dekstop/LX
- Knoppix
- Mandrake Linux
- Red Hat Linux
- Slackware Linux
- SuSE Linux
- Xandros OS
- Libranet

## 2.1 Mandrake

Mandrake Linux, creada por Gaël Duval, es una distribución que ha experimentado un enorme aumento de popularidad desde su primera versión de julio de 1998. Los desarrolladores partieron de la distribución de Red Hat, cambiaron el entorno de escritorio predeterminado por KDE, y añadieron un instalador fácil de usar rompiendo el mito de que Linux es difícil de instalar. Las herramientas de detección de hardware de Mandrake y sus programas para la partición de discos son consideradas por muchos como las mejores de la industria, y muchos usuarios se encontraron usando Mandrake cuando otras distribuciones no habían conseguido entregar el desempeño necesario. Desde entonces Mandrake Linux ha madurado y se ha convertido en una distribución popular entre los nuevos usuarios de Linux y aquellos hogares que buscan un sistema operativo alternativo. El desarrollo de Mandrake es completamente abierto y transparente, con paquetes nuevos que se añaden al directorio llamado "cooker" a diario. Cuando una nueva versión entra en fase beta, la primera beta se crea a partir de los paquetes que se encuentran en "cooker" en ese momento.

El proceso de pruebas de la beta solía ser corto e intensivo, pero desde la versión 9.0 ha pasado a ser más largo y exigente. Las listas de correo sobre la versión beta suelen estar saturadas, pero sigue siendo posible recibir una respuesta rápida sobre cualquier fallo o duda. Como resultado de este tipo de desarrollo se obtiene una distribución puntera y altamente actualizada. Como contrapartida, los usuarios pueden encontrarse con más fallos que en otras distribuciones. Mucha gente encuentra este 'pero' razonable para sus equipos, se obtienen las últimas versiones de software y los problemas ocasionales de las aplicaciones son un riesgo calculado. Tan pronto como el desarrollo se completa, el software se pone a la libre disposición de la gente desde réplicas en todo el mundo.

- Pros: Amigable para el usuario, herramientas de configuración gráfica, enorme soporte de la comunidad, posibilidad de cambiar el tamaño de particiones NTFS.
- Contras: Algunas versiones contienen fallos, la compañía está pasando por problemas financieros.
- Sistema de paquetes: RPM
- Descarga Gratuita: Si

## 2.2 Red Hat

Para gran mayoría de la comunidad de los SO, el nombre de Red Hat equivale a Linux, ya que probablemente se trata de la compañía de Linux más popular del mundo. Fundada en 1995 por Bob Young y Marc Ewing, Red Hat Inc. solo ha mostrado beneficios recientemente gracias a otros servicios en lugar de a la distribución en sí. Aun y así, Red Hat es la primera elección para muchos profesionales y parece que seguirá siendo un peso pesado durante mucho tiempo. Afortunadamente se resistieron a realizar ningún plan de rápida expansión durante el boom de las punto-com durante

los años 1998 - 1999, concentrándose en su negocio principal. Este tipo de gestión prudente, de continuar, es propensa a garantizar estabilidad y dependencia.

Lo que hace a Red Hat tan popular es su curiosa mezcla de conservadurismo y paquetes punteros mezclados sobre muchas aplicaciones desarrolladas en casa. Los paquetes no son los más actuales, una vez se anuncia una nueva versión beta, las versiones de los paquetes se mantienen, excepto para actualizaciones de seguridad. Como resultado se obtiene una distribución bien probada y estable. El programa de betas y las facilidades para enviar fallos están abiertas al público y hay un gran espíritu en las listas de correo públicas. Red Hat Linux se ha convertido en la distribución Linux dominante en servidores en todo el mundo.

Otra de las razones del éxito de Red Hat es la gran variedad de servicios populares que ofrece la compañía. Los paquetes de software son fácilmente actualizables usando la Red Hat Network, un repositorio oficial de software e información. Una larga lista de servicios de soporte son accesibles en la compañía y, aunque no siempre baratos, se asegura un excelente soporte de personal altamente calificado. La compañía ha desarrollado incluso un programa de certificación para popularizar su distribución, el RHCE (Certificado de Ingeniería de Red Hat), academias y centros examinadores están disponibles en el casi todas partes del mundo. Todos estos factores han contribuido a que Red Hat sea una marca reconocida en el mundo de la industria de las TI.

- Pros: Ampliamente usada, excelente soporte de la comunidad, muchas innovaciones.
- Contras: Limitada vida útil de la edición gratuita, soporte multimedia pobre.
- Sistema de paquetes: RPM
- Descarga Gratuita: Si

### **2.3 Gentoo**

Gentoo Linux fue creada por Daniel Robbins, un conocido desarrollador de Stampede Linux y FreeBSD. Fue el contacto del autor con FreeBSD y su función de autobuild llamada "ports" lo que le inspiró a incorporar los "ports" en Gentoo bajo el nombre de "portage". La primera versión estable de Gentoo fue anunciada en Marzo del 2002. Gentoo Linux es una distribución basada en código fuente, la única en esta lista.

Mientras que los sistemas de instalación proveen de varios niveles de paquetes pre-compilados, para obtener un sistema Linux básico funcionando, el objetivo de Gentoo es compilar todos los paquetes de código en la máquina del usuario. La principal ventaja de esto es que todo el software se encuentra altamente optimizado para la arquitectura de tu computadora. También, actualizar el software instalado a una nueva versión es tan fácil como teclear un comando, y los paquetes, mantenidos en un repositorio central, se mantienen bastante actualizados. En la otra cara de la moneda, instalar Gentoo y convertirla en una distribución completa, con los últimos entornos gráficos, multimedia y de desarrollo es un trabajo largo y tedioso, de varios días, incluso en una máquina rápida.

- Pros: Fácil instalación de paquetes de software individuales, altamente actualizada, la autosuficiencia de crear tu propia distribución atendiendo a las necesidades del usuario.
- Contras: Instalación larga y tediosa, ocasionalmente inestable y con riesgos de romperse, no aconsejada para servidores con funciones críticas.
- Sistema de paquetes: SRC
- Descarga gratuita: Si

### 2.3 Debian

Debian GNU/Linux inició su andadura de la mano de Ian Murdock en 1993. Es un proyecto totalmente no-comercial; posiblemente el más puro de los ideales que iniciaron el movimiento del software libre. Cientos de desarrolladores voluntarios de alrededor del mundo contribuyen al proyecto, que es bien dirigido y estricto, asegurando la calidad de una distribución conocida como Debian.

En cualquier momento del proceso de desarrollo existen tres ramas en el directorio principal: "estable", "en pruebas" e "inestable" (también conocida como "sid"). Cuando aparece una nueva versión de un paquete, se sitúa en la rama inestable para las primeras pruebas, si las pasa, el paquete se mueve a la rama de pruebas, donde se realiza un riguroso proceso de pruebas que dura muchos meses. Esta rama solo es declarada estable tras una muy intensa fase de pruebas. Como resultado de esto, la distribución es posiblemente la más estable y confiable, aunque no la más actualizada.

Mientras que la rama estable es perfecta para servidores con funciones críticas, muchos usuarios prefieren usar las ramas de pruebas o inestable, más actualizadas, en sus ordenadores personales. Debian es también famosa por su reputación de ser difícil de instalar, a menos que el usuario tenga un profundo conocimiento del hardware de la computadora. Compensando este fallo está "apt-get", un instalador de paquetes Debian. Muchos usuarios de Debian hacen bromas sobre que su instalador es tan malo por que solo lo han de usar una vez, tan pronto como Debian está en funcionamiento, todas las actualizaciones, de cualquier tipo pueden realizarse mediante la herramienta apt-get.

- Pros: 100% libre, web y recursos de la comunidad excelentes, bien probada, instalación de software sencillísima usando apt-get.
- Contras: Instalador arcaico, la versión estable no está actualizada.
- Sistema de paquetes: DEB
- Descarga gratuita: Si

### 2.4 SuSE

SuSE es otra compañía orientada a los escritorios, aunque variedad de otros productos para empresas están disponibles. La distribución ha recibido buenas críticas por su instalador y la herramienta de configuración YaST, desarrollada por los desarrolladores de la propia SuSE. La documentación que viene con las versiones

comerciales, ha sido repetidas veces evaluada como la más completa, útil y usable con diferencia a la de sus competidores. SuSE Linux 7.3 recibió el premio "Producto del año 2001" que entrega el Linux Journal. La distribución tiene un gran porcentaje de mercado en Europa y América del norte, pero no se vende en Asia y otras partes del mundo.

El desarrollo de SuSE se realiza completamente a puerta cerrada, y no se lanzan betas públicas para probar. Siguen la política de no permitir descargar el software hasta tiempo después de que salgan a la venta las versiones comerciales. A pesar de todo, SuSE no entrega imágenes ISO de fácil instalación de su distribución, usando el software empaquetado para la gran mayoría de su base de usuarios.

- Pros: Atención profesional en cada detalle, herramienta de configuración de fácil uso (YaST).
- Contras: Solo disponible en algunas partes del mundo en las tiendas de software o mediante instalación FTP, incluye componentes propietarios, que no permiten su redistribución.
- Sistema de paquetes: RPM
- Descarga gratuita: SuSE no proporciona imágenes ISO para descarga, no obstante la versión Profesional de su distribución es accesible para la instalación FTP normalmente 1 o 2 meses más tarde de la versión oficial. La instalación mediante FTP no es difícil, pero requiere una buena conexión.

## **2.5 Knoppix**

Desarrollada por Klaus Knopper en Alemania, es seguro que esta distribución basada en Debian ha puesto muy alto el listón, especialmente con su detección automática de hardware que deja a la altura del barro la de muchas distribuciones comerciales. Su arranque automático, gran cantidad de software, su sistema de descompresión al vuelo y la posibilidad de instalarlo al disco duro han convertido a Knoppix en una herramienta indispensable. Puede ser usada como un disco de rescate, una herramienta para enseñar Linux para aquellos que no lo han visto o una herramienta para probar una nueva computadora antes de comprarla. También puede ser usada como una completa distribución Linux para el uso diario. Frecuentemente se lanzan nuevas versiones de Knoppix, aproximadamente, una nueva versión cada 1 o 2 semanas. Las actualizaciones incluyen parches de fallos, así como el último software de la rama inestable de Debian. Se discute mucho sobre Knoppix en los canales de Linux en la red.

- Pros: Excelente auto detección de hardware, funciona directamente del CD sin instalación en el disco duro, puede ser usada como herramienta de recuperación.
- Contras: Bajo rendimiento y velocidad si se utiliza directamente del CD.
- Sistema de paquetes: DEB
- Descarga gratuita: Si

## 2.6 Slackware

Creada por Patrick Volkerding en 1992, Slackware Linux es la distribución más antigua que sobrevive hoy en día. No ofrece extras vistosos, y se mantiene con un instalador basado en texto, y sin herramientas de configuración gráfica. Mientras otras distribuciones intentan desarrollar interfaces fáciles de usar para muchas utilidades comunes, Slackware no ofrece nada amistoso, y toda la configuración se realiza mediante los archivos de configuración. Es por esto que Slackware solo se recomienda a aquellos usuarios nuevos que deseen perder el tiempo aprendiendo acerca de Linux.

A pesar de todo, Slackware tiene una especie de aura mágica para muchos usuarios. Es extremadamente estable y segura, muy recomendada para servidores. Los administradores con experiencia en Linux encuentran que es una distribución con pocos fallos, ya que usa la mayoría de paquetes en su forma original, sin demasiadas modificaciones propias de la distribución, que son un riesgo potencial de añadir nuevos fallos. Es raro que se produzcan lanzamientos de nuevas versiones (aproximadamente una al año), aunque siempre se pueden encontrar paquetes actualizados para descargar después del lanzamiento oficial. Slackware es una buena distribución para aquellos interesados en profundizar en el conocimiento de las entrañas de Linux. Se considera que, posiblemente, la mejor característica de esta distribución es que los usuarios de Slackware pueden resolver casi cualquier problema, aún de otras distribuciones.

- Pros: Alta estabilidad y ausencia de fallos, sigue fielmente los principios de UNIX.
- Contras: Toda la configuración se realiza mediante la edición de ficheros de texto, auto detección de hardware limitada.
- Sistema de paquetes: TGZ
- Descarga gratuita: Si

## 2.7 Lycoris

Lycoris, conocida formalmente como Redmond Linux, fue fundada por Joseph Cheek, CEO de la compañía, y trabajador de Linuxcare y Microsoft. El objetivo principal era crear una instalación fácil de usar para realizar la transición entre Windows y Linux lo más fácilmente posible. Para realizar esto, Lycoris Desktop/LX tiene cientos de modificaciones, incluyendo un clon de Mi PC y del Entorno de Red, así como un atractivo tema que se parece mucho a Windows XP. Esta distribución se basa actualmente en Caldera OpenLinux. El número de paquetes de Lycoris Desktop/LX es bastante reducido, normalmente una aplicación por tarea y el Panel de Control para las tareas comunes de administración. Se le ha criticado que algunos paquetes están obsoletos, a lo que sus desarrolladores responden que su objetivo es hacer una distribución fácil de usar.

- Pros: Amigable para principiantes, diseñada para parecerse a Windows.
- Contras: Contiene paquetes obsoletos, requiere una licencia para uso comercial.
- Sistema de paquetes: RPM
- Descarga gratuita: Si

## **2.8 Xandros**

Xandros nació de las cenizas de Corel Linux, un fallido intento de crear un Linux para las masas del año 1999, pero abandonado poco tiempo después cuando la compañía cayó en dificultades financieras. Xandros compró la distribución en agosto del 2001 y lanzó su primera y única versión en octubre del 2002. Xandros Desktop es sin lugar a dudas la distribución más sencilla de usar del mercado, y altamente recomendable para los nuevos usuarios de Linux.

Mientras que el número de aplicaciones incluidas en la distribución es limitado, los desarrolladores se han asegurado de que funciona perfectamente. Xandros provee cantidad de mejoras únicas así como su propio administrador de archivos y otros componentes propietarios. Hay dos ediciones disponibles, la Estándar (40\$) y la versión De Lujo (99\$). La única diferencia entre ambas es que en la versión de lujo se incluye el plugin CrossOver Office (que permite usar las aplicaciones de Microsoft Office y otros productos de Windows en Linux) y una herramienta de partición y cambio de tamaño de particiones NTFS (para los usuarios de WindowsXP).

- Pros: Diseñada para principiantes, funciona al instante, excelente administrados de archivos y otras utilidades.
- Contras: Incluye software propietario, lo que no permite su redistribución, no hay descarga gratuita.
- Sistema de paquetes: DEB
- Descarga gratuita: No

## **2.9 Lindows**

Lindows.com fue iniciado por Michael Robertson, fundador y CEO de MP3.com, en octubre del 2001, en San Diego, EEUU. El objetivo inicial era desarrollar un SO basado en Linux capaz de utilizar no solo las aplicaciones de Linux, sino también las principales herramientas de Windows tales como MS Office. Este ambicioso objetivo fue abandonado ya que los desarrolladores de LindowsOS parecían haber subestimado el esfuerzo necesario para cumplirlo. La primera versión hecha pública de LindowsOS fue anunciada en Noviembre del 2002 bajo el nombre de LindowsOS 3.0.

Las principales características del producto son una instalación rápida y sencilla, y Click-N-Run, una infraestructura para instalar software del repositorio de la compañía. Las opiniones sobre el producto son para todos los gustos, parece ser que Click-N-Run todavía necesita mucho trabajo para pulir sus fallos. La enorme campaña de marketing puede resultar agobiante, pero han conseguido que LindowsOS (y Linux)

aparezcan en los principales medios y que vengan preinstalados en ordenadores nuevos de tiendas de EEUU y Reino Unido.

- Pros: Diseñado para principiantes, instalación rápida y fácil, instalación de software con un click.
- Contras: Cara, cota anual para acceder al repositorio de software, prácticas comerciales preocupantes.
- Sistema de paquetes: DEB
- Descarga gratuita: No

## **2.10 Libranet**

Libranet es una distribución canadiense, que sin ser una de las grandes, tiene potencial para estar algún día en todo lo alto. Es una distribución comercial basada en Debian, que añade a las excelentes herramientas y características de Debian; otras nuevas como un excelente sistema de detección de hardware, versiones más actuales de los paquetes de software, un sistema de administración sencillo de usar, y potentes paquetes de configuración de software. La última versión disponible es la 2.8, consiste de dos CDs llenos de software que se pueden adquirir desde sólo 39 dólares (licencia de estudiantes).

Disponible para descarga gratuita tenemos la versión 2.0 en [www.Linuxiso.org](http://www.Linuxiso.org) Libranet es una distribución centrada en la velocidad, es por esto, que a diferencia de otras distribuciones basadas en Debian (como las ya comentadas Lindows y Xandros) en Libranet no contiene un sistema gráfico sobrecargado. Por otro lado, el sistema de instalación ofrece un buen equilibrio entre simplicidad y opciones de configuración, así como una buena cantidad de software disponible. Pero si todavía no es suficiente, pueden usarse las herramientas de administración de Debian y sus repositorios de paquetes usando apt-get, y obtener un sistema actualizado. Solo tiene un contra y es que a diferencia de Debian, Libranet no es libre. A pesar de todo, es una distribución que gana adeptos cada día, y muy a tener en cuenta para un futuro próximo.

- Pros: Herramientas de Debian, instalación sencilla, auto detección de hardware, velocidad, compatible 100% con Debian.
- Contras: No es libre, Versión de descarga gratuita anticuada.
- Sistema de paquetes: DEB
- Descarga gratuita: Si



## 3. Instalación de Red Hat Linux

Este capítulo explica cómo realizar una instalación personalizada de Red Hat Linux desde el CD-ROM, utilizando el programa de instalación gráfico basado en el ratón. Se tratarán los siguientes temas:

- Familiarización con la interfaz de usuario del programa de instalación
- Inicio del programa de instalación
- Elección del método de instalación
- Pasos para la configuración durante la instalación (idioma, teclado, ratón, etc.)
- Finalizar la instalación

### 3.1. Interfaz gráfica de usuario del programa de instalación

Si ha utilizado la *interfaz de usuario en modo gráfico (GUI)* con anterioridad, este proceso le será muy familiar. Use el ratón para navegar por las pantallas, pulse los botones o rellene los campos de texto. Puede también utilizar las teclas [Tab] e [Intro] para navegar.



#### Nota

Si no se desea utilizar el programa de instalación GUI, está disponible el programa de instalación en modo texto. Para utilizarlo, se teclea el siguiente comando de arranque en el prompt de instalación:

```
boot:text
```

### 3.2 Nota sobre las consolas virtuales

El programa de instalación de Red Hat Linux ofrece más que las ventanas de diálogo del proceso de instalación. Además de ofrecer la posibilidad de insertar comandos desde el intérprete de comandos del shell, hay en disponibilidad muchos tipos diferentes de mensajes. El programa de instalación despliega estos mensajes en cinco *consolas virtuales*, entre las que puede cambiarse usando una combinación de teclas.

Estas consolas virtuales pueden ser de ayuda en el caso de existir problemas durante la fase de instalación de Red Hat Linux. Los mensajes visualizados durante la instalación o en las consolas del sistema, pueden señalar el problema. En la Tabla 3.a se encuentra la lista de las consolas virtuales, las combinaciones de teclas para cambiar de una de ellas a otra y sus contenidos.

En general, no hay ninguna razón para dejar la consola predeterminada (consola virtual #7) a no ser que exista intención de detectar problemas de instalación.

Consola	Combinación de teclas	Contenido
1	[Ctrl]-[Alt]-[F1]	diálogo de instalación
2	[Ctrl]-[Alt]-[F2]	intérprete de comandos de la shell
3	[Ctrl]-[Alt]-[F3]	Log de instalación (mensajes del programa de instalación)
4	[Ctrl]-[Alt]-[F4]	mensajes de sistema
5	[Ctrl]-[Alt]-[F5]	otros mensajes
7	[Ctrl]-[Alt]-[F7]	pantalla gráfica de X

**Tabla 3.a Consolas, combinaciones de teclas y contenidos**

### 3.3 Arranque del programa de instalación

El programa de instalación puede ser arrancado usando cualquiera de los siguientes medios (en función del medio compatible con el sistema):

- *CD-ROM de arranque*
- *Disquete de arranque.*

Se introduce el disquete o CD-ROM de arranque en la primera unidad de disco y se reinicia la computadora. Debe tenerse cuidado de especificar la unidad utilizada como inicial en el orden de arranque del BIOS. Después de una breve pausa aparecerá una pantalla con el indicador de comandos boot:. Esta pantalla contiene información sobre algunas opciones de arranque. Cada una de estas opciones está asociada a una pantalla de ayuda. Para acceder a ella, se pulsa la tecla de función correspondiente que se encuentra en la lista de la parte inferior de la pantalla.

Al arrancar el programa de instalación, deberá tenerse en cuenta dos cuestiones:

- Una vez que vea el indicador de comandos boot:, la pantalla inicial lanzará automáticamente el programa de instalación si no se realiza alguna acción durante un minuto. Para inhabilitar esta característica, pulsar una de las teclas de función de la pantalla de ayuda es suficiente.
- Al pulsar una tecla de función de una pantalla de ayuda, es necesario esperar unos segundos para la lectura desde el medio elegido para el arranque.

Normalmente, habrá de pulsarse simplemente [Intro] para empezar. Es pertinente observar los mensajes de arranque para ver si el kernel de Linux detecta su hardware.

Si el hardware no ha sido detectado de forma correcta, se reinicia en modo experto. Si el hardware se detecta adecuadamente, se puede continuar con la siguiente sección.

### 3.4 Opciones de arranque adicionales

Aunque para un usuario resulta más sencillo arrancar desde un CD-ROM y ejecutar una instalación gráfica, en ocasiones hay instalaciones en las que puede ser necesario realizar el arranque con un método distinto. En esta sección se describen las opciones de arranque adicionales que hay disponibles para el sistema Red Hat Linux.

- Si no se desea realizar una instalación gráfica, puede iniciarse una instalación en modo texto mediante el siguiente comando de arranque:

```
boot: Linux text
```

- Las imágenes ISO ahora tienen un md5sum integrado. Para probar la integridad checksum de una imagen ISO, en el indicador de comandos de arranque de la instalación, se escribe:

```
boot: Linux mediacheck
```

- El programa de instalación solicitará que se inserte un CD o que se seleccione la imagen ISO que se desea verificar. Esta operación de checksum se puede realizar en cualquier CD-ROM de Red Hat Linux y no tiene que realizarse en un orden específico (por ejemplo, el CD-ROM número 1 no tiene que ser el primer CD-ROM que verifique). Se recomienda realizar esta operación en cualquier CD-ROM de Red Hat Linux que se haya creado a partir de la descarga de imágenes ISO. Este procedimiento es válido con instalaciones basadas en CD-ROM y con instalación de disco duro y NFS usando imágenes ISO.
- Si el programa de instalación no detecta correctamente el hardware instalado, probablemente deberá reiniciar la instalación en modo *experto*. Para usar el modo experto, se escribe el siguiente comando de arranque:

```
boot: Linux noprobe
```

- Para ejecutar la instalación en modo texto, se utiliza:

```
boot: Linux text noprobe
```

- El modo experto desactiva las operaciones de pruebas del hardware y ofrece la posibilidad de introducir opciones para los controladores que se cargan durante la instalación.

- Si se necesita realizar una instalación en *modo serie*, se escribe el siguiente comando:

```
boot: Linux console=<dispositivo>
```

- Para ejecutar la instalación en modo texto, se utiliza:

```
boot: Linux text console=<dispositivo>
```

- En el comando anterior, *<dispositivo>* es el dispositivo que está utilizando (como ttyS0 o ttyS1). Por ejemplo, Linux text console=ttyS0,115200n8.

### 3.5 Opciones del kernel

Estas opciones se pueden transmitir directamente al kernel. Si por ejemplo, se desea que el kernel utilice toda la RAM en un sistema de 128 MB, se introduce:

```
boot: Linux mem=128M
```

Para ejecutar la instalación en modo texto, se utiliza:

```
boot: Linux text mem=128M
```

Si es necesario especificar las opciones para la identificación del hardware, es el momento de introducirlas. Las opciones de arranque serán necesarias durante una parte de la configuración del gestor de arranque.

### 3.6 Arranque sin disquetes

El CD-ROM de Red Hat Linux lo pueden ejecutar todos los ordenadores que admitan el uso de discos CD-ROM de arranque. No obstante, no todos son compatibles con esta característica. Si el sistema no puede arrancar desde un CD-ROM, existe otro modo de iniciar esta instalación sin el uso de un disquete de arranque. El siguiente método es exclusivo para los ordenadores basados en x86.

Si está instalado MS-DOS en el sistema, se puede arrancar directamente desde una unidad de CD-ROM sin usar un disquete de arranque. Para ello, (se asume que el CD-ROM está en la unidad d:), se utiliza los siguientes comandos:

```
C:\> d:
D:\> cd \dosutils
D:\dosutils> autoboot.bat
```

Este método no funciona en una ventana DOS de Windows — el archivo autoboot.bat debe ejecutarse con DOS como el único sistema operativo. En otras palabras, Windows no se puede ejecutar. Si la computadora no puede arrancar directamente desde un CD-ROM (y no puede utilizar el auto arranque basado en DOS), será necesario utilizar un disquete de arranque.

### 3.7 Selección de un método de instalación

#### CD-ROM

Si se posee un lector de CD-ROM y tiene el CD-ROM de Red Hat Linux, se puede utilizar este método. Se necesitará un disquete de arranque o un CD-ROM para arrancar. También puede usar discos de arranque PCMCIA.

#### Disco duro

Si se ha copiado las imágenes ISO de Red Hat Linux en el disco duro local, se puede utilizar este método. Se necesitará un disquete de arranque. También se pueden utilizar disquetes de controlador PCMCIA.

#### Imagen NFS

Si se está realizando la instalación desde un servidor NFS utilizando imágenes ISO o una imagen réplica de Red Hat Linux, se puede utilizar este método. Se necesitará un disquete de arranque de red. También se pueden utilizar disquetes de controlador PCMCIA. Las instalaciones NFS también se pueden realizar en el modo GUI.

#### FTP

Si está realizando la instalación directamente desde un servidor FTP, utilice este método. Necesitará un disquete de arranque de red. También se pueden utilizar disquetes de controlador PCMCIA.

#### HTTP

Si se está realizando la instalación directamente desde un servidor Web HTTP, se utiliza este método. Se necesitará un disquete de arranque de red. También se pueden utilizar disquetes de controlador PCMCIA.

### 3.8 Bienvenido a Red Hat Linux

La pantalla de **Bienvenida** no le pide ninguna información. Es conveniente leer el texto de ayuda en el panel de la izquierda para instrucciones adicionales e información sobre el registro de su producto Red Hat Linux. Obsérvese que el botón **Esconder ayuda** se

encuentra en la parte inferior izquierda de la pantalla. La pantalla de ayuda aparece abierta por defecto. Si no quiere visualizar la información, Click en **Esconder ayuda** para minimizar esta parte de la pantalla.

Click en **Siguiente** para continuar.

### 3.9 Selección del idioma

Utilizando el ratón, se elige el idioma que quiere usar por defecto para la instalación y para el sistema (**Figura 3.b**). La selección del idioma apropiado ayuda también a seguir los pasos correctos para configurar la zona horaria (huso horario), más tarde. El programa de instalación intentará definir el huso horario adecuado basándose en la localización.



**Figura 3.b Selección del idioma**

Una vez que haya seleccionado el idioma, Click en **Siguiente** para continuar.

### 3.10 Configuración del teclado

Con el ratón, elija el tipo de teclado que mejor se adapte a el sistema (Figura 3.c).

Click en **Siguiente** para continuar.



**Figura 3.c Configuración del teclado**

### 3.11 Configuración del ratón

Elija el ratón adecuado a el sistema. Si no encuentra el tipo exacto, elija el que crea que será compatible con el suyo (Figura 3.d). Para determinar la interfaz del ratón, mire el conector de su ratón y siga los siguientes diagramas. Si está instalando Red Hat Linux en un ordenador portátil, en la mayoría de los casos el dispositivo en cuestión será compatible con PS/2.



Si su ratón es serial, el puerto será parecido a

Si su ratón es un ratón PS/2, el puerto será parecido a



Si su ratón es un ratón USB, el puerto será parecido a



Si su ratón es un ratón AT, el puerto será parecido a



Si no existe un ratón del que se esté seguro que es compatible con el sistema, se selecciona una de las entradas **Genérico**, basadas en el número de botones de su ratón y de su interfaz. Si tiene un ratón de scroll, seleccione la entrada **Genérico - Wheel Mouse** (con el puerto del ratón correcto) como un tipo de ratón compatible. Si tiene un ratón PS/2, USB, o Bus, no necesita seleccionar un puerto y dispositivo. Si tiene un ratón serial, seleccione el puerto correcto y dispositivo en el que su ratón está conectado.

La casilla de verificación **Emular 3 Botones** permite simular el funcionamiento de un ratón de tres botones si tiene uno de dos. Si selecciona esta casilla de verificación, podrá emular un tercer botón, el que se encontraría en el "medio", presionando simultáneamente ambos botones disponibles en su ratón de dos botones.



**Figura 3.d Configuración del ratón**



Click en **Siguiente** para continuar.

### 3.12 Seleccionar Actualizar o Instalar

La pantalla **Examinar la actualización** aparece automáticamente si el programa de instalación detecta una versión previa de Red Hat Linux en el sistema.



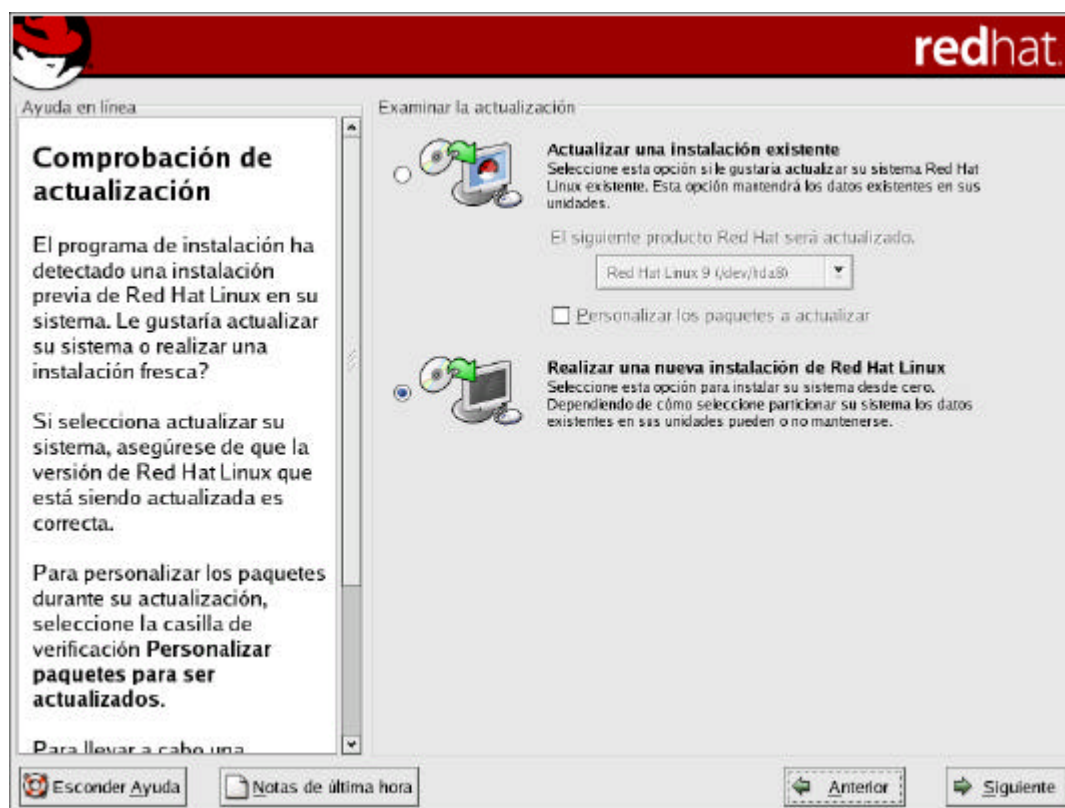
#### Nota

Si los contenidos del archivo `/etc/redhat-release` han sido cambiados de los valores predeterminados, la instalación Red Hat Linux puede no detectar cuando se intente actualizar a Red Hat Linux 9.

Pueden limitarse las verificaciones en este archivo mediante el arranque con el comando siguiente:

```
boot: Linux upgradeany
```

Click en **Siguiente** para continuar.



**Figura 3.e Seleccionar Actualizar o Instalar**

### 3.12 Opciones de instalación

Elija qué tipo de instalación desea realizar (**Figura 3.f**). El sistema Red Hat Linux le permitirá elegir el tipo de instalación que mejor se ajuste a sus necesidades. Las opciones disponibles son: **Estación de trabajo**, **Servidor**, **Portátil**, **Personalizada** y **Actualización**



**Figura 3.f Elegir el tipo de instalación**

### 3.13 Configuración del partición del disco

La partición permite dividir el disco duro en secciones aisladas, donde cada sección se comporta como un propio disco duro. La partición es especialmente útil si ejecuta más de un sistema operativo. En esta pantalla, se puede elegir entre realizar una partición automática o una partición manual con **Disk Druid**.

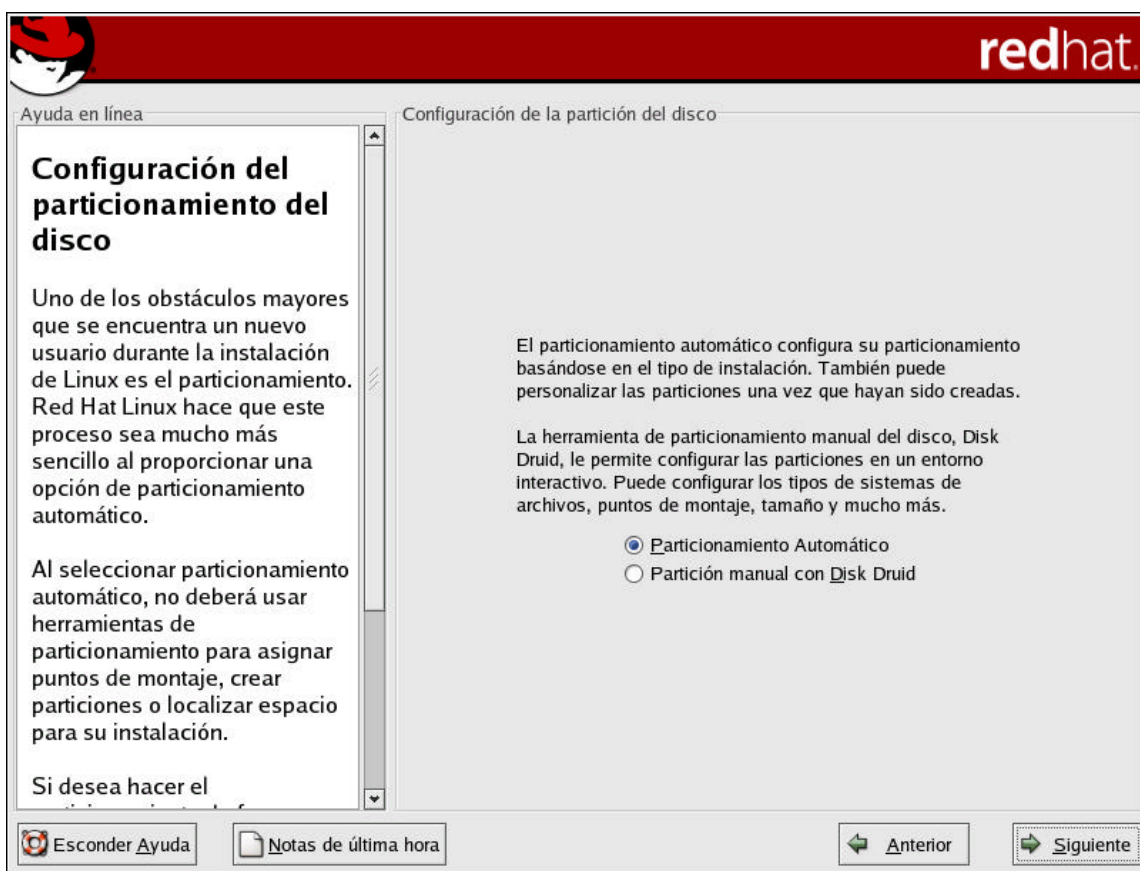
La partición automática permite realizar una instalación sin tener que particionar los discos uno mismo. De no sentirse seguro durante la partición manual del disco, se aconseja elegir la partición automática, a fin de que el proceso de instalación decida

por default qué tipo de partición adoptar. Para particionar de forma manual, se escoge la herramienta de partición **Disk Druid**.



#### Aviso

El **Agente de actualización de Red Hat** descarga paquetes actualizados a `/var/spool/up2date` por defecto. Si se particiona el sistema manualmente, y crea una partición `/var`, hay que verificar que la partición sea lo suficientemente grande para poder descargar todas las actualizaciones del paquete.



**Figura 3.g Configuración del partición del disco**

### 3.14 Partición automática

La partición automática permite tener control de los datos que se han eliminado en el sistema. Existen las siguientes opciones:

- **Eliminar todas las particiones Linux del sistema** — Se selecciona esta opción para eliminar tan sólo las particiones Linux (particiones creadas en una

instalación Linux previa). No borrará el resto de particiones que tenga en el disco(s) duro(s) (tal como VFAT o particiones FAT32).

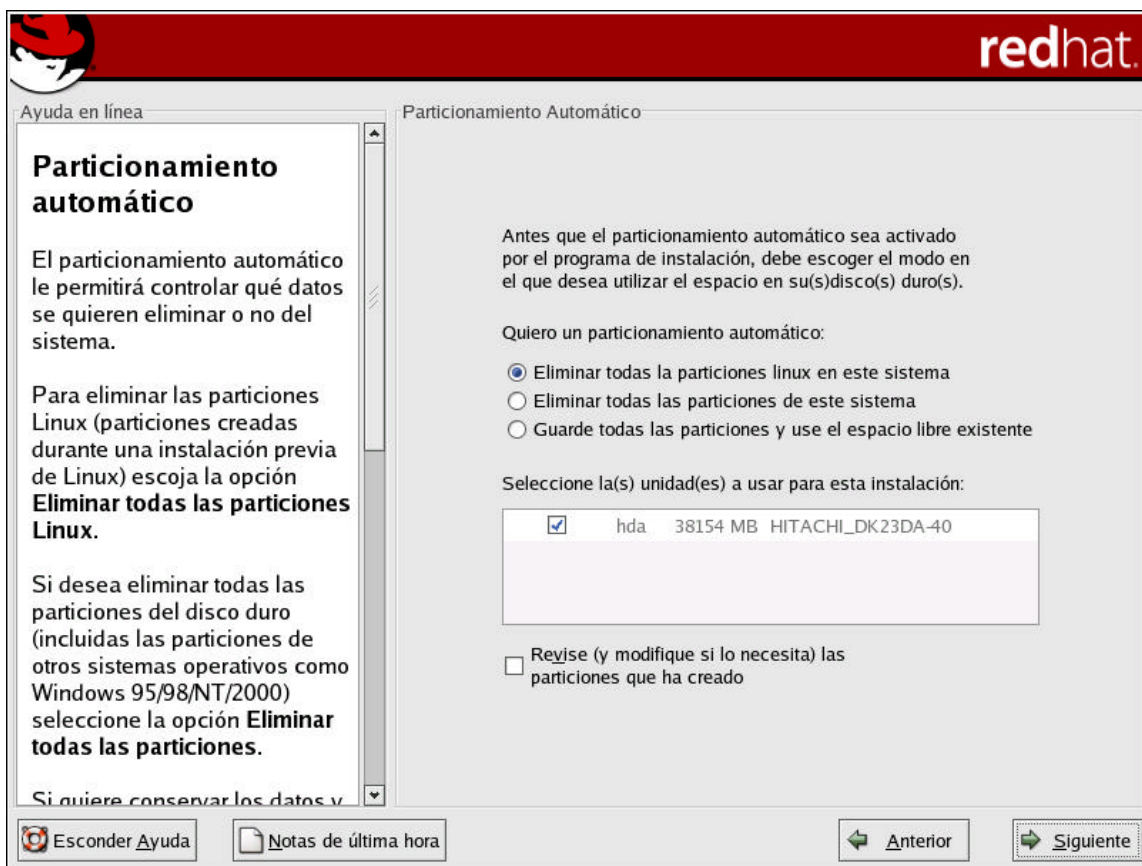
- **Eliminar todas las particiones del sistema** — Se selecciona esta opción para eliminar todas las particiones del disco duro (esto incluye las particiones creadas por otros sistemas operativos tales como Windows 95/98/NT/2000).



#### Atención

Si se selecciona esta última opción, todos los datos en el disco seleccionado serán eliminados por el programa de instalación. No se selecciona esta opción si hay información que se desee mantener en los discos duros en los que se está instalando Red Hat Linux.

- **Mantener todas las particiones y usar el espacio libre existente** — Se selecciona esta opción para conservar los datos y las particiones actuales, presumiendo que tiene suficiente espacio disponible en los discos duros.



**Figura 3.h Partición automático**

Mediante el uso del ratón, se escogen los discos duros en los que se desea instalar Red Hat Linux. Si hay presentes dos o más discos duros, es posible escoger qué disco duro debería contener la instalación. Esto no repercutirá a los discos duros no seleccionados ni a ninguno de los datos.



#### **Nota**

Siempre es una buena idea respaldar los datos que tenga en los diversos sistemas. Por ejemplo, si se está actualizando o creando un sistema de arranque dual, debería respaldarse los datos que se desea conservar en el(los) disco(s) duro(s). Ocurren errores y pueden resultar en la pérdida de todos los datos.

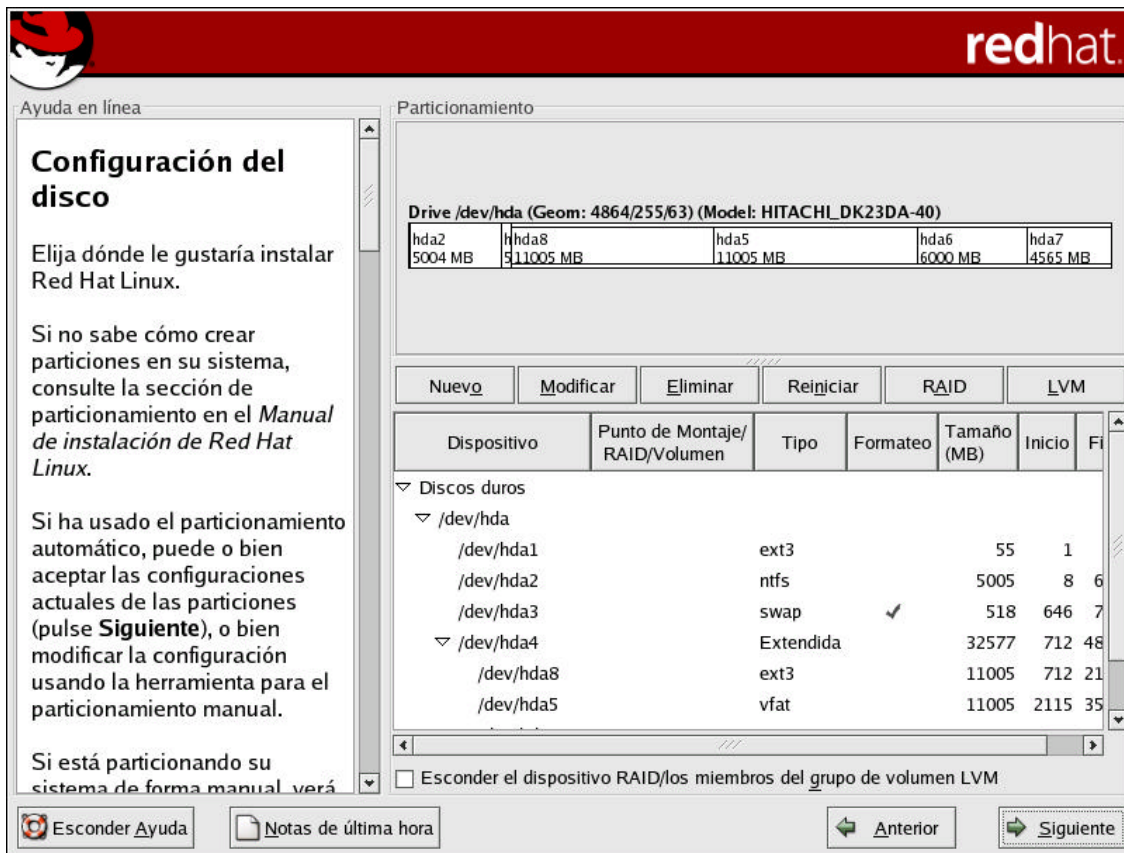
Para revisar y realizar los cambios necesarios en las particiones creadas con el partición automático, se selecciona la opción **Revisar**. Después de seleccionar **Revisar** y hacer click en **Siguiente** para continuar, se observan las particiones creadas en la aplicación **Disk Druid**. También se presenta la opción de modificar estas particiones si no cumplen las necesidades estipuladas.

Click en **Siguiente** para continuar.

### **3.15 Partición del sistema**

Si ha escogido partición automático y seleccionado **Revisar**, puede aceptar la configuración de las particiones actuales (Click en **Siguiente**) o modificar la configuración mediante el uso de Disk Druid, la herramienta de partición manual. Llegados a este punto, deberá indicar al programa de instalación donde instalar Red Hat Linux. Esto se hace mediante la definición de los puntos de montaje para una o más particiones de disco en las que se instalará Red Hat Linux. Necesitará también crear y/o eliminar particiones (Figura 3.i).





**Figura 3.i Partición con Disk Druid**

La herramienta de partición usada en Red Hat Linux 9 será el **Disk Druid**. Con la excepción de ciertas situaciones "esotéricas", el **Disk Druid** normalmente mantiene los requisitos de partición de una instalación normal de Red Hat Linux.

### 3.15.1 Visualización gráfica del/los disco/s duro/s

**Disk Druid** ofrece una representación gráfica de su/s disco/s duro/s. Click con el ratón para evidenciar un campo determinado en la visualización gráfica. Doble click para modificar la partición existente o para crear una partición fuera del espacio libre existente.

Sobre la visualización se podrá ver el nombre de la unidad (tal como /dev/hda), la **geometría** (la cual consiste de tres números representando el número de cilindros, cabezales y sectores del disco duro), y el **modelo** del disco duro como lo detectó el programa de instalación.

### 3.15.2. Botones de Disk Druid

Estos botones controlan las acciones de **Disk Druid**. Se utilizan para cambiar los atributos de una partición (por ejemplo, el tipo de sistema de archivos y el punto de montaje) y también para crear dispositivos RAID. Los botones de esta pantalla se utilizan también para aceptar los cambios que hemos realizado, o para salir de **Disk Druid**.

- **Nuevo:** Se usa para pedir una nueva partición. Cuando se selecciona, aparece un cuadro de diálogo que contiene los campos (como por ejemplo punto de montaje y tamaño) que deben ser rellenados.
- **Modificar:** Se usa para modificar los atributos de la partición que tenemos seleccionada en la sección **Particiones**. Si se acciona el botón **Modificar**, se abrirá un cuadro de diálogo. Se puede modificar cualquiera de los campos, en función de si la información ya ha sido escrita o no en el disco.

También se puede modificar el espacio libre, tal y como se muestra en la representación gráfica, para crear una nueva partición dentro de aquel espacio. Se resalta el espacio libre y selecciona el botón **Modificar** o se hace doble click en el espacio libre para modificarlo.

- **Borrar:** Se utiliza para borrar la partición que tenemos seleccionada en el momento de pulsar el botón en la sección **Particiones presentes en el disco**. Se pedirá confirmación en el caso de que se desee borrar cualquier partición.
- **Reiniciar:** Se utiliza para restaurar **Disk Druid** a su estado original. Todos los cambios efectuados se perderán si selecciona **Reiniciar** las particiones.
- **Creación del dispositivo RAID :** La **Creación del dispositivo RAID** puede ser utilizada si se quiere proporcionar redundancia a cualquier partición del disco. *Debe utilizarse tan sólo si se tiene experiencia con RAID.*

Para crear un dispositivo RAID, en primer lugar se deberá crear particiones de software RAID. Una vez que creadas dos o más particiones RAID de software, se selecciona **Crear RAID** para añadir las particiones RAID de software en un dispositivo RAID.

- **LVM:** Permite crear un volumen LVM lógico. La función del LVM (Logical Volume Manager) es presentar una visualización simple lógica del espacio físico almacenado como por ejemplo el/los disco(s) duro(s). El LVM administra los discos físicos individuales — es decir, las particiones individuales de los discos. *Se tendría que usar solamente si ya se tienen conocimientos sobre el LVM.* Para crear un volumen lógico LVM, primero se tiene que crear particiones físicas LVM. Una vez creadas, se selecciona **LVM** para crear el volumen lógico LVM.

### 3.15.3. Campos de la partición

La diferentes etiquetas de cada partición presentan información sobre las particiones que está creando. Las etiquetas son las que siguen a continuación:

- **Dispositivo:** Este campo muestra el nombre del dispositivo de la partición.
- **Punto de montaje:** Un punto de montaje es el lugar en la jerarquía de directorios a partir del cual un volumen existe; el volumen se "monta" en este lugar. Este campo indica dónde se montará la partición. Si la partición existe pero no se ha definido un punto de montaje, necesitará definir uno. Doble click sobre la partición o en el botón **Modificar** para cambiar los parámetros de la partición.
- **Tipo:** Este campo muestra el tipo de partición (por ejemplo, ext2, ext3, o vfat).
- **Formato:** Este campo muestra si la partición que se está creando se formateará.
- **Tamaño :** Este campo muestra el tamaño de la partición (en MB).
- **Comienzo:** Este campo muestra el cilindro en su disco duro donde la partición comienza.
- **Final:** Este campo muestra el cilindro en su disco duro donde la partición termina.
- **Ocultar los miembros del grupo del dispositivo RAID/volumen LVM:** Se selecciona esta opción si no se desea visualizar los miembros del grupo del dispositivo RAID o del volumen LVM que se han creado.

### 3.15.4. Esquema de partición recomendado

A no ser que tenga razones para hacerlo de otro modo, le recomendamos que cree las siguientes particiones:

- Una partición swap (de al menos 32 MB) — Las particiones swap se usan para habilitar la memoria virtual. En otras palabras, los datos se escriben en la partición swap cuando no hay suficiente RAM para almacenar los datos que el sistema está procesando. El tamaño mínimo de la partición swap debería ser igual al doble de la cantidad de memoria RAM que tiene el sistema o 32 MB.
- Una partición /boot (100MB) — la partición montada sobre /boot contiene el kernel del sistema operativo (que permitirá al sistema arrancar Red Hat Linux), junto a otros archivos utilizados para el proceso de arranque. Debido a las limitaciones de la mayoría de las BIOS de los ordenadores, se aconseja crear una partición pequeña para guardar estos archivos. Para la mayoría de los usuarios, una partición de arranque de 75 MB es suficiente.





### Aviso

No se debe crear la partición /boot del tipo LVM. Los gestores de arranque que se incluyen con Red Hat Linux no pueden leer particiones LVM y no podrán arrancar el sistema Red Hat Linux.



### Atención

Cuando se particione el disco, debe recordarse que las BIOS de algunos sistemas viejos no pueden acceder más allá de los primeros 1024 cilindros del disco duro. Si este es el caso, conviene dejar suficiente espacio para la partición Linux /boot en los primeros 1024 cilindros del disco duro para arrancar Linux. Las otras particiones Linux pueden estar ubicadas después del cilindro 1024. Si el disco duro tiene más de 1024 cilindros, puede necesitarse una partición /boot si se desea que la partición / (root) utilice todo el resto del espacio en su disco duro.

- Una partición root (1.7-5.0GB) — aquí es donde se ubica "/" (el directorio raíz). En esta instalación, todos los archivos (excepto los almacenados en /boot) se encuentran en la partición raíz. Una partición raíz de 1,7 GB permitirá una instalación del tipo escritorio personal (con *muy poco* espacio libre), mientras que una partición de 5.0GB le permitirá instalar todos los paquetes.

### 3.15.5. Añadir particiones

Para añadir una nueva partición, seleccione el botón **Nuevo**. Aparecerá una ventana de diálogo (Figura 3.j).

**Figura 3.j Creación de una nueva partición**

- **Punto de montaje:** Introducir el punto de montaje de la partición. Por ejemplo, si la partición va a ser la partición raíz, / o **/boot** para la partición **/boot**, etc. También se puede utilizar los menús desplegables para elegir el punto de montaje correcto para su partición.
- **Tipo de sistema de archivos:** Usando el menú desplegable, se selecciona el tipo de sistema de archivo apropiado para esta partición.
- **Unidades admisibles:** Este campo contiene una lista de los discos instalados en el sistema. Si la caja de un disco duro está seleccionada, se puede crear una partición en ese disco. Si la caja *no* está seleccionada entonces la partición *nunca* se creará en ese disco. Usando diferentes combinaciones de estas casillas de verificación se puede hacer que **Disk Druid** coloque las particiones tal y como se ven, o bien se puede dejar que **Disk Druid** decida dónde deberán ir las particiones.
- **Tamaño (Megas):** Introducir el tamaño de la partición (en megabytes). Téngase en cuenta que este campo comienza con 100MB y, a no ser que se cambie este valor, sólo creará una partición de 100 MB.
- **Opciones de tamaño adicionales:** Escoger si se desea mantener esta partición con un tamaño fijo o permitir que "crezca" (que llene el espacio del disco duro disponible) hasta un cierto punto, o permitir que crezca para llenar cualquier espacio de disco duro restante disponible.

Si se escoge **Llenar todo el espacio hasta (MB)**, se debe dar el límite del tamaño en el campo de la derecha de esta opción. Esto permite que se mantenga una cierta cantidad de espacio libre en el disco duro para usos futuros.

- **Forzar a que sea una partición primaria:** Se selecciona si la partición que se está creando debería ser una de las primeras cuatro particiones de su disco duro. Si no se selecciona, la partición que se cree será una partición lógica.
- **Buscar bloques dañados:** Buscar bloques dañados en una unidad puede ayudar en la prevención de pérdida de datos al localizar los bloques dañados en una unidad y hacer una lista de éstos para prevenir el uso en el futuro. Si se desea buscar bloques dañados mientras formatea cada sistema de archivos, debe seleccionarse esta opción.

La selección de **Buscar bloques dañados** puede aumentar drásticamente el tiempo de instalación total. Al ser los discos cada vez más grandes, este proceso puede llegar a durar mucho tiempo; esta duración dependerá directamente del tamaño del disco. Si se elige esta opción, podrá monitorizarse el estado en la consola virtual #5.

### 3.15.6 Tipos de sistemas de archivos

Red Hat Linux permite crear diferentes tipos de particiones, basadas en el sistema de archivos que usarán. A continuación se presenta una breve descripción de los diferentes sistemas de archivos disponibles y de cómo utilizarlos.

- **ext2** — Un sistema de archivos ext2 soporta tipos de archivo estándar Unix (archivos regulares, directorios, enlaces simbólicos, etc). Proporciona la habilidad de asignar nombres de archivos largos, hasta 255 caracteres. Las versiones anteriores a Red Hat Linux 7.2 utilizaban el sistema de archivos ext2 por defecto.
- **ext3** — El sistema de archivos ext3 está basado en el sistema de archivos ext2 y tiene una ventaja principal — journaling. El uso de un sistema de archivos journaling reduce el tiempo de recuperación tras una caída, ya que no es necesario hacer **fsck** al sistema de archivos. El sistema de archivos ext3 está seleccionado por defecto y su uso es recomendado.
- **volumen físico (LVM)** — Mediante la creación de una o más particiones LVM se permite crear un volumen lógico LVM.
- **software RAID** — La creación de dos o más particiones de software RAID permite crear un dispositivo RAID.
- **swap** — Las particiones swap se usan para permitir el uso de memoria virtual. En otras palabras, los datos se escriben en una partición swap cuando no hay suficiente RAM para guardar los datos que el sistema está procesando.
- **vfat** — El sistema de archivos VFAT es un sistema de archivos Linux compatible con Windows 95/NT en el sistema de archivos FAT de nombres largos.

### 3.15.7 Modificación de particiones

Para modificar una partición, Click en el botón **Modificar** o doble click en la partición existente.



#### Nota

Si la partición ya existe en el disco duro, sólo tendrá que cambiarse el punto de montaje de la partición. Si se desea realizar otros cambios, tendrá que borrarse la partición y volverse a crear.

### 3.15.8 Borrar particiones

Para borrar una partición, se resalta en la sección **Particiones** y click en el botón **Borrar**.

### 3.16 Configuración del gestor de arranque

Para poder arrancar el sistema Red Hat Linux sin el disquete de arranque, se necesita instalar un gestor de arranque. El gestor de arranque es el primer software que se ejecuta cuando se arranca la computadora. Es responsable de la carga y de la transferencia del control al software del sistema operativo del kernel. El kernel, por otro lado, inicia el resto del sistema operativo. El programa de instalación ofrece dos gestores de arranque, GRUB y LILO.

GRUB (GRand Unified Bootloader), que se instala por defecto, es un gestor de arranque muy potente ya que puede cargar una gran variedad de sistemas operativos gratuitos así como sistemas operativos de propietarios con el sistema de cargado en cadena (el mecanismo para cargar sistemas operativos no soportados mediante la carga de otro gestor de arranque, tal como DOS o Windows).

LILO (Linux LOader) es un gestor de arranque versátil para Linux. No depende de un determinado sistema de archivos y puede arrancar las imágenes del kernel de Linux de los disquetes y de los discos duros e incluso arrancar otros sistemas operativos.

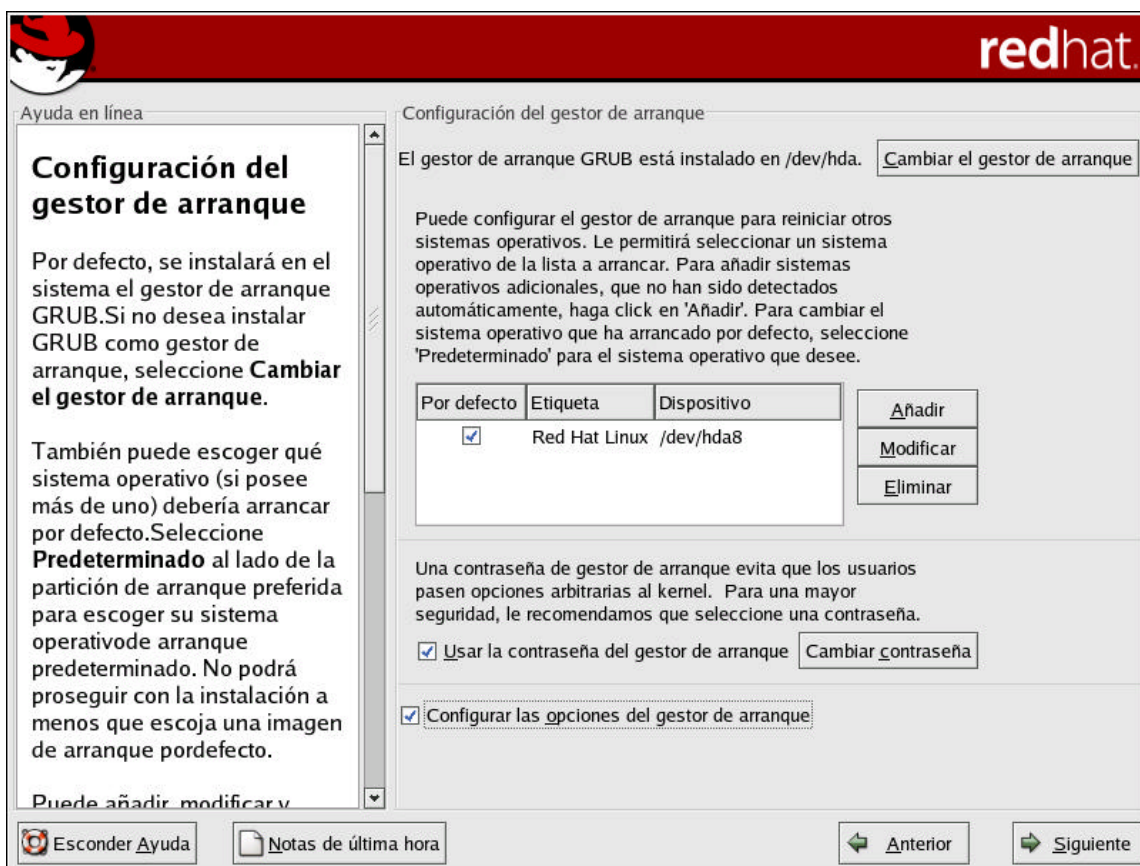


Figura 3.k Configuración del gestor de arranque

Si no se desea instalar GRUB como el gestor de arranque, Click en **Cambiar el gestor de arranque**. Se puede escoger entre instalar LILO o instalar ninguno. Si ya existe otro gestor de arranque que arranque Linux y no se desea sobreescribirlo el gestor de arranque actual o si se desea arrancar el sistema con disquetes, debe elegirse **No instalar gestor de arranque** haciendo click en el botón **Cambiar el gestor de arranque**.



#### **Atención**

Si se decide no instalar LILO o GRUB no podrá arrancarse el sistema directamente y se necesitará usar otro método de arranque (como el disquete de arranque). ¡Solamente úsese esta opción si conoce otra manera de arrancar el sistema!

Todas las particiones que se pueden arrancar aparecen en una lista, incluso las particiones que usan otros sistemas operativos. La partición que contiene el sistema de archivos root del sistema tiene la **Etiqueta** de Red Hat Linux (para GRUB) o Linux (para LILO). Las otras particiones pueden que también tengan etiquetas de arranque. Si se desea añadir o cambiar la etiqueta de arranque de las otras particiones que el programa de instalación ya ha detectado, Click en una de ellas para seleccionarla. Una vez seleccionada puede realizarse los cambios de la etiqueta de arranque haciendo click en el botón **Modificar**. Se selecciona **Por defecto** junto con la partición root preferida para escoger el sistema operativo que se desee arrancar por defecto. No podrá avanzarse en la instalación mientras no se escoja la imagen de arranque por defecto.

Las contraseñas del gestor de arranque ofrecen un mecanismo de seguridad en un ambiente en el que se tenga acceso físico al servidor. Si está instalado un gestor de arranque, debe crearse una contraseña para proteger el sistema. Sin dicha contraseña, los usuarios con acceso a el sistema pueden pasar opciones al kernel que pueden comprometer la seguridad de el sistema. Con ella, se tiene que introducir la contraseña para poder seleccionar cualquier opción de arranque que no sea estándar.

Si se selecciona colocar una contraseña para aumentar la seguridad del sistema, asegúrese de seleccionar la casilla **Usar la contraseña del gestor de arranque**. Una vez seleccionada, se introduce la contraseña y confirma.

### **3.16.1 Configuración avanzada del gestor de arranque**

Ahora que se ha decidido cuál gestor de arranque instalar, tiene que decidirse dónde se quiere instalarlo. Se puede instalar el gestor de arranque en uno de los dos sitios siguientes:

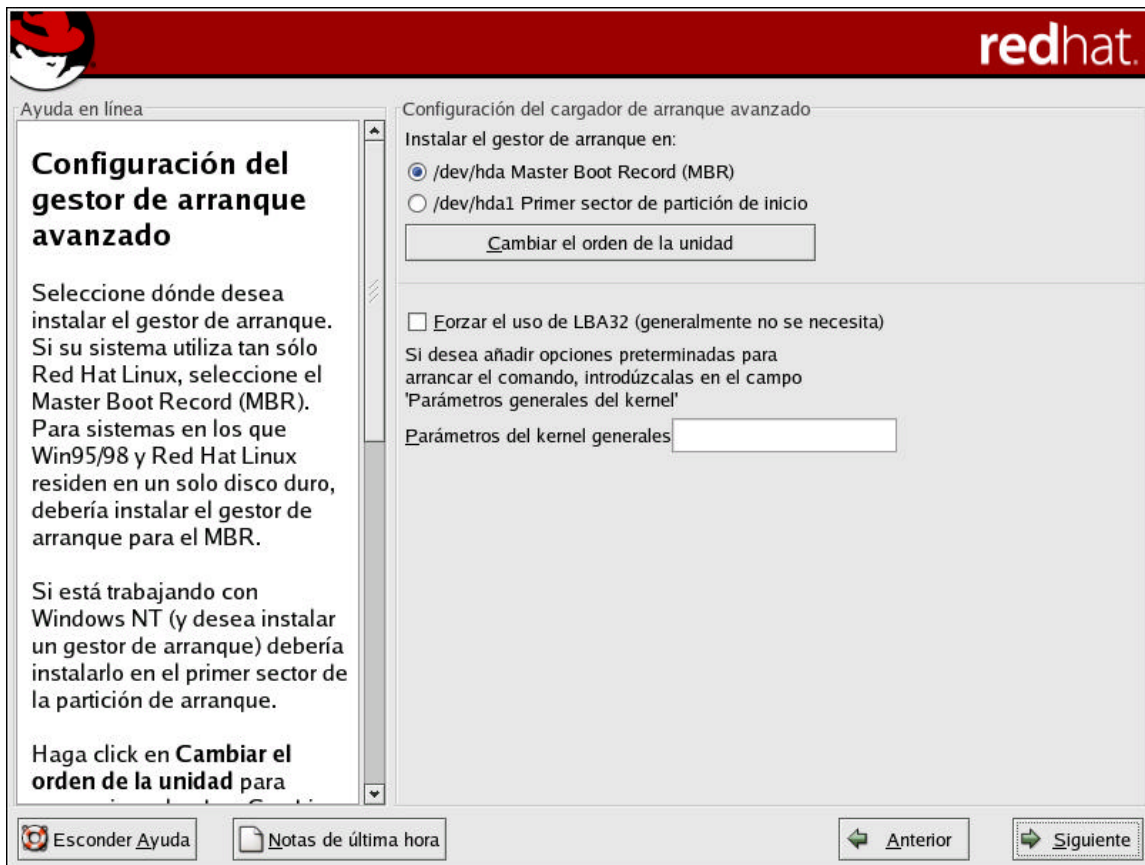
El master boot record (MBR)

Este es el sitio recomendado para instalar un gestor de arranque, a no ser que el MBR esté ocupado por el gestor de arranque de otro sistema operativo, como

System Commander. El MBR es un área especial del disco duro que la BIOS de la computadora carga automáticamente y el punto más próximo en el que el gestor de arranque puede tomar el control de la secuencia de arranque. Si se instala en el MBR, al arrancar la máquina, GRUB (o LILO) presentará un indicador de comandos de arranque. Podrá entonces iniciarse Red Hat Linux o cualquier otro sistema operativo que haya indicado al gestor de arranque.

El primer sector de la partición raíz

Se recomienda si se está utilizando otro gestor de arranque en el sistema. En este caso, el otro gestor de arranque tendrá el control en un primer momento. Podrá configurarse ese gestor de arranque para que inicie GRUB (o LILO), que iniciará a su vez Red Hat Linux.



**Figura 3.1 Instalación del gestor de arranque**

Si el sistema sólo utilizará Red Hat Linux, debería seleccionarse el MBR. Para sistemas con Windows 95/98, también debería instalarse el gestor de arranque en el MBR para que se puedan iniciar los dos sistemas operativos.

Click en **Cambiar el orden de la unidad** si se desea cambiar el orden de la unidad o si el BIOS no devuelve el orden correcto. Será útil si se tiene más de un adaptador SCSI o si se tiene ambos tipos de adaptadores SCSI e IDE y se desea arrancarlos con el dispositivo SCSI. La opción **Forzar el uso de LBA32 (no requerida normalmente)** permite exceder el límite de cilindro 1024 para la partición /boot. Si existe un sistema que es compatible con la extensión LBA32 para arrancar los sistemas operativos por encima del límite de cilindro 1024 y se desea ubicar la partición /boot más allá de este límite, deberá seleccionarse esta opción. Si se desea añadir opciones predeterminadas al comando boot, se ingresan en el campo **Parámetros generales del Kernel**. Cualquier opción ingresada se pasarán al kernel del Linux cada vez que arranque.

### 3.16.2 Modo rescue

Si llega a necesitarse el modo rescue, hay varias opciones:

- Usando el CD-ROM de arranque, escribiendo **Linux rescue** en el indicador de comandos boot.
- Arrancando el sistema desde un disquete de arranque de instalación creado desde la imagen boot.img. Este método requiere que se inserte el CD-ROM #1 de Red Hat Linux como la imagen de rescate o que la imagen de rescate esté en el disco duro como una imagen ISO. Una vez arrancado el sistema usando este disquete, se escribe **Linux rescue** en el intérprete de comandos boot.
- Usando el disquete de arranque en red creado a partir de bootnet.img o del disquete de arranque PCMCIA de pcmcia.img. Una vez arrancada utilizando este disquete, se escribe **Linux rescue** en el indicador de comandos boot:. Sólo puede hacerse esto si la conexión en red está funcionando. Deberá identificarse el host de la red y el tipo de transferencia.

### 3.16.3 Gestores de arranque alternativos

Como se ha explicado con anterioridad, si no se desea usar un gestor de arranque, existen varias alternativas:

Disquete de arranque

Puede utilizarse el disquete de arranque creado desde el programa de instalación (si ha elegido crearlo).

LOADLIN

Puede cargarse Linux desde MS-DOS. Desafortunadamente, esto requiere una copia del kernel de Linux (y un disco RAM inicial si dispone de un dispositivo SCSI) disponible en la partición MS-DOS. La única forma de realizar esto es arrancando el sistema Red Hat Linux con otro método (por ejemplo, desde un



disquete de arranque) y después copiar el kernel a una partición MS-DOS. LOADLIN está disponible en

<ftp://metalab.unc.edu/pub/Linux/system/boot/dualboot/>

y en los sitios réplicas asociados.

## SYSLINUX

SYSLINUX es un programa de MS-DOS muy similar a LOADLIN. También está disponible desde

<ftp://metalab.unc.edu/pub/Linux/system/boot/loaders/>

y en los sitios espejo asociados.

## Gestores de arranque comerciales

Puede cargarse Linux usando gestores de arranque comerciales. Por ejemplo, System Commander o Partition Magic pueden arrancar Linux (pero todavía necesitan que GRUB o LILO estén instalados en la partición raíz de Linux).



### Nota

Los gestores de arranque tales como LOADLIN y System Commander son considerados gestores de terceros y no son soportados por Red Hat.

### 3.16.4 Tarjetas madre SMP, GRUB, y LILO

Esta sección es específica de las tarjetas madre SMP. SMP, acrónimo de Symmetric Multiprocessing (Multiprocesamiento simétrico), es una arquitectura de computadores que proporciona un rápido rendimiento al poner varias CPUs disponibles para realizar simultáneamente procesos individuales (multiprocesamiento).

Si el programa de instalación detecta una tarjeta madre SMP en el sistema, automáticamente creará dos entradas para el gestor de arranque.



### Nota

Los sistemas Intel® Pentium® 4 con hipertrámado tendrán un kernel SMP instalado por defecto.

Las dos entradas de GRUB serán Red Hat Linux (*kernel version*) y Red Hat Linux (*kernel version-smp*). La Red Hat Linux (*kernel version-smp*) arrancará por defecto.



Sin embargo, si tiene problemas con el kernel SMP, se puede elegir arrancar la entrada Red Hat Linux (*kernel version*). Se mantendrá toda la funcionalidad, pero tan sólo estará operándose con un procesador.

Las dos entradas LILO serán Linux y Linux-up. La entrada Linux arrancará por defecto. Sin embargo, si se tiene problemas con el kernel SMP, puede elegirse si se quiere arrancar Linux-up en su lugar. Se mantendrá toda la funcionalidad que se tenía antes, pero tan sólo se utilizará un procesador.

### 3.17 Configuración de red

Si se tiene un dispositivo de red y no se ha configurado todavía la red (como por ejemplo proporcionar un disco de arranque de red que se haya creado y entrar en la información de red como se indica), se tiene la oportunidad de hacerlo (Figura 3.m).

**Configuración de la red**

Cualquier dispositivo de red que posea en el sistema será detectado automáticamente por el programa de instalación y aparecerá en la lista de **Dispositivos de red**.

Para configurar el dispositivo de red, seleccione el dispositivo y haga click en **Modificar**. En la pantalla de **Modificar la interfaz**, podrá escoger tener información de IP y máscara de red configurada por DHCP o puede introducirlo de forma manual. Puede escoger también que el dispositivo sea activo en el momento del arranque.

Si no tiene acceso a cliente

Activar al inicio	Dispositivo	IP/Máscara de red
<input checked="" type="checkbox"/>	eth0	DHCP

**Nombre del Host**

Configurar el nombre del host:

☒ de forma automática a través de DHCP

☐ de forma manual

**Configuración miscelánea**

Puerta de enlace: [ ][ ][ ][ ]

DNS Primario: 172 . 16 . 5 . 3

DNS Secundario: [ ][ ][ ][ ]

DNS Terciario: [ ][ ][ ][ ]

Esconder Ayuda    Notas de última hora    Anterior    Siguiente

**Figura 3.m Configuración de red**

El programa de instalación automáticamente detecta los dispositivos de red que tiene y los muestra en la lista **Dispositivos de red**. Una vez que se ha seleccionado el dispositivo de red, Click en **Modificar**. En la pantalla desplegable **Modificar interfaz**

puede elegirse la dirección IP o la máscara de red del dispositivo con el DHCP (o manualmente si no se ha seleccionado DHCP ) y puede también activarse el dispositivo en el intervalo de arranque. Si se selecciona **Activar en arranque**, el dispositivo de red arrancará cuando arranque el sistema. Si no se tiene el acceso al cliente DHCP o no se está seguro debe contactarse al administrador de red.

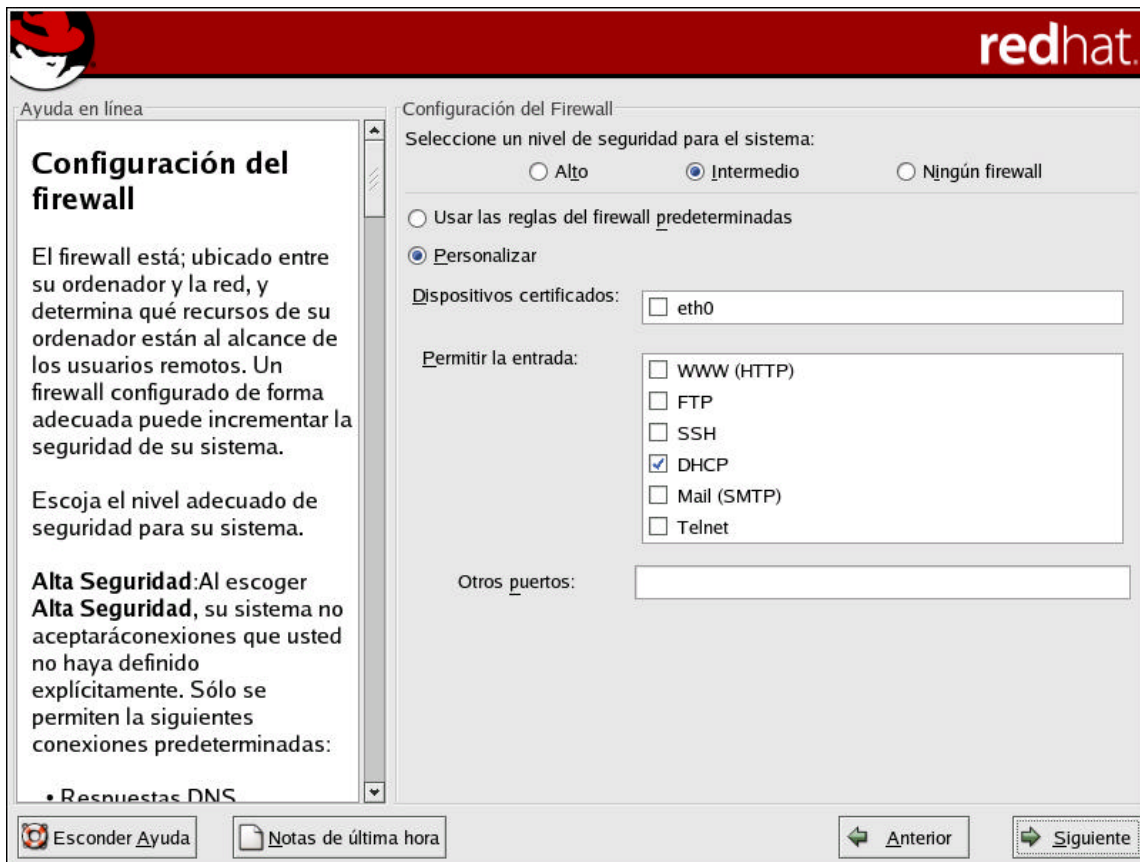
The image shows a window titled "Modificar la interfaz eth0". Inside, under the heading "Configure eth0", there are two checkboxes: "Configurar usando DHCP" (unchecked) and "Activar al inicio" (checked). Below the checkboxes are two rows of input fields. The first row is labeled "Dirección IP:" and contains four boxes with the values 192, 0, 2, and 0. The second row is labeled "Máscara de red:" and contains four boxes with the values 255, 255, 255, and 0. At the bottom of the window are two buttons: "Cancelar" (with a red X icon) and "OK" (with a green checkmark icon).

**Figura 3.n Modificación del dispositivo de red**

Si se tiene un nombre de dominio para el dispositivo de red, se introduce en el campo **Nombre del host** o puede elegirse tener el DHCP automáticamente. Para finalizar, se introduce la **Puerta de enlace** y el **DNS primario** (si se conocen, también el **DNS secundario** y **DNS terciario**).

### 3.18 Configuración del cortafuegos

Red Hat Linux también ofrece protección vía cortafuegos (firewall) para una seguridad mejorada del sistema. Un cortafuegos se dispone entre la computadora y la red y determina qué recursos del equipo están accesibles para los usuarios remotos de la red. Un cortafuegos bien configurado puede aumentar significativamente la seguridad del sistema.



**Figura 3.o Configuración del cortafuegos**

Selección del nivel de seguridad apropiado del sistema:

### **Alto**

Si se elige **Alto**, el sistema no aceptará más que las conexiones (además de las definidas por defecto) que hayan sido explícitamente definidas. Por defecto, sólo se aceptan las siguientes conexiones:

- Respuestas de DNS
- DHCP — para que cualquier interfaz de red que use DHCP puedan ser configuradas correctamente

Si se elige **Alto**, el cortafuegos no permitirá:

- FTP en modo activo (el modo pasivo de FTP, usado por defecto en la mayoría de los clientes deberá funcionar)
- Transferencias de archivos IRC DCC
- RealAudio™

- Clientes del Sistema X Window remotos

Si se conecta el sistema a Internet, pero no piensa utilizarse como servidor, ésta es la opción más segura. Si se necesita servicios adicionales, puede elegirse **Personalizar** para permitir servicios específicos a través del cortafuegos.



#### Nota

Si se selecciona configurar un cortafuegos con un nivel de seguridad medio o alto durante la instalación, los métodos de autenticación de red (NIS y LDAP) no funcionarán.

### Medio

Si se elige **Medio**, su cortafuegos no se permitirá a las máquinas remotas acceder a ciertos recursos de el sistema. Por defecto, los accesos a los recursos siguientes no están permitidos:

- Puertos menores del 1023 — los puertos reservados estándar, usados por la mayoría de los servicios, como **FTP**, **SSH**, **telnet**, **HTTP**, y **NIS**.
- Puerto del servidor NFS (2049) — NFS está desactivado para los servidores remotos y los clientes locales.
- La pantalla local del Sistema X Window para clientes X remotos.
- Puerto del servidor de fuentes de X (por defecto, **xfs** no escucha a la red, ya que está deshabilitado en el servidor de fuentes).

Si se quiere, puede permitirse recursos, como por ejemplo, **RealAudio™**, mientras que se bloqueen los recursos normales (elegir **Medio**). Puede seleccionarse **Personalizar** para permitir a servicios específicos pasar a través del cortafuegos.

### Ningún cortafuegos

La configuración "ningún cortafuegos" proporciona un acceso completo al sistema y no realiza ningún tipo de verificación de seguridad. La comprobación de seguridad es la desactivación del acceso a determinados servicios. Tan sólo se recomienda esta opción si se está usando una red certificada y segura (no Internet), o si se planea realizar una configuración detallada del cortafuegos más adelante.

Se elige **Personalizar** para añadir dispositivos autenticados o para permitir nuevos servicios.

## Dispositivos autenticados

Al seleccionar **Dispositivos autenticados** se permitirá el acceso al sistema a todo el tráfico del dispositivo; se excluirá de las reglas del cortafuegos (firewall). Por ejemplo, si se está ejecutando una red local, pero se está conectado a Internet vía PPP, puede seleccionarse **eth0** y todo el tráfico que llegue de la red local será permitido. Al seleccionar **eth0** como autenticado significa que se aceptará todo el tráfico Ethernet, pero la interfaz ppp0 seguirá bajo el cortafuegos. Si se quiere restringir todo el tráfico de una interfaz, se deja sin seleccionar.

## Permitir la entrada

Al activar esta opción, permitirá que algunos servicios específicos puedan pasar a través del cortafuegos. Téngase en cuenta que, durante una instalación del tipo estación de trabajo, la mayoría de estos servicios *no* están instalados en el sistema.

### DHCP

Si se permite la entrada y salida de peticiones DHCP, se permitirá el acceso a cualquier interfaz de red que esté utilizando DHCP para determinar su propia dirección IP. DHCP suele estar activado. Si DHCP no está activado, la computadora no podrá obtener una dirección IP.

### SSH

Secure *S*hell (SSH) es un paquete de utilidades para poder conectarse y ejecutar programas en una máquina remota. Si se prevé utilizar las herramientas SSH para que pasen a través de un cortafuegos, deberá activarse esta opción. Se necesitará tener el paquete para servidor de SSH instalado para acceder a la máquina remotamente utilizando herramientas SSH.

### Telnet

Telnet es un protocolo para conectarse remotamente a máquinas fuera del alcance físico. Las comunicaciones Telnet no son encriptadas y no proporcionan seguridad de red. No se recomienda permitir Telnet remoto. Si quiere permitir el servicio Telnet entrante, deberá instalarse el paquete de servidor de Telnet.

### WWW (HTTP)

El protocolo HTTP es utilizado por Apache (y otros servidores Web) para ofrecer el servicio de páginas Web. Si se quiere crear un servidor Web disponible públicamente, se habilita esta opción. Esta opción no es necesaria para la publicación de páginas Web en local. Necesitará instalarse el paquete httpd si

se quiere servir páginas Web. La habilitación de **WWW (HTTP)** no abrirá un puerto para HTTPS. Para habilitar HTTPS, es necesario especificarlo en el campo **Otros puertos**.

### **Mail (SMTP)**

Si se desea permitir la entrega de correo a través de su cortafuegos, de tal forma que los hosts remotos se puedan conectar directamente a su máquina para repartir el correo, se habilita esta opción. No es necesaria esta opción para obtener su correo desde el servidor ISP usando POP3 o IMAP, o si se utiliza una herramienta como fetchmail. Obsérvese que un servidor configurado inadecuadamente puede permitir que máquinas remotas usen el servidor para enviar correo basura.

### **FTP**

El protocolo FTP se utiliza para transferir archivos entre máquinas en una red. Si se considera dentro de la planeación colocar un servidor FTP disponible públicamente, se habilita esta opción. Necesitará instalarse el paquete de servidor de FTP para que esta opción sea útil.

## **3.19 Selección del soporte del idioma**

Puede instalarse y soportarse múltiples idiomas para usar en el sistema. Debe instalarse un idioma para usarlo como idioma por defecto. El idioma por defecto será usado por el sistema una vez que la instalación se haya completado. Si se escoge instalar otros idiomas, puede cambiarse entre los idiomas escogidos tras la instalación.

Si tan sólo va a utilizarse un idioma en el sistema, se puede ganar bastante espacio en disco. El idioma por defecto es el idioma que se haya seleccionado durante el proceso de instalación.



**Figura 3.p Selección del soporte del idioma**

Para usar más de un idioma en el sistema, deben escogerse idiomas específicos para instalar o seleccionarse todos los idiomas para tenerlos todos a disposición en el sistema Red Hat Linux. El botón **Reiniciar** se utiliza para cancelar las selecciones. Al reiniciar se volverá a la opción predeterminada y tan sólo el idioma que se ha seleccionado para el uso durante la instalación será instalado.

### 3.20 Configuración del huso horario

Puede elegirse el huso horario o bien seleccionarse la localización física de la computadora o bien especificarse el huso horario en función del Universal Time Coordinated (UTC). Las dos pestañas en lo alto de la pantalla (Figura 3.q) indican esas opciones. La primera pestaña permite configurar el huso horario en función de la localización.

En el mapa interactivo, puede hacerse click en una ciudad específica, que esté marcada con un punto amarillo; una **X** roja aparecerá indicando la selección. Se puede también desplazarse en la lista y seleccionar un huso horario.



La segunda pestaña permite especificar un desplazamiento UTC. Esta pestaña muestra una lista de desplazamientos para elegir, así como las opciones para el cambio del horario de invierno/verano.



**Figura 3.q Configuración del huso horario**

En ambas pestañas, puede seleccionarse **El reloj del sistema usa UTC**. Se elige esta opción cuando se sabe que el equipo está en hora UTC.

### 3.21 Configuración de la contraseña de root

La configuración de la cuenta y la contraseña root es uno de los pasos más importantes durante la instalación. La cuenta root es similar a la cuenta del administrador usada en las máquinas Windows NT. La cuenta root es usada para instalar paquetes, actualizar RPMs y realizar la mayoría de las tareas de mantenimiento del sistema. Conectándose como root proporciona control completo sobre el sistema.





### Nota

El usuario root (también conocido como supersuario) posee acceso completo al sistema; por este motivo, registrarse como usuario root es aconsejable hacerlo *sólo* para ejecutar el mantenimiento o administración del sistema.

The screenshot shows a Red Hat installation window titled "redhat.". On the left, there is a sidebar with a link "Ayuda en línea" and a section "Configurar contraseña de root". The text in this section advises using the root account only for administration and creating a regular user account for general use. The main area is titled "Coloque la contraseña de root" and contains a yellow shield icon, the instruction "Introduzca la contraseña de root (administrador) de este sistema.", and two input fields for "Contraseña de root:" and "Confirmar:", both masked with asterisks. Below the fields, it says "Contraseña de root aceptada.". At the bottom, there are buttons for "Esconder Ayuda", "Notas de última hora", "Anterior", and "Siguiente".

**Figura 3.r Contraseña root**

Se aconseja crear una cuenta que no sea root para uso general y ejecutar el comando "su -" para actuar como root cuando se necesite configurar algo de forma rápida. Estas normas básicas minimizarán las posibilidades de que un comando incorrecto o de un error de tipografía puedan dañar el sistema.

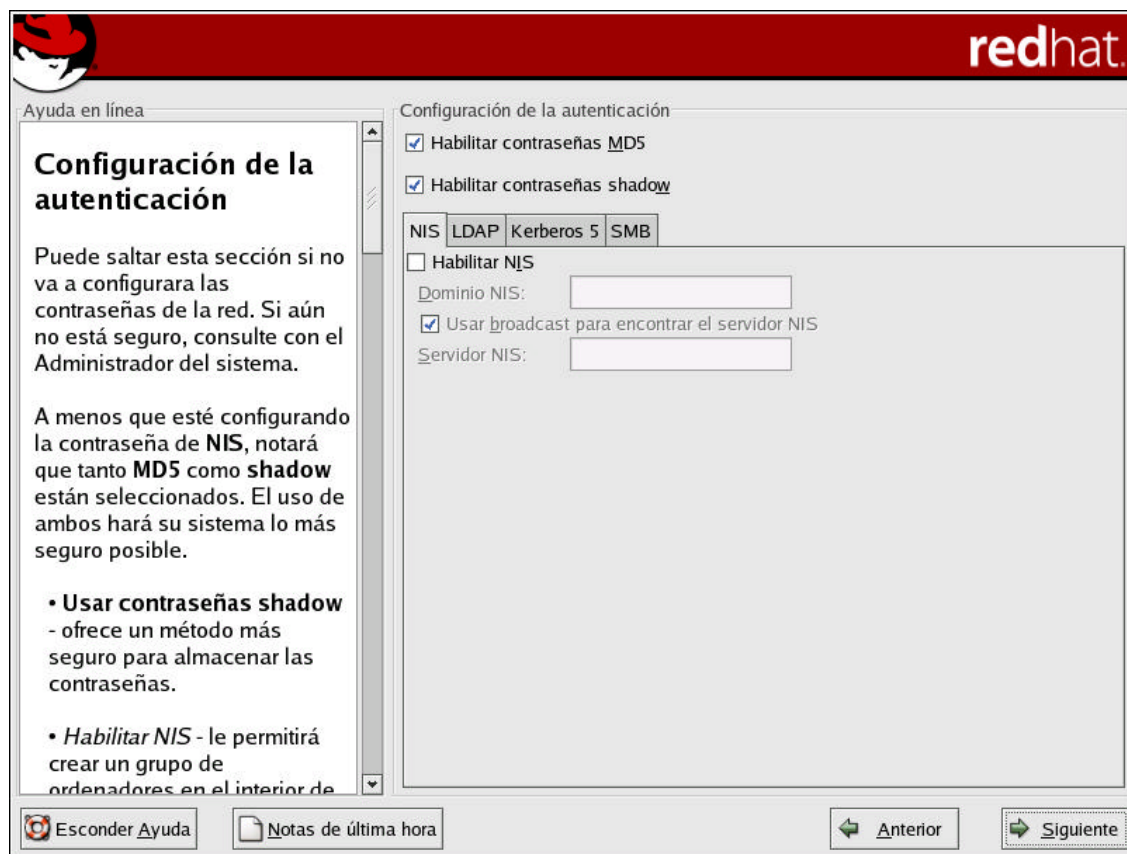
El programa de instalación dará indicaciones para que se configure una contraseña de root para el sistema. Debe introducirse una contraseña de root. El programa de instalación no permitirá seguir a la siguiente sección sin introducir una contraseña de root.

La contraseña de root debe de tener al menos seis caracteres y no aparecerá en la pantalla cuando se teclee. Deberá introducirse dos veces; si las dos contraseñas no

coinciden, el programa de instalación pedirá que se vuelvan a introducir. Lo correcto es escribir una contraseña de root fácil de recordar, pero que no sea obvia o fácil de adivinar. *qwerty*, *contraseña*, *root*, *123456* y *anteayer* serían ejemplos de malas contraseñas. Las contraseñas mejores son aquellas que mezclan números con letras mayúsculas y minúsculas que no formen palabras contenidas en diccionarios, como por ejemplo : *Aard387vark* o *420BMttNT*. La contraseña es sensible a las mayúsculas y minúsculas. Se recomienda nunca escribir la contraseña pero, si se escribe en un papel, es mejor guardarlo en un lugar seguro. Asimismo, el cambiar de contraseña periódicamente puede ser otra buena medida preventiva.

### 3.22 Configuración de la autenticación

A no ser que esté utilizando autenticación **NIS**, puede verse que sólo las contraseñas tipo **MD5** y **shadow** están seleccionadas (Figura 3.s). Se recomienda que se utilice ambos tipos de contraseñas para que la máquina sea lo más segura posible. Para configurar la opción NIS, se debe estar conectado a una red NIS.



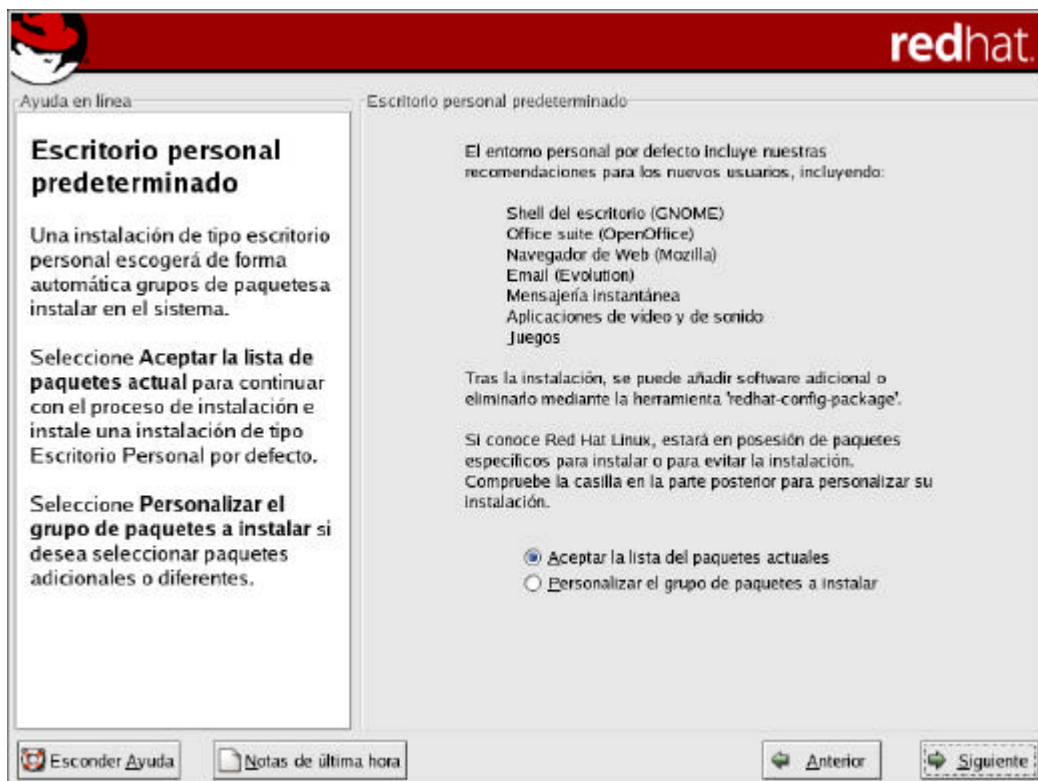
**Figura 3.s Configuración de la autenticación**

- **Habilitar contraseñas MD5** — permite usar una contraseña larga (de hasta 256 caracteres), en vez de las ocho letras o menos estándar.
- **Habilitar contraseñas shadow** — proporciona un método seguro para conservar contraseñas. Las contraseñas se almacenan en `/etc/shadow`, al que tan sólo se puede acceder como root.
- **Habilitar NIS** — permite ejecutar un grupo de ordenadores en el mismo dominio Network Information Service con una contraseña común y un archivo de grupo. Puede escogerse desde las siguientes opciones:
  - **Dominio NIS** — permitirá especificar el dominio o grupo de ordenadores al que pertenece el sistema.
  - **Usar broadcast para encontrar el servidor NIS** — permitirá hacer un broadcast en la red local para encontrar el servidor NIS.
  - **Servidor NIS** — hace que la computadora utilice un servidor NIS específico, en vez de enviar un broadcast a toda la red para encontrar el servidor NIS disponible.
- **Activar LDAP** — le dice a la computadora que utilice LDAP para algunas o todas las autenticaciones. LDAP consolida ciertos tipos de información en el seno de la organización. Por ejemplo, todas las diferentes listas de usuarios que puede haber en la organización se pueden unir en un sólo directorio LDAP. Podrá elegirse entre las siguientes opciones:
  - **Servidor LDAP** — permite acceder a un servidor específico (mediante su dirección IP) que esté ejecutando el protocolo LDAP.
  - **LDAP Base DN** — permite buscar información acerca de los usuarios mediante su Distinguished Name (DN).
  - **Usar TLS (Transport Layer Security) lookups** — esta opción permite a LDAP enviar información encriptada de nombres de usuario y contraseñas al servidor LDAP antes de la autenticación.
- **Activar Kerberos** — Kerberos es un sistema de seguridad que proporciona servicios de autenticación en red. Para más información acerca Puede elegirse entre tres opciones:
  - **Reino** — esta opción permite acceder a una red que utiliza Kerberos, compuesta por uno o varios servidores (también conocidos como KDCs) y un número potencialmente alto de usuarios.
  - **KDC** — esta opción permite acceder al Centro de Distribución de Claves (Key Distribution Center, KDC), una máquina que emite tickets de Kerberos (a veces se llama también Ticket Granting Server o TGS).

- **Servidor de Administración** — esta opción permite acceder a un servidor que ejecute **kadmind**.
- **Activar la autenticación SMB** — configura PAM para usar un servidor SMB para la autenticación de usuarios. Deberá proporcionarse dos tipos de información:
  - **Servidor SMB** — indica qué servidor SMB contactará la estación de trabajo para la autenticación.
  - **Grupo de trabajo SMB** — indica en qué grupo de trabajo están los servidores SMB.

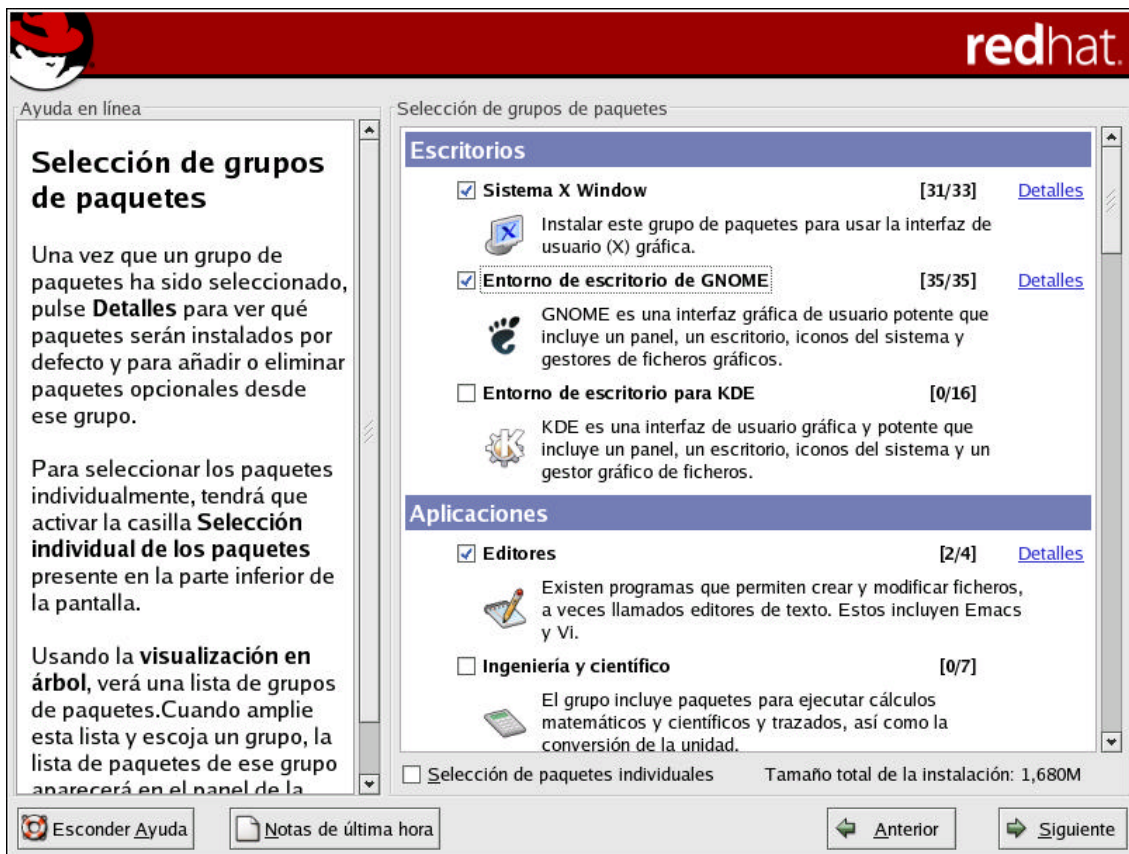
### 3.23 Selección de grupos de paquetes

Tras haber seleccionado las particiones y haberlas configurado para su formateo, ya está preparada la instalación de los paquetes. A no ser que se elija una instalación de tipo personalizado, el programa de instalación elegirá la mayoría de los. Por ejemplo, si se está realizando una instalación de tipo Escritorio Personal, se verá una pantalla similar a la siguiente:



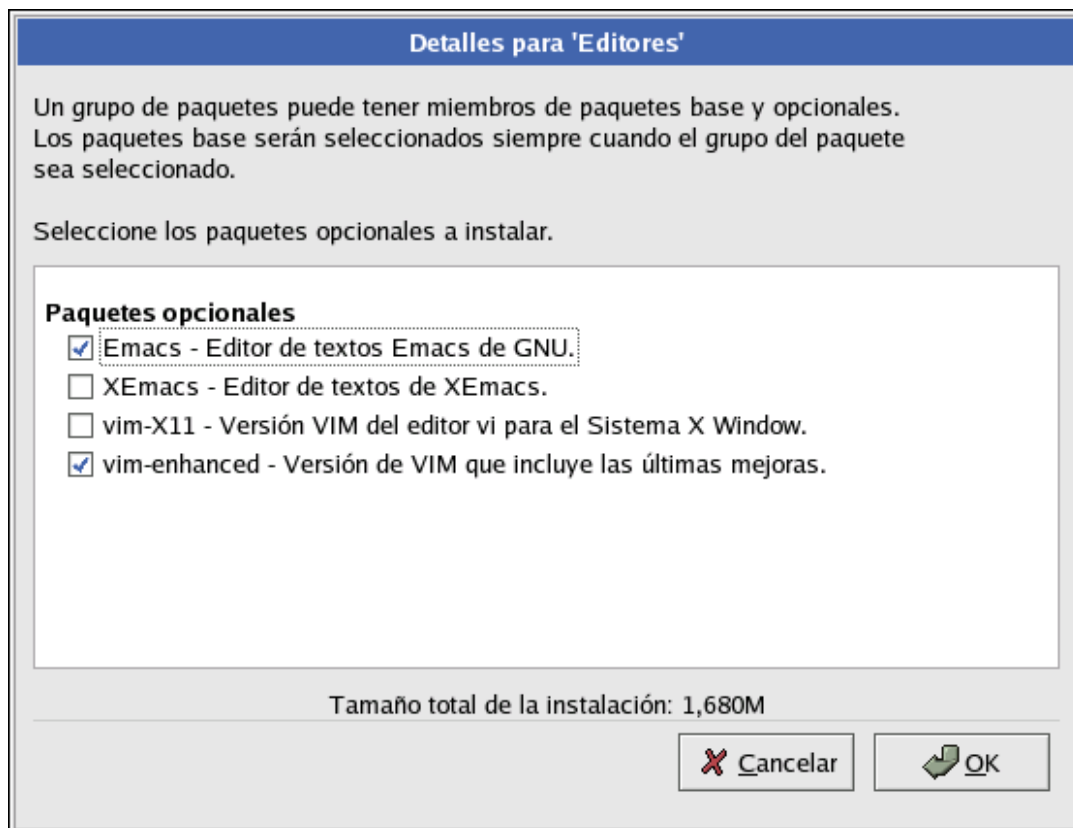
**Figura 3.t Instalación predeterminada para Escritorio Personal**

Para seleccionar paquetes individualmente, debe seleccionarse la casilla de verificación **Personalizar el conjunto de paquetes a instalar**. Puede seleccionarse grupos de paquetes, los cuales agrupan componentes de acuerdo a una función (por ejemplo, **Sistema X Window** y **Editores**), paquetes individuales, o una combinación de los dos. Para seleccionar un componente, Click en la casilla de verificación (Figura 3.u).



**Figura 3.u Selección de grupos de paquetes**

Se selecciona los componentes que se desea instalar. Al seleccionar **Todo** (al final de la lista de componentes) en una instalación tipo personalizado se instalarán todos los paquetes incluidos con Red Hat Linux. Una vez seleccionado el grupo de paquetes, Click en **Detalles** para visualizar los paquetes que se instalarán por defecto y los paquetes que se desee eliminar o añadir a ese grupo.



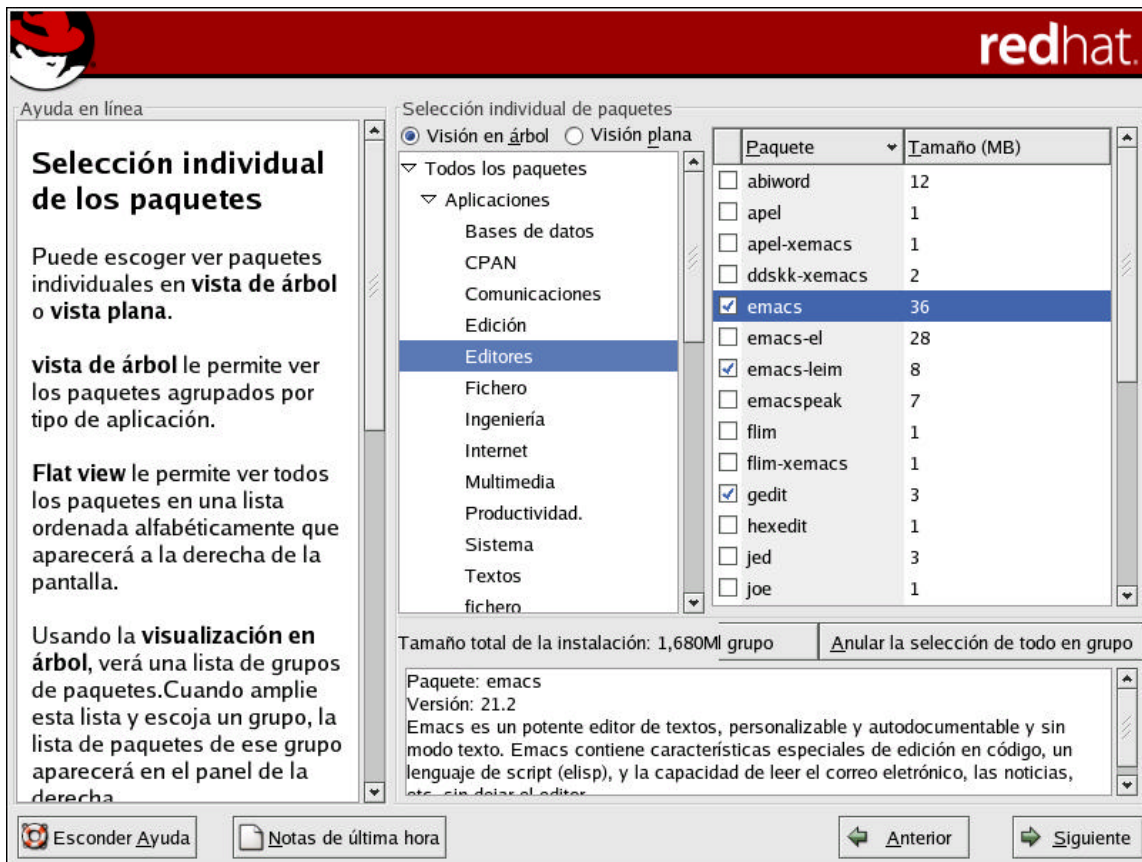
**Figura 3.v Detalles del grupo de paquetes**

Para seleccionar los paquetes individualmente, se activa la opción **Selección individual de paquetes** en la parte inferior de la pantalla.

### **3.23.1. Selección individual de paquetes**

Tras haber seleccionado los paquetes que se quiera instalar, podrá seleccionarse o anularse la selección de los paquetes individualmente usando el ratón (Figura 3.w). Puede escogerse visualizar los paquetes individuales en **Vista de árbol** o **Vista plana**. La **Vista de árbol** permite ver los paquetes agrupados según el tipo de aplicación. La **Vista plana** permite ver todos los paquetes listados en orden alfabético en la parte derecha de la pantalla.





**Figura 3.w Selección individual de paquetes**

Utilizando la **Vista de árbol**, se observa una lista de grupos de paquetes. Cuando se despliegue esta lista (haciendo doble click en la flecha de la carpeta al lado del nombre del paquete) y se escoja un grupo, la lista de paquetes en ese grupo aparecerá en el panel de la derecha. La **Vista plana** permite ver todos los paquetes en orden alfabético a la derecha de la pantalla. Para clasificarlos alfabéticamente, Click en la pestaña **Paquete**. Para clasificarlos por tamaño, Click en la pestaña **Tamaño (MB)**.

Para seleccionar un paquete individual, doble click en la casilla de verificación situada junto al nombre del paquete. Si la casilla de verificación aparece marcada, esto quiere decir que será instalado. Para más información sobre un paquete específico, Click sobre el nombre del paquete. Aparecerá información acerca del paquete en la parte inferior de la pantalla. Se puede seleccionar o anular la selección de todos los paquetes que pertenecen a un grupo en particular haciendo click en los botones **Seleccionar todo en el grupo** o **Anular la selección de todo en el grupo**.



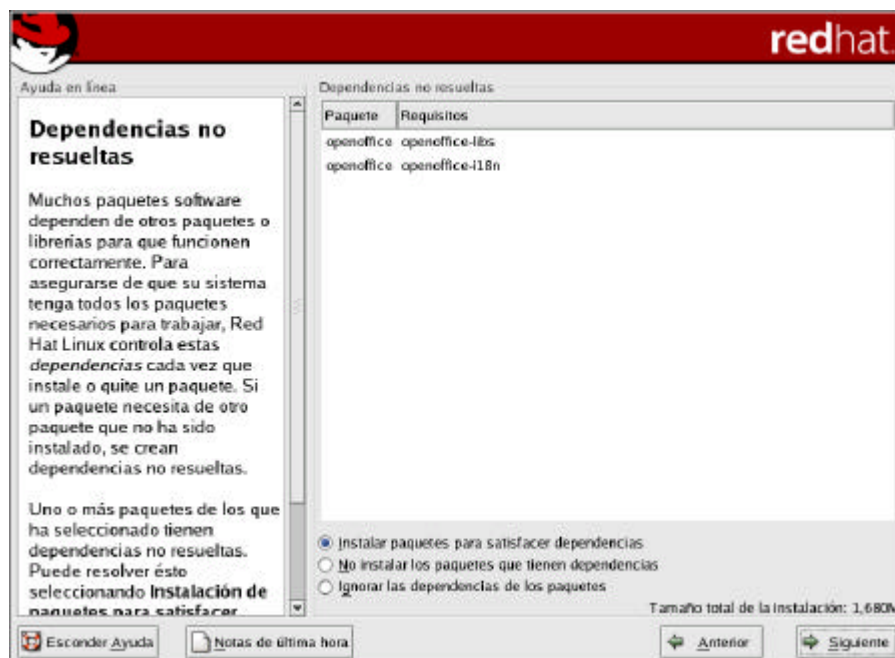
### Nota

Algunos paquetes (como el kernel o algunas librerías) son necesarios para poder ejecutar el sistema y no están disponibles para ser o no seleccionados. Estos paquetes básicos se seleccionan por defecto.

### 3.23.2. Dependencias no resueltas

Muchos paquetes de software, para poder funcionar correctamente, dependen de otros paquetes de software que deben estar instalados en el sistema. Por ejemplo, la mayoría de las utilidades gráficas de administración de Red Hat requieren los paquetes `python` y `pythonlib`. Para estar seguro de que el sistema tiene todos los paquetes que necesita para poder ser completamente funcional, el programa de instalación comprueba las *dependencias* de los paquetes cada vez que instala o borra paquetes de software. Si algún paquete necesita otro paquete que no ha seleccionado para instalar, el programa mostrará una lista de estas dependencias no resueltas y dará la oportunidad de resolverlas (Figura 3.x).

La pantalla **Dependencias no resueltas** tan sólo aparece si los paquetes seleccionados la requieren. En la parte inferior de la pantalla, bajo la lista de los paquetes faltantes, una casilla de verificación **Instalar paquetes para satisfacer las dependencias** estará seleccionada por defecto. Si se deja seleccionada, el programa de instalación resolverá las dependencias automáticamente añadiendo los paquetes necesarios en función de la lista de paquetes que haya seleccionado.



**Figura 3.x Dependencias no resueltas**



Si no se desea instalar paquetes que requieran otros paquetes, se selecciona **No instalar paquetes que tengan dependencias**. Para instalar sólo los paquetes seleccionados y dejar las dependencias sin resolver, se selecciona **Ignorar dependencias de paquetes**.

### 3.24 Preparación para la instalación

Deberá verse una pantalla de preparación para la instalación de Red Hat Linux. Si se necesita documentación se consulta en /root/install.log una vez que ha re arrancado el sistema. Para cancelar este proceso de instalación, basta reiniciar la máquina.

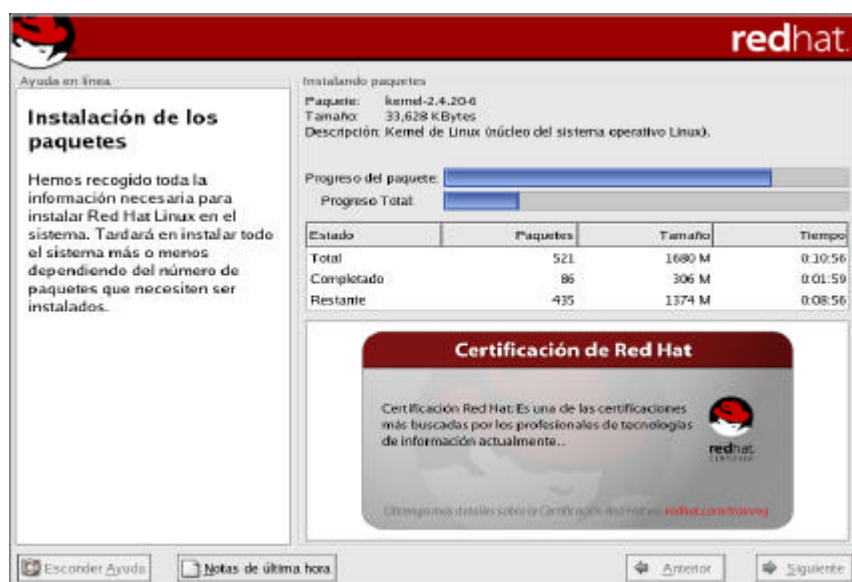


#### Aviso

Si, por algún motivo, no se quiere continuar con el proceso de instalación, ésta es la última oportunidad para cancelar de forma segura el proceso y re arrancar la máquina. Una vez que haya pulsado el botón **Siguiente**, se escribirán las particiones y se instalarán los paquetes. Si se desea abortar la instalación, deberá volver a arrancarse antes de que el disco duro sea reescrito.

### 3.25 Instalación de paquetes

A partir de este momento, no podrá hacerse nada hasta que todos los paquetes hayan sido instalados (Figura 3.y), excepto pasar a las demás terminales virtuales. La rapidez de este proceso dependerá del número de paquetes que se haya seleccionado y de la velocidad de la computadora.

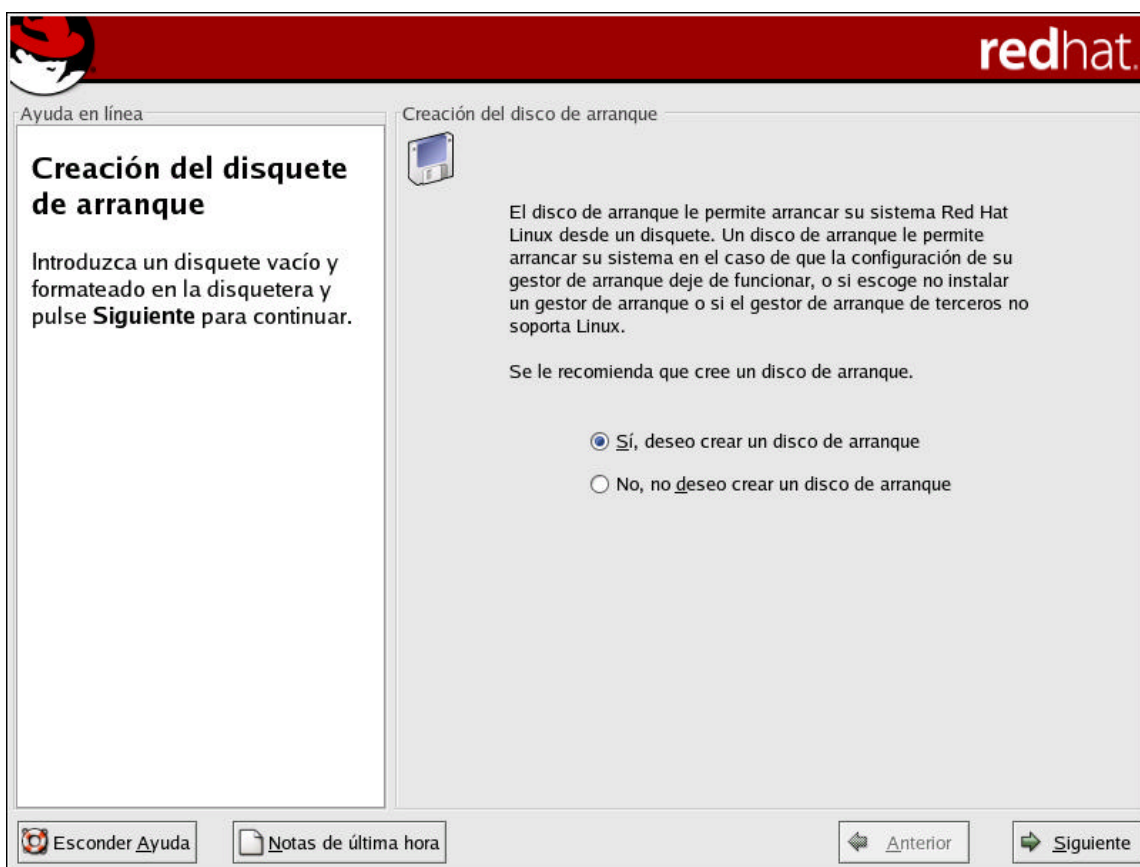


**Figura 3.y Instalación de paquetes**

### 3.26 Creación de un disquete de arranque

Para crear un disco de arranque, se introduce un disco en blanco, formateado en su unidad de disco (Figura 3.z) y Click en **Siguiente**. Se recomienda crear un disquete de arranque. Si, por alguna razón, el sistema no fuera capaz de arrancar adecuadamente usando GRUB, LILO, u otro cargador externo, un disco de arranque permitirá arrancar de forma adecuada el sistema Red Hat Linux.

Tras un pequeño tiempo de espera, el disquete de arranque estará creado. Si se necesitara crear otro disco de inicio una vez terminada la instalación esto podrá hacerse más tarde.

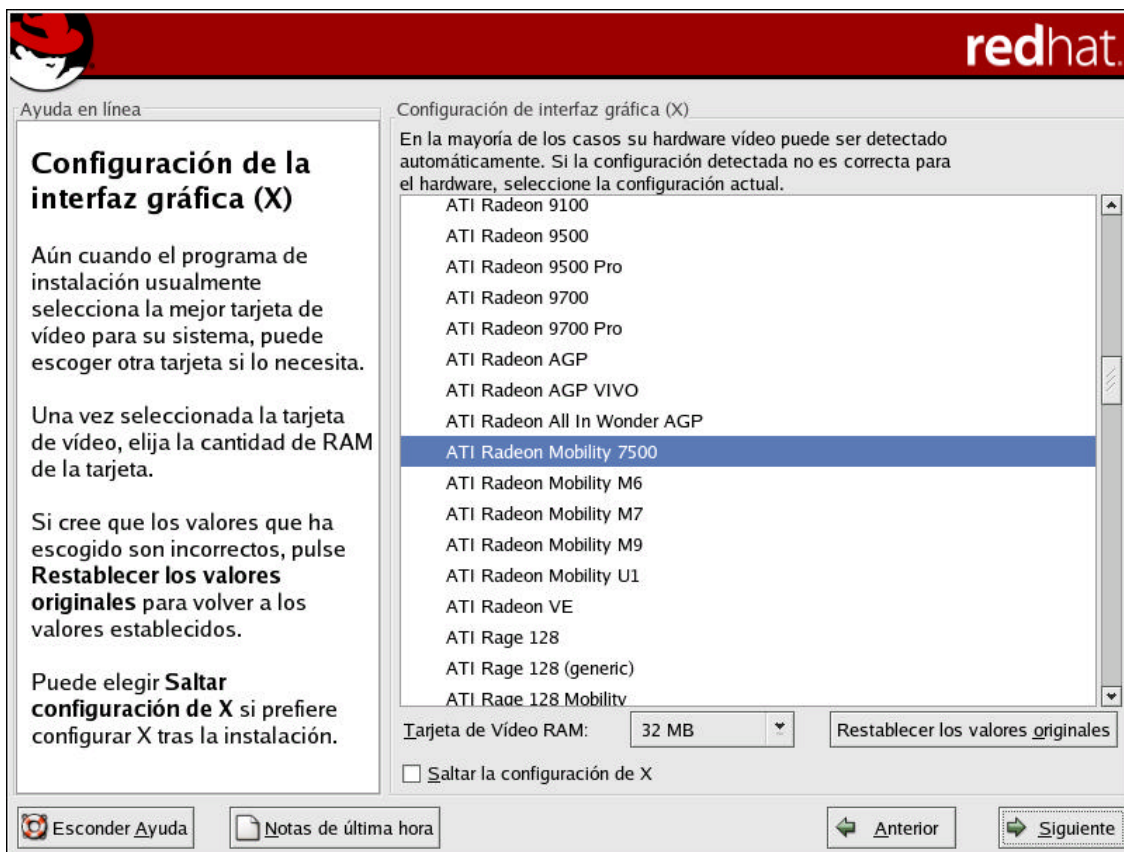


**Figura 3.z Creación del disquete de arranque**

### 3.27 Configuración de la tarjeta de vídeo

El programa de instalación a continuación proporcionará una lista de tarjetas de vídeo entre las que escoger. Si se decide instalar los paquetes del Sistema X Window, se presenta la oportunidad de configurar un servidor X para el sistema. Si la tarjeta de vídeo no aparece en la lista (Figura 3.aa), X puede que no la soporte. No obstante, si se

posee conocimiento técnico sobre la tarjeta, puede escogerse **Tarjeta no listada** e intentar configurarla al hacer corresponder su chipset de tarjeta de vídeo con uno de los servidores X disponibles.



**Figura 3.aa Configuración de la tarjeta de vídeo**

A continuación, deberá introducirse la cantidad de memoria instalada en la tarjeta de vídeo. No se daña la tarjeta si se elige más memoria de la que tiene disponible, pero el servidor X puede no funcionar correctamente. Si se decide que los valores seleccionados son incorrectos, puede hacerse click en el botón **Restaurar valores originales** para volver a los valores sugeridos por el programa de instalación. Puede seleccionarse también la opción **Saltarse la Configuración de X** si se prefiere configurar X después de la instalación o si se quiere hacerlo nunca.

### 3.28 Configuración de X — Control y personalización

Para completar la configuración de X, deberá configurarse el monitor y personalizarse la configuración de X.

### 3.28.1 Configuración del monitor

El programa de instalación presentará una lista de monitores de la cual seleccionar el apropiado. Desde esta lista, puede usarse el monitor que se detecte de forma automática o escoger otro monitor.



#### Nota

Si se instala el sistema Red Hat Linux en una portátil con pantalla LCD, deberá seleccionarse el modelo **Genérico** más adecuado.

Si el monitor no aparece en la lista, se selecciona el modelo **Genérico** más apropiado dentro de los modelos disponibles. Si se selecciona un monitor **Genérico**, el programa de instalación sugerirá valores de sincronización horizontales y verticales. Estos valores suelen encontrarse en la documentación que acompaña al monitor o pueden obtenerse a través del vendedor o fabricante del monitor.



#### Atención

No se selecciona un monitor *similar* al monitor propio a menos de estar seguro de que el monitor seleccionado no excede las capacidades del monitor. De lo contrario puede sobrecargarse y dañarse o destruirse.

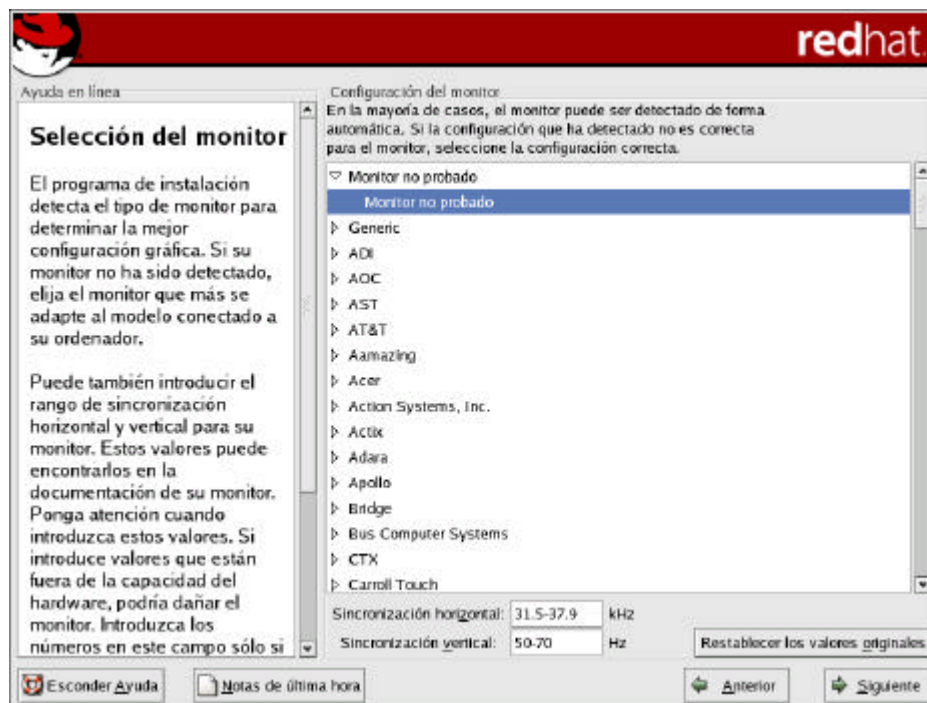


Figura 3.bb Selección del monitor

Los rangos de sincronización horizontal y vertical que el programa de instalación sugiere para su monitor se mostrarán debajo de la lista de monitores. Si se decide que los valores que se ha introducido son incorrectos, puede pulsarse el botón **Restaurar Valores Originales** para volver a los valores recomendados originalmente.

### 3.28.2 Configuración personalizada

Se elige la densidad del color y la resolución para la configuración de X. Si está realizándose una instalación personalizada o de servidor, también puede escogerse si desea arrancar el sistema en modo gráfico o texto una vez que la instalación se termine. A menos que se tenga necesidades especiales, se recomienda el arranque en ambiente gráfico (similar al entorno Windows). Si se elige arrancar en un ambiente de texto, se presentará una línea de comandos (similar al entorno DOS). Las instalaciones de escritorio personal y de estación de trabajo automáticamente arrancarán en ambiente gráfico.



**Figura 3.cc Personalización de X**

### 3.29 Fin de la instalación

El programa de instalación pedirá que prepare el sistema para reiniciarse, lo cual incluye retirar floppys y cd-rom. Si no se instaló un gestor de arranque y se configuró, necesitará usarse el disco de arranque que se ha creado durante la instalación. Después de que la secuencia de encendido se haya terminado, deberá verse el intérprete de comandos del gestor de arranque gráfico en el que puede hacerse cualquiera de las siguientes cosas:

- Apretar [Intro] — se reiniciará la entrada de inicio por defecto.
- Seleccionar una etiqueta de arranque seguida de [Intro] — provocará que el gestor de arranque inicie el sistema operativo correspondiente a la etiqueta de arranque. (Apretar [?] o [Tab] en el intérprete de comandos del cargador de arranque en modo texto para una lista de etiquetas de arranque válidas.)
- No hacer nada — tras un período de espera, LILO, (por defecto, cinco segundos) iniciará la primera partición automáticamente.

Deberá verse una o más ventanas de mensajes. También deberá verse un intérprete de comandos login: o una pantalla gráfica de login (si se ha instalado el sistema X Window y se ha escogido arrancarlo en modo gráfico por defecto).

## 4. Gestores de arranque

Antes de poder ejecutar Red Hat Linux, debe ser cargado en memoria por un programa especial llamado *gestor de arranque*. El programa de gestor de arranque existe en el disco duro primario del sistema (o en otros dispositivos) y es responsable de la carga del kernel de Linux con sus archivos necesarios, o, en algunos casos, de otros sistemas operativos en la memoria.

### 4.1 Gestores de arranque y arquitectura del sistema

Cada arquitectura de sistemas que pueda ejecutar Red Hat Linux usa un gestor de arranque diferente. Por ejemplo, la arquitectura Alpha usa el gestor de arranque *aboot*, mientras que la arquitectura Itanium usa el gestor de arranque *ELILO*. Este capítulo explica comandos y opciones de configuración para los dos cargadores de arranque suministrados con Red Hat Linux para la arquitectura x86: GRUB y LILO.

### 4.2 GRUB

*GNU GRand Unified Boot loader* o GRUB es un programa que habilita al usuario a seleccionar qué sistema operativo instalado o kernel descargar en el momento de arranque del sistema. Permite también que el usuario transmita argumentos al kernel.

#### 4.2.1 Proceso de arranque en un sistema x86 y GRUB

Esta sección explica con más detalle el papel específico que desempeña GRUB al arrancar un sistema x86. GRUB se carga asimismo en la memoria en las diferentes etapas:

1. *La etapa 1 o cargador de arranque primario se lee en la memoria con el BIOS desde el MBR.* El gestor de arranque primario existe en menos de 512 bytes de espacio en disco entre el MBR y es capaz de cargar bien sea la etapa 1.5 o la etapa 2 del gestor de arranque.
2. *El gestor de arranque de la etapa 1 lee en la memoria al gestor de arranque de la etapa 1.5, si es necesario.* Determinado hardware requiere un paso intermedio para obtener el cargador de arranque de la etapa 2. Esto sucede a menudo cuando la partición */boot* está por encima de 1024 cilindros de disco duro o cuando se usa el modo LBA. Este cargador de arranque de la etapa 1.5 se encuentra en la partición */boot* o en una pequeña parte del MBR y la partición */boot*.
3. *La etapa 2 o el gestor de arranque secundario se lee en la memoria.* El gestor de arranque secundario visualiza el menú GRUB y el entorno del comando. Esta interfaz le permite seleccionar qué sistema operativo o kernel de Linux



arrancar, pasar argumentos al kernel o ver los parámetros del sistema, tales como la RAM disponible.

4. *El gestor de arranque secundario lee el sistema operativo o el kernel y initrd en la memoria.* Una vez que GRUB determina qué sistema operativo iniciar, éste lo carga en la memoria y transfiere el control de la máquina a dicho sistema operativo.

El método de arranque usado para arrancar Red Hat Linux se conoce como método de *carga directa* porque el gestor de arranque carga el sistema operativo directamente. No existe punto medio entre el gestor de arranque y el kernel.

El proceso de arranque usado por otros sistemas operativos puede variar. Por ejemplo, los sistemas operativos de Microsoft DOS y Windows, así como otros sistemas operativos de propietarios, se cargan mediante un método de arranque de *carga encadenada*. Bajo este método, el MBR señala el primer sector de la partición que tiene el sistema operativo. Allí encuentra los archivos necesarios para arrancar el sistema operativo. GRUB soporta ambos métodos de arranque, directo y carga encadenada, permitiendo arrancar desde casi cualquier sistema operativo.



#### **Aviso**

Durante la instalación, la instalación de DOS de Microsoft y Windows sobre escriben completamente el MBR, destruyendo cualquier cargador de arranque ya existente. Si se crea un sistema de arranque dual, es preferible que instalar el sistema operativo Microsoft primero.

### **4.2.2 Funciones de GRUB**

GRUB contiene una serie de funciones que lo convierten en el método favorito respecto al resto de gestores de arranque disponibles para la arquitectura x86. A continuación tiene una lista de las características más importantes:

- *GRUB proporciona un entorno pre-OS basado en comandos verdaderos para máquinas x86.* Esto proporciona una flexibilidad máxima en la carga de los sistemas operativos con determinadas opciones o con la recopilación de información sobre el sistema. Durante muchos años arquitecturas que no son x-86 han usado entornos previos al sistema operativo que permiten arrancar el sistema desde una línea de comandos. Mientras que algunas características del comando están disponibles con LILO y otros gestores de arranque para x86, GRUB tiene una mayor variedad de características.
- *GRUB soporta el modo Direccionamiento Lógico de Bloques (LBA).* El modo LBA coloca la conversión de direccionamiento utilizada para buscar archivos en la unidad de disco duro del firmware y se utiliza en muchos discos IDE y en todos los discos duros SCSI. Antes de LBA, los gestores de arranque



encontraban la limitación del cilindro 1024 del BIOS, donde el BIOS no podía encontrar un archivo después de ese cabezal de cilindro del disco. El soporte LBA permite que GRUB arranque los sistemas operativos desde las particiones más allá del límite de 1024 cilindros, siempre y cuando el BIOS del sistema soporte el modo LBA. Las revisiones más modernas de la BIOS soportan el modo LBA.

- *GRUB puede leer las particiones ext2.* Esto permite que GRUB acceda a su archivo de configuración, `/boot/grub/grub.conf`, cada vez que el sistema arranca, eliminando la necesidad que tiene el usuario de escribir una nueva versión de la primera etapa del gestor de arranque al MBR en caso de que se produzcan cambios de la configuración. El único caso en el que el usuario necesitaría reinstalar GRUB en el MBR es en caso de que la localización física de la partición `/boot` se traslade en el disco.

### 4.3 Instalación de GRUB

Si durante el proceso de instalación de Red Hat Linux no se instaló GRUB, se puede hacer después. Una vez instalado se convierte en el gestor de arranque por defecto. Antes de instalar GRUB, debería asegurarse de que cuenta con el último paquete disponible de GRUB o use el paquete GRUB desde los CD-ROMs de instalación de Red Hat Linux. Para instrucciones sobre la instalación de paquetes, vea el capítulo titulado *Gestión de paquetes con RPM* en el *Manual de personalización de Red Hat Linux*.

Una vez que el paquete GRUB esté instalado, abra un intérprete de comandos de la shell, y ejecute el comando `/sbin/grub-install <location>`, donde `<location>` es la localización de la etapa 1 de GRUB en la que el gestor de arranque debería ser instalado.

El siguiente comando instala GRUB en el MBR del dispositivo IDE maestro en el bus IDE primario: `/sbin/grub-install /dev/hda`. La próxima vez que arranque el sistema, el menú del gestor de arranque gráfico GRUB aparecerá antes del que el kernel se cargue en memoria.

### 4.4 Terminología de GRUB

Una de las cuestiones más importantes que deben entenderse antes de utilizar GRUB es cómo el programa hace referencia a los dispositivos, por ejemplo, a los discos duros y a las particiones. Esta información es muy importante si desea configurar GRUB para arrancar varios sistemas operativos.

#### 4.4.1 Nombres de dispositivos

Suponga que un sistema tiene más de un disco duro. El primer disco duro del sistema es llamado `(hd0)` por GRUB. La primera partición en ese disco es llamada `(hd0,0)`, y la quinta partición en el segundo disco duro es llamada `(hd1,4)`. En general, la

nomenclatura utilizada para los sistemas de archivos al usar GRUB se desglosa del siguiente modo:

(*<type-of-device><bios-device-number>,<partition-number>*)

Los paréntesis y las comas son muy importantes en el nombre. *<tipo-de-dispositivo>* hace referencia a si es un disco duro (hd) o una unidad de disquete (fd).

*<número-dispositivo-bios>* es el número de dispositivo según la BIOS del sistema, empezando desde 0. El disco duro IDE principal tiene asignado el número 0 y el disco duro IDE secundario el número 1. El orden es aproximadamente equivalente al modo en el que el kernel de Linux organiza los dispositivos con letras, donde la letra *a* en *hda* corresponde al número 0, y la letra *b* en *hdb* corresponde al número 1, y así sucesivamente.



#### **Nota**

El sistema de numeración de GRUB para los dispositivos empieza por 0 y no por 1. Este es uno de los errores que cometen con más frecuencia los usuarios que empiezan a utilizar GRUB.

*<número-partición>* hace referencia al número de una partición concreta en dicho dispositivo. Al igual que en el caso de *<número-dispositivo-bios>*, la numeración de las particiones empieza por 0. Aunque la mayoría de las particiones se especifican con números, si el sistema usa particiones BSD a éstas se hará referencia con letras, por ejemplo *a* o *c*.

GRUB usa las reglas siguientes para denominar los dispositivos y las particiones:

- No es relevante si los discos duros que utiliza son IDE o SCSI. Todos los discos duros empiezan con *hd*. Las unidades de disquete empiezan con *fd*.
- Para especificar todo un dispositivo sin respetar sus particiones, simplemente debe suprimir la coma y el número de partición. Esto es importante para indicarle a GRUB que configure el registro MBR para un disco concreto. Por ejemplo, *(hd0)* especifica la MBR en el primer dispositivo y *(hd3)* especifica la MBR en el cuarto dispositivo.
- Si tiene varios discos duros, es muy importante saber el orden de la unidad de arranque de la BIOS. Esto es muy sencillo si sólo tiene discos IDE o SCSI, pero si tiene una combinación de ambos, el asunto se complica un poco.

#### **4.4.2 Nombres de archivos y listas de bloqueo**

Al escribir comandos en GRUB que hagan referencia a un archivo, como una lista de menús que debe usarse para permitir el arranque de varios sistemas operativos, debe

incluir el archivo inmediatamente después de especificar el dispositivo y la partición. Una especificación de archivo de ejemplo que haga referencia a un nombre de archivo absoluto se organiza del modo siguiente:

```
(<type-of-device><bios-device-number>,<partition-number>)/path/to/file
```

La mayoría de las veces, un usuario especificará los archivos por la ruta del directorio en esa partición más el nombre del archivo. También puede especificar archivos a GRUB que no aparecen realmente en el sistema de archivos, tal como un gestor de arranque de cadena que aparece en los primeros bloques de la partición. Para especificar estos archivos, deberá indicar una *lista de bloques*, que indique a GRUB, bloque por bloque, la ubicación exacta del archivo en la partición. Puesto que un archivo puede estar formado por varios conjuntos de bloques, hay un modo específico de escribir listas de bloques. Cada ubicación de sección de archivo se describe con un número de desplazamiento de bloques seguido de un número de bloques de ese punto de desplazamiento, y las secciones se colocan juntas de forma ordenada y separadas por comas.

La siguiente es una lista de bloques de ejemplo:

```
0+50,100+25,200+1
```

Esta lista de bloques indica a GRUB que debe utilizar un archivo que empieza en el primer bloque de la partición y que usa los bloques del 0 al 49, del 99 al 124, y el 199. Saber cómo escribir listas de bloques es útil al utilizar GRUB para cargar sistemas operativos que usan el método de carga encadenada, como Microsoft Windows. Puede suprimir el número de desplazamiento de bloques si empieza por el bloque 0. Por ejemplo, el archivo de carga encadenada de la primera partición del primer disco duro tendrá el nombre siguiente:

```
(hd0,0)+1
```

Lo siguiente muestra el comando chainloader con una designación de lista de bloques similar en la línea de comandos de GRUB después de establecer el dispositivo correcto y la partición adecuada como raíz:

```
chainloader+1
```

#### 4.4.3 Sistema de archivos raíz de GRUB

Algunos usuarios se confunden con el uso del término "sistema de archivos" en GRUB. Es importante recordar que el sistema de archivos raíz de GRUB no tiene nada que ver con el sistema de archivos raíz de Linux. El sistema de archivos raíz de GRUB es la partición raíz de un dispositivo concreto. GRUB usa esta información para montar el dispositivo y carga los archivos desde el mismo.

Con Red Hat Linux, una vez que GRUB ha cargado la partición raíz (que es lo mismo que la partición /boot y contiene el kernel de Linux), el comando kernel puede ejecutarse con la localización del archivo del kernel como una opción. Una vez que el kernel de Linux inicia, establece el sistema de archivos raíz con los cuales los usuarios de Linux están familiarizados. El sistema de archivos root de GRUB original y los montajes deben olvidarse en este punto; la única finalidad de su existencia era arrancar el archivo del kernel.

#### **4.5. Interfaces de GRUB**

GRUB dispone de tres interfaces eficaces que proporcionan distintos niveles de funcionalidad. Cada una de estas interfaces le permite arrancar el kernel de Linux u otros sistemas operativos.

Las interfaces son como sigue:

##### **Interfaz de menú**

Si el programa de instalación de Red Hat Linux ha configurado automáticamente GRUB, ésta es la interfaz que ya conoce. En esta interfaz hay un menú de sistemas operativos o kernels preconfigurados con sus propios comandos de arranque en forma de lista ordenada por nombre, después de arrancar el sistema por primera vez. Puede utilizar las teclas de flecha para seleccionar una opción en lugar de la selección por defecto y pulsar la tecla [Intro] para arrancar el sistema.

Como alternativa, se puede establecer un período de inactividad, de modo que GRUB inicie la carga de la opción por defecto. Presione la tecla [e] para entrar en la interfaz del editor o la tecla [c] para cargar la interfaz de línea de comandos.

##### **Interfaz del editor de menú de entrada**

Para tener acceso al editor de entradas del menú, presione la tecla [e] desde el menú del gestor de arranque. Los comandos de GRUB de dicha entrada se muestran aquí y puede alterar estas líneas de comandos antes de arrancar el sistema operativo agregando una línea de comandos {o} inserta una nueva línea después de la línea actual y [O] inserta una nueva línea antes de ella), modificandola ([e]), o borrando una ([d]).

Una vez realizados los cambios, la tecla [b] ejecuta los comandos y arranca el sistema operativo. Con la tecla [Esc] se omiten los cambios y el usuario vuelve a la interfaz de menú estándar. Con la tecla [c] se carga la interfaz de línea de comandos.

## Interfaz de línea de comandos

Esta es la interfaz de GRUB más básica, pero también la que proporciona un mayor control. En esta interfaz puede escribir cualquier comando de GRUB seguido de la tecla [Intro] para proceder a la ejecución correspondiente. Esta interfaz cuenta con algunas funciones similares a las de shell avanzadas, incluyendo el uso de [Tab] para autocompletar, y las combinaciones de teclas con [Ctrl] al escribir comandos, tales como [Ctrl]-[a] para moverse al comienzo de la línea y [Ctrl]-[e] para moverse al final. Además, las teclas de flecha, [Inicio], [Fin], y [Supr] funcionan de forma similar al bash shell.

### 4.5.1 Orden de uso de Interfaces

Cuando GRUB carga la segunda etapa de su gestor de arranque, primero busca por su archivo de configuración. Cuando lo encuentra, lo utiliza para crear la lista de menú y despliega la interfaz de menú. Si no puede encontrar el archivo de configuración o si éste no se puede leer, GRUB carga la interfaz de línea de comandos para permitirle al usuario escribir manualmente los comandos necesarios para completar el proceso de arranque.

En el caso de que el archivo de configuración no sea válido, GRUB imprimirá el error y solicitará la introducción de valores. Esto puede ser muy útil, porque podrá ver con exactitud donde está el problema y corregirlo en el archivo. Si pulsa cualquier tecla se volverá a cargar la interfaz de menú, donde podrá modificar la opción de menú y corregir el problema según el error que GRUB haya notificado. Si la corrección falla, GRUB informa del error y puede empezar de nuevo.

## 4.6 Comandos de GRUB

GRUB permite varios comandos en su línea de comandos. Algunos de los comandos aceptan opciones después del nombre y estas opciones deben ir separadas del comando por comas y de otras opciones de esa línea por caracteres de espacio.

En la lista siguiente se indican los comandos más útiles:

- **boot** — Arranca el sistema operativo o gestor de encadenamiento que se ha especificado y cargado previamente.
- **chainloader <nombre-archivo>** — Carga el archivo especificado como gestor de encadenamiento. Para extraer el archivo en el primer sector de la partición especificada, puede utilizar +1 como nombre de archivo.
- **displaymem** — Muestra el uso actual de memoria, en función de la información de la BIOS. Esto es útil si no está seguro de la cantidad de RAM que tiene un sistema y todavía tiene que arrancarlo.

- `initrd <nombre-archivo>` — Le permite especificar un disco RAM inicial para utilizarlo al arrancar. `initrd` es necesario cuando el kernel necesita ciertos módulos para poder arrancar adecuadamente, tales como cuando la partición se formatea con el sistema de archivos `ext3`.
- `install <stage-1> <install-disk> <stage-2> p <config-file>` — Instala GRUB en la MBR del sistema.

Cuando esté usando el comando `install`, el usuario debe especificar lo siguiente:

- `<stage-1>` — Significa un dispositivo, partición y archivo donde el primer gestor de arranque puede ser encontrado, tal como `(hd0,0)/grub/stage1`.
- `<install-disk>` — Especifica el disco donde la etapa 1 del gestor de arranque debería ser instalado, tal como `(hd0)`.
- `<stage-2>` — Pasa la ubicación de la etapa 2 del gestor de arranque a la etapa 1, tal como `(hd0,0)/grub/stage2`.
- `p <config-file>` — Esta opción le indica al comando `install` que busque por el archivo de configuración de menú especificado por `<config-file>`. Un ejemplo de una ruta válida al archivo de configuración es `(hd0,0)/grub/grub.conf`.



#### **Aviso**

Este comando sobrescribirá cualquier información del MBR. Si se ejecuta, cualquier herramienta utilizada para arrancar el sistema operativo que no sea GRUB se perderá.

- `kernel <kernel-file-name> <option-1> <option-N>` — Especifica el archivo del kernel a cargar desde el sistema de archivos raíz de GRUB cuando se esté usando la carga directa para arrancar el sistema operativo. Las opciones pueden estar después del comando `kernel` y se pasarán al kernel cuando éste se cargue.

Para el sistema Red Hat Linux, es probable que tenga una línea similar a la siguiente:

```
kernel/vmlinuzroot=/dev/hda5
```

Esta línea especifica que el archivo `vmlinuz` se carga desde un sistema de archivos raíz de GRUB, por ejemplo, `(hd0,0)`. También se transfiere una opción al kernel que especifica que el sistema de archivos raíz del kernel del Linux debe encontrarse, al cargarse, en `hda5`, la quinta partición en el primer disco.

duro IDE. Después de esta opción se pueden insertar varias opciones, si es necesario.

- `root <device-and-partition>` — Configura la partición raíz de GRUB para que sea el dispositivo y la partición concreta, por ejemplo, `(hd0,0)`, y monta la partición de modo que se puedan leer los archivos.
- `rootnoverify <device-and-partition>` — Realiza las mismas funciones que el comando `root` pero no monta la partición.

Hay otros comandos disponibles aparte de los indicados. Escriba `info grub` para obtener una lista completa de los comandos.

## 4.7 archivo de configuración de menú de GRUB

El archivo de configuración (`/boot/grub/grub.conf`), usado para crear la lista en la interfaz de menú de GRUB de los sistemas operativos para el arranque, básicamente permite al usuario seleccionar un grupo predefinido de comandos para su ejecución.

### 4.7.1 Comandos especiales del archivo de configuración

Los comandos siguientes sólo pueden usarse en el archivo de configuración de menú de GRUB:

- `color <normal-color> <selected-color>` — Le permite configurar los colores específicos que se usarán en el menú. Se configuran dos colores: uno de fondo y otro de primer plano. Use nombres de colores simples, tales como `red/black`. Por ejemplo:

```
colorred/blackgreen/blue
```

- `default <nombre-título>` — Nombre del título por defecto de la entrada que se cargará si se supera el tiempo de inactividad de la interfaz de menú.
- `fallback <nombre-título>` — Si se utiliza, el nombre de título de la entrada que deberá probarse si falla el primer intento.
- `hiddenmenu` — Si se utiliza, no se podrá mostrar la interfaz de menú de GRUB ni cargar la entrada `default` si caduca el período `timeout`. El usuario puede ver el menú estándar de GRUB si pulsa la tecla `[Esc]`.
- `password <contraseña>` — Si se utiliza, el usuario que no conozca la contraseña no podrá modificar las entradas de esta opción de menú.

Opcionalmente, puede especificar un archivo de configuración de menú alternativo después de la password <contraseña>. En este caso, GRUB reiniciará el Nivel 2 del gestor de arranque y utilizará este archivo de configuración alternativo para crear el menú. Si se omite este archivo de configuración alternativo del comando, el usuario que sepa la contraseña podrá modificar el archivo de configuración actual.

- **timeout** — Si se utiliza se establece la cantidad de tiempo, en segundos, antes de que GRUB cargue la entrada designada por el comando default.
- **splashimage** — Especifica la ubicación de la imagen de pantalla splash que se utilizará al arrancar.
- **title** — Establece el título que se utilizará con un grupo de comandos concreto para cargar un sistema operativo.

El carácter (#) se puede usar al principio de una línea para insertar comentarios en el archivo de configuración de menú.

#### 4.7.2 Estructura del archivo de configuración

El archivo de configuración de menú de GRUB es /boot/grub/grub.conf. Los comandos para configurar las preferencias globales para la interfaz de menú están ubicados al inicio del archivo, seguido de las diferentes entradas para cada sistema operativo o kernels listados en el menú.

El siguiente es un ejemplo de archivo de configuración de menú muy básico diseñado para arrancar bien sea Red Hat Linux o Microsoft Windows 2000:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz

#sectiontoloadLinux
titleRedHatLinux(2.4.18-5.47)
root(hd0,0)
kernel/vmlinuz-2.4.18-5.47root=/dev/sda2
initrd/initrd-2.4.18-5.47.img

#sectiontoloadWindows2000
titlewindows
rootnoverify(hd0,0)
chainloader+1
```

Este archivo indicará a GRUB que cree un menú con Red Hat Linux como el sistema operativo predeterminado y que establezca un arranque automático después de 10



segundos. Se proporcionan dos secciones, una para cada entrada de sistema, con comandos específicos para la tabla de partición de cada sistema.



#### Nota

Observe que la opción predeterminada está especificada como un número. Esto hace referencia a la primera línea title con la que GRUB se encuentra. Si desea que windows sea predeterminada, cambie el valor default=0 a default=1.

Este capítulo no abarca la configuración de un archivo de configuración del menú de GRUB para arrancar sistemas operativos múltiples. Consulte **Sección 2.11** para una lista de los recursos adicionales.

## 4.8 LILO

*LILO* es un acrónimo de *Linux LOader* (cargador) y ha sido usado para arrancar Linux en sistemas x86 por muchos años. Aunque ahora GRUB es el gestor de arranque por defecto, algunos prefieren usar LILO porque les es más familiar y otros porque GRUB puede causar problemas al arrancar determinado tipo de hardware.

### 4.8.1 LILO y el proceso de arranque de x86

LILO se carga así mismo en la memoria casi de forma idéntica a GRUB, con la diferencia de que es un gestor de dos etapas.

1. *La etapa 1 o el gestor de arranque primario se lee en la memoria por la BIOS desde el MBR.* El gestor de arranque primario existe en menos de 512 bytes de espacio en disco dentro del MBR. Su función es cargar la etapa 2 del gestor de arranque y pasarle la información de la geometría del disco.
2. *La etapa 2 o el gestor de arranque secundario se lee en memoria..* El gestor de arranque secundario visualiza la pantalla inicial de Red Hat Linux. Esta pantalla le permite seleccionar el sistema operativo o el kernel de Linux que desee arrancar.
3. *La etapa 2 lee el sistema operativo o el kernel y lleva a cabo initrd en memoria.* Una vez que LILO determina qué sistema operativo iniciar, éste lo carga en la memoria y lleva el control de la máquina a ese sistema operativo.

Una vez que se ha llevado a cabo la etapa 2 en memoria, LILO visualiza la pantalla inicial de Red Hat Linux con los diferentes sistemas operativos o kernel que han sido configurados para arrancar. Por defecto, si Red Hat Linux es el único sistema instalado, **Linux** será la única opción disponible. Si el sistema tiene múltiples procesadores habrá una opción **Linux-up** para el kernel del procesador único y una opción **Linux** para los kernel de múltiples procesadores (SMP). Si LILO está

configurado para arrancar otros sistemas operativos, estas entradas de arranque también aparecerán en pantalla.

Las flechas direccionales permiten al usuario resaltar el sistema operativo deseado y la tecla [Intro] comenzará el proceso de arranque.

Para acceder una línea de comandos boot:, se presiona [Ctrl]-[X].

#### 4.8.2 LILO versus GRUB

En general, LILO funciona de forma parecida a GRUB a excepción de tres diferencias:

- No posee ninguna interfaz del comando interactiva.
- Almacena información sobre la localización del kernel o de si otro sistema operativo se debe cargar en el MBR.
- No puede leer las particiones ext2.

El primer punto significa que el intérprete de comandos para LILO no es interactivo y permite tan sólo un comando con argumentos.

Los últimos dos puntos significan que si usted cambia el archivo de configuración de LILO o instala un kernel nuevo, debe re escribir el gestor de arranque LILO de la etapa 1 al MBR llevando a cabo el comando siguiente:

```
/sbin/lilo-v-v
```

Este método es más arriesgado que el de GRUB, porque un MBR que no haya sido configurado adecuadamente deja el sistema sin poder arrancar. Con GRUB, si el archivo de configuración está configurado de forma errónea, se disparará por defecto la interfaz de la línea de comandos de modo que el usuario pueda arrancar el sistema manualmente.

#### 4.9 Opciones en /etc/lilo.conf

El archivo de configuración de LILO es /etc/lilo.conf. El comando /sbin/lilo usa este archivo para determinar que información debe escribir al MBR.



##### **Aviso**

Si desea modificar /etc/lilo.conf, asegúrese de que ha hecho una copia de seguridad del archivo antes de cualquier otro cambio. Asegúrese de que posee un disquete de arranque que funcione de manera que sea capaz de arrancar el sistema y realizar cambios en el MBR si existe algún problema. Consulte las

páginas de manual para mkbootdisk para obtener más información en la creación de un disco de arranque.

El archivo `/etc/lilo.conf` es usado por el comando `/sbin/lilo` para determinar qué sistema operativo o kernel iniciar, así como para saber donde instalarlo.

Un archivo de ejemplo para `/etc/lilo.conf` será muy parecido a:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
lba32
default=Linux

image=/boot/vmlinuz-2.4.0-0.43.6
label=Linux
initrd=/boot/initrd-2.4.0-0.43.6.img
read-only
root=/dev/hda5

other=/dev/hda1
label=dos
```

Este ejemplo le muestra un sistema configurado para arrancar dos sistemas operativos: Red Hat Linux y DOS. A continuación una vista más detallada de las líneas de este archivo:

- `boot=/dev/hda` — Instruye a LILO para que se instale en el primer disco duro del primer controlador IDE.
- `map=/boot/map` — localiza el archivo del mapa. En uso normal, esto no se debería modificar.
- `install=/boot/boot.b` — Hace que LILO instale el archivo específico como el nuevo sector de arranque de boot. En un uso normal, esto no debería ser alterado. Si falta la línea `install`, LILO asumirá que el archivo `/boot/boot.b` es el predeterminado a utilizar.
- `prompt` — Instruye a LILO a que muestre cualquier cosa que sea referenciado en la línea `message`. No se le recomienda que elimine la línea `prompt`, si la elimina, todavía podrá tener acceso a un intérprete manteniendo pulsada la tecla [Shift] mientras que su máquina empieza a arrancar.

- `timeout=50` — Configura la cantidad de tiempo que LILO esperará la entrada del usuario antes de proceder con el arranque de la entrada de la línea default. Esto se mide en décimas de segundo, con 50 por defecto.
- `message=/boot/message` — Se refiere a la pantalla que LILO visualiza para permitirle seleccionar el sistema operativo o el kernel a arrancar.
- `lba32` — Describe la geometría del disco para LILO. Otra entrada común es `linear`. No debería cambiar esta línea a menos de que esté bien seguro de lo que está haciendo. De lo contrario, pondría su sistema en un estado de no arranque.
- `default=Linux` — Se refiere al sistema operativo por defecto que LILO arrancará de acuerdo a las opciones listadas bajo esta línea. El nombre `Linux` se refiere a la línea `label` bajo cada una de las opciones de arranque.
- `image=/boot/vmlinuz-2.4.0-0.43.6` — Especifica el kernel de Linux para arrancar con esta opción de arranque en particular.
- `label=Linux` — Nombra la opción del sistema operativo en la pantalla de LILO. En este caso, es también el nombre al que se refiere la línea `default`.
- `initrd=/boot/initrd-2.4.0-0.43.6.img` — Se refiere a la imagen *initial ram disk* que se usa en el tiempo de arranque para inicializar los dispositivos que hacen que el inicio del kernel sea posible. El disco ram inicial es una colección de controladores específicos de cada máquina necesarios para operar una tarjeta SCSI, una unidad de disco duro o cualquier otro dispositivo necesario para cargar el kernel. No debería nunca intentar compartir los discos ram iniciales entre las máquinas.
- `read-only` — Especifica que la partición `root` (consulte la línea `root` más abajo) es de sólo lectura y no puede ser alterada durante el proceso de arranque.
- `root=/dev/hda5` — Especifica cual partición de disco a usar como la partición raíz.
- `other=/dev/hda1` — Especifica la partición que contiene DOS.

#### 4.10 Cambiar los niveles de ejecución en el tiempo de arranque

Bajo Red Hat Linux, es posible cambiar el nivel de ejecución predeterminado en el momento de arranque. Si usa LILO, acceda al intérprete de comandos `boot:` presionando [Ctrl]-[X]. Luego escriba:

```
Linux <runlevel-number>
```

En este comando, reemplace el número *<runlevel-number>* con el número de nivel de ejecución a arrancar (de **1** a **5**), o las palabras **single** o **emergency**. Si está usando GRUB como gestor de arranque, siga los pasos siguientes:

- En la pantalla gráfica del gestor de arranque GRUB, seleccione la etiqueta de arranque **Red Hat Linux** y pulse [e] para modificarla.
- Vaya en la parte inferior a la línea del kernel y pulse [e] para modificarla.
- En el intérprete de comandos, escriba el número del nivel de ejecución en el que desea arrancar (desde **1** a **5**), o las palabras **single** o **emergency** y presione [Intro].
- Volverá a la pantalla de GRUB con la información sobre el kernel. Pulse [b] para arrancar el sistema.

#### 4.11 Recursos adicionales

El objetivo de este capítulo sólo es servir de introducción a GRUB y a LILO. Consulte los siguientes recursos para descubrir más cosas sobre cómo funcionan GRUB y LILO.

##### 4.11.1 Documentación instalada

- */usr/share/doc/grub-<version-number>/* — Este directorio contiene muy buena información sobre el uso y configuración de GRUB. El *<version-number>* en la ruta a este archivo corresponde a la versión del paquete de GRUB instalado.
- La página de información de GRUB, a la que se puede acceder si se escribe el comando `info grub`, contiene un tutorial, un manual de referencia para el usuario, un manual de referencia para el programador y un documento de Preguntas más frecuentes (FAQ) sobre GRUB y su uso.
- */usr/share/doc/lilo-<version-number>/* — Este directorio contiene información sobre el uso y la configuración de LILO. En particular, el subdirectorio `doc/` contiene un archivo postscript llamado `User_Guide.ps` que da mucha información. El número *<version-number>* en la ruta a este directorio, corresponde al número de versión del paquete LILO instalado.

##### 4.11.2 Sitios Web útiles

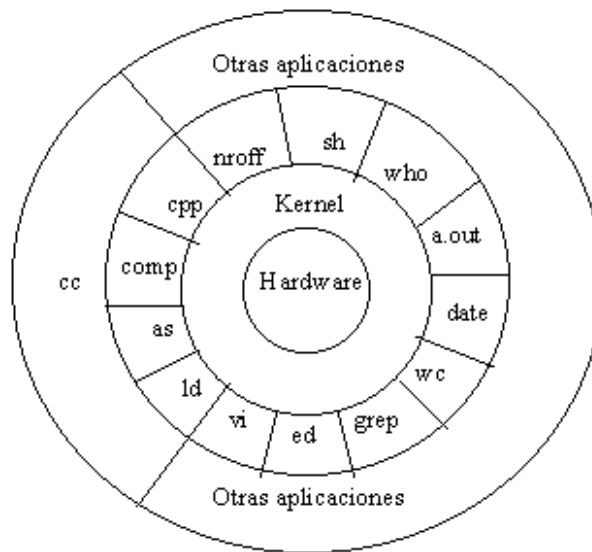
- <http://www.gnu.org/software/grub/> — Página principal del proyecto GNU GRUB. Este sitio contiene información sobre el estado de desarrollo de GRUB y una sección de Preguntas más frecuentes (FAQ).
- <http://www.uruk.org/orig-grub/> — La documentación original GRUB antes de que el proyecto se entregue a la Free Software Foundation para su posterior desarrollo.

## 5. Estructura del sistema

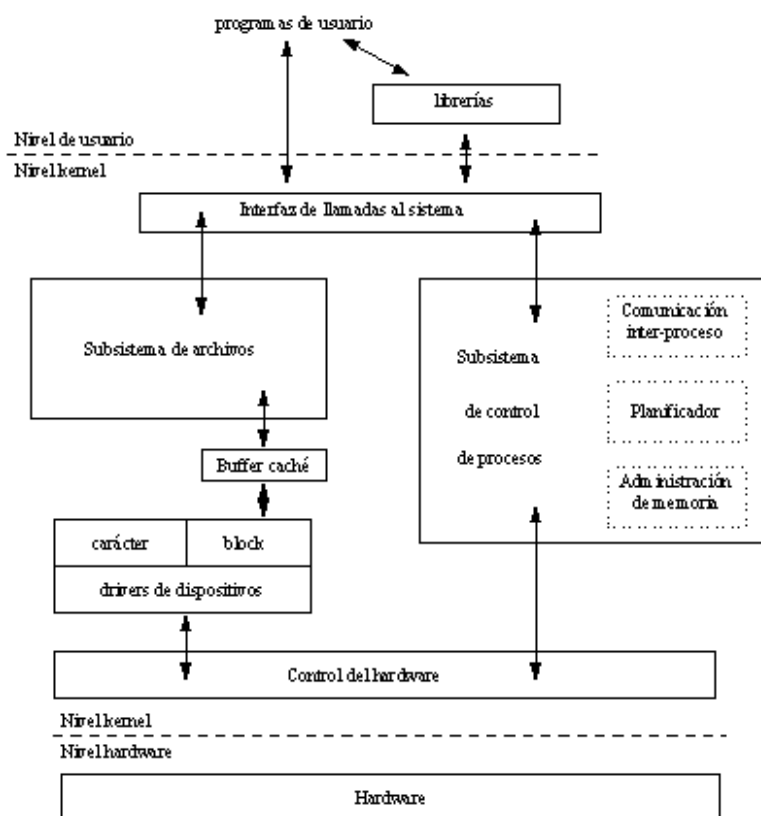
La figura 5.a describe la arquitectura de alto nivel de UNIX. El sistema operativo interactúa directamente con el hardware, suministrando servicios comunes a los programas y aislándolos de la particularización del hardware. Viendo el sistema como un conjunto de capas, el sistema operativo es comúnmente llamado como núcleo del sistema o kernel. Como los programas son independientes del hardware que hay por debajo, es fácil moverlos desde sistemas UNIX que corren en diferentes máquinas si los programas no hacen referencia al hardware subyacente. Por ejemplo, programas que asumen el tamaño de una palabra de memoria será más difícil de mover a otras máquinas que los programas que no lo asumen.

Los programas como el shell y los editores (ed y vi) mostrados en la capa siguiente interactúa con el kernel invocando un conjunto bien definido de llamadas al sistema. Las llamadas al sistema ordenan al kernel realizar varias operaciones para el programa que llama e intercambiar datos entre el kernel y el programa. Varios programas mostrados en la figura 1 están en configuraciones del sistema estándares y son conocidos como comandos, pero los programas de usuario deben estar también en esta capa, indicándose con el nombre a.out, el nombre estándar para los archivos ejecutables producidos por el compilador de C. Otros programas de aplicaciones pueden construirse por encima del nivel bajo de programas, por eso la existencia de la capa más exterior en la figura 1. Por ejemplo, el compilador de C estándar, cc, está en el nivel más exterior de la figura: invoca al pre-procesador de C, compilador, ensamblador y cargador, siendo todos ellos programas del nivel inferior. Aunque la figura muestra una jerarquía a dos niveles de programas de aplicación, los usuarios pueden extender la jerarquía a tantos niveles como sea apropiado. En realidad, el estilo de programación favorecida por UNIX estimula la combinación de programas existentes para realizar una tarea.

Muchos programas y subsistemas de aplicación que proporcionan una visión de alto nivel del sistema tales como el shell, editores, SCCS (Source Code Control System) y los paquetes de documentación, están convirtiéndose gradualmente en sinónimos con el nombre de "Sistema UNIX". Sin embargo, todos ellos usan servicios de menor nivel suministrados finalmente por el kernel, y se aprovechan de estos servicios a través del conjunto de llamadas al sistema. Hay alrededor de 64 llamadas al sistema en System V, de las cuales unas 32 son usadas frecuentemente. Tienen opciones simples que las hacen fáciles de usar pero proveen al usuario de gran poder. El conjunto de llamadas al sistema y los algoritmos internos en los que se implementan forman el cuerpo del kernel. En resumen, el kernel suministra y define los servicios con los que cuentan todas las aplicaciones dLinux.



### 5.a Arquitectura del sistema Linux



### 5. b Bloques funcionales del sistema operativo UNIX.

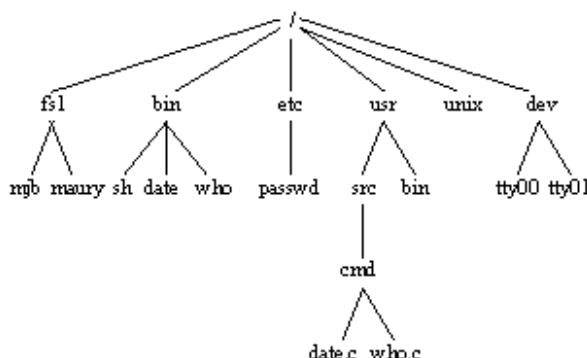
El sistema operativo UNIX se compone de bloques funcionales que se encuentran representados en la figura 5.b y serán comentados en los apartados que siguen.

## 5.1 Estructuras Básicas

### 5.1.1 Organización de la información

El sistema operativo UNIX está diseñado para manejar información contenida normalmente en discos. Para que esta manipulación sea realmente efectiva, es necesario que la información esté organizada de alguna forma. La manera estándar de organizar la información es en archivos. Los archivos son localizados dentro del disco porque son apuntados desde un lugar determinado, a este lugar se le denomina directorio.

Sin embargo en UNIX no se utiliza un único directorio para apuntar a todos los archivos del sistema, sino que se crea una estructura jerárquica de directorios conocida como estructura en árbol. Se muestra un ejemplo de esta estructura jerárquica en la figura 3.



### 5.c Estructura del sistema de archivos.

Aunque en un principio pueda parecer una estructura excesivamente complicada por la amplia ramificación de los directorios, esta organización permite agrupar los archivos de los diferentes usuarios e incluso de las diferentes aplicaciones en directorios separados, con lo que se evita el que se interfieran entre sí. Por otro lado, un mismo usuario puede organizar su propia información separando los archivos en diferentes directorios de acuerdo a su contenido.

### 5.1.2 Concepto de archivo

La estructura fundamental que utiliza Linux para almacenar información es el archivo. Linux mantiene registro de cada uno de los archivos asignándole a cada uno, un número de identificación (i-number) que apunta a la tabla de i-nodos. El usuario, no obstante, indicará cada archivo por medio de su nombre asignado por el propio usuario. Linux mantiene toda la información propia del archivo, excepto su nombre, en



su i-nodo correspondiente. El nombre se encuentra en la entrada correspondiente del directorio al que pertenece.

Todos los archivos Linux son tratados como una simple secuencia de bytes (caracteres), comenzando en el primer byte del archivo y terminando con el último. Un byte en concreto dentro del archivo, es identificado por la posición relativa que ocupe en el archivo. La organización de los archivos en registros o bloques de una longitud fija, corresponde a la forma tradicional de organización de los datos en el disco. Linux protege específicamente al usuario del concepto de registros o bloques. En su lugar, Linux puede permitir que el usuario divida al archivo en registros, de acuerdo a un byte o conjunto de bytes especiales. Los programas son libres para organizar sus archivos, con independencia de la forma en que los datos estén almacenados en el disco.

### 5.1.3 Directorios

Prácticamente para todos los efectos, un directorio se comporta como un archivo, con la característica de que sus registros son de longitud fija. A pesar de lo comentado anteriormente, el contenido de los directorios no apunta directamente a los bloques de datos de los archivos o subdirectorios que dependen de él, sino a unas tablas (i-nodos) separadas de la estructura. Es desde estas tablas desde las que se apunta a los bloques físicos de los archivos dentro del disco. Cada registro en el directorio tiene una longitud de 16 bytes de los que:

- Los dos primeros contienen un apuntador (en binario) a la tabla de los i-nodos (i-number).
- Los catorce restantes contienen el nombre del archivo o subdirectorio (el número bytes por registro depende de la versión de Linux, ya que hay versiones en que la longitud del nombre de un archivo puede llegar a 255 bytes).

Cualquier directorio contiene un mínimo de dos entradas:

- Una referencia a sí mismo (.).
- Una referencia al directorio de que depende o directorio padre (..). Esto sucede para todos salvo en el directorio raíz que, al ser el comienzo de la estructura, apunta esta referencia también sobre sí mismo.

i-NODO	Nombre del archivo
20	.
2	..
45	datos
53	fich
.	.
.	.
.	.

#### 5.d Estructura de un directorio.

### 5.2 Sistema de archivos (File System)

El conjunto formado por la estructura de archivos y directorios y las tablas de i-nodos, se denomina Sistema de Archivos o File System. Cada sistema de archivos está residente en un soporte físico, normalmente disco magnético, pero si el disco es de gran capacidad puede dividirse en varios discos lógicos o "particiones", cada uno de los cuales sería un sistema de archivos independiente. En el caso de que el sistema disponga más de un disco físico, cada uno será forzosamente un file system. Por lo tanto se puede asociar el concepto de sistema de archivos al de disco lógico o partición de disco. Es importante no perder la idea de que, aparte de otros componentes que se verán más adelante, cada sistema de archivos dispondrá de su propia tabla de i-nodos que apuntarán a los bloques de datos de los archivos y directorios contenidos en él.

El sistema de archivos que contiene el directorio raíz (normalmente el primer disco o la primera partición del primer disco en su caso) es el principal. En principio, cuando se carga el sistema operativo, la única parte accesible de la estructura es la contenida en el file system principal. Para poder acceder a la información contenida en los restantes sistemas de operación es necesario realizar una operación denominada "montaje". Esta operación consiste en enlazar cada sistema de archivos con directorios vacíos de la estructura principal. A partir de ese momento, los archivos y directorios de los sistemas de archivos secundarios figurarán dentro de la estructura, colgando del directorio de montaje.

Una vez montado todos los sistemas de archivos, la estructura es única, por lo que el disco o la partición en que esté físicamente un determinado archivo, es transparente para el usuario. La función de montaje y desmontaje de los sistemas de archivo es, normalmente, realizada por el administrador del sistema.

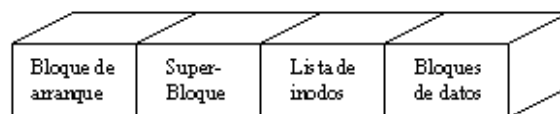
El sistema de archivos, o file system, de UNIX está caracterizado por:

- Una estructura jerárquica.
- Tratamiento consistente de los datos.
- La habilidad de crear y borrar archivos.

- Crecimiento dinámico de archivos.
- La protección de los datos.
- El tratamiento de los dispositivos periféricos como archivos.

Cada sistema de archivos consta fundamentalmente de las siguientes partes

- **Bloque de carga:** Este bloque, que es el primero de cada sistema (bloque cero), está reservado para un programa de carga. El bloque cero no tiene ningún significado en el sistema de archivos. Toda la información del sistema comienza en el bloque uno del dispositivo. Sólo se utiliza en el sistema de archivos principal.
- **Súper bloque:** el súper bloque es el bloque uno del dispositivo. Este bloque contiene información sobre el sistema de archivos, tales como su tamaño en bloques, el nombre del sistema de archivos, número de bloques reservados para i-nodos, la lista de i-nodos libres y el comienzo de la cadena de bloques libres. También contiene el nombre del volumen, momento de la última actualización y tiempo del último backup. Siempre reside en un bloque de 512 bytes.
- **I-nodos:** A continuación del súper bloque están situados los bloques que contienen los i-nodos. El número de bloques de i-nodos varía dependiendo del número de bloques del sistema de archivos. El número de i-nodos está especificado en el súper bloque. Es una tabla que contiene información sobre las características de los archivos. Esencialmente es el bloque de control de los archivos. Hay un i-nodo por cada directorio y archivo del sistema de archivos. El i-nodo contiene una descripción del directorio o archivo, así como el lugar físico que ocupan sus bloques de datos. Los i-nodos sólo apuntan a los archivos o directorios de su mismo sistema de archivos.
- **Bloques de datos:** El resto del espacio del dispositivo lógico consta de bloques de datos. Bloques de datos que contienen los datos actualmente almacenados en los archivos. Algunos bloques de datos sirven como bloques indirectos, conteniendo números de bloques (direcciones) de grandes archivos.



**5.e Estructura de un Sistema de Archivos.**

El sistema de archivos está organizado en una estructura jerárquica. Este sistema jerárquico permite agrupar la información de los usuarios de una forma lógica. Permite manipular eficientemente un grupo de archivos como una sola unidad. No tiene ninguna limitación en su desarrollo, permitiendo un crecimiento dinámico tanto en su número como en su tamaño. Linux trata a los archivos de entrada o de salida del mismo modo que cualquier otro archivo. Los usuarios deberán recorrer el árbol hasta

encontrar el archivo deseado. Dentro de Linux el usuario puede acceder a cuatro clases de archivos:

- Directorios.
- Ordinarios.
- Especiales.
- fifo (Pipes).

Los directorios y los archivos ordinarios han sido ya comentados. Los archivos especiales se verán más adelante.

### **5.3 Manejo de los i-nodos**

Cuando se intenta acceder a la información contenida en un archivo, el sistema accede al directorio al que pertenece y busca su nombre secuencialmente. Una vez encontrado toma el i-number asociado a ese archivo y con él accede a la entrada correspondiente a la tabla de inodos, que contendrá toda la información correspondiente a ese archivo excepto su nombre. Con esta información ya puede acceder físicamente a los bloques de datos dentro del sistema de archivos.

Esta forma de acceder físicamente a los bloques de datos puede parecer más compleja que la utilizada en otros sistemas operativos en los que se accede directamente con la información existente en el directorio, sin necesidad de tablas de i-nodos ni nada por el estilo, sin embargo presenta algunas ventajas. Por ejemplo, si en un directorio se añade una entrada con un nombre de archivo cualquiera, pero con un inodo ya utilizado en otro directorio, se podrá acceder a los mismos bloques de datos desde los dos directorios e incluso con diferentes nombres de archivos. En este caso se dice que existe un enlace entre ellos.

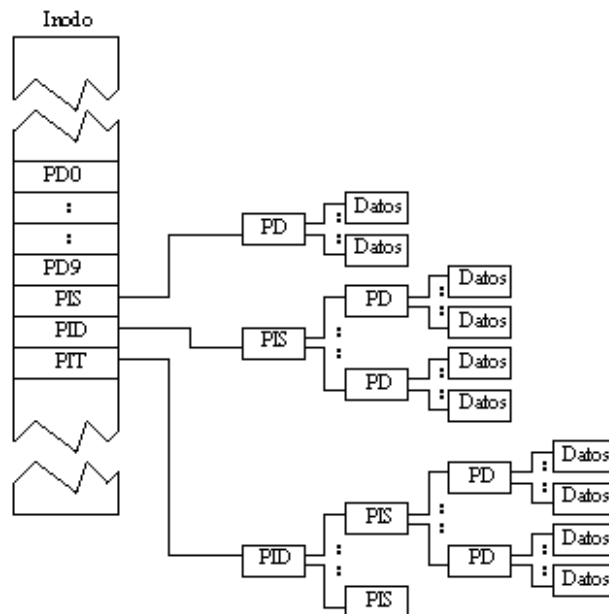
Linux mantiene la siguiente información para cada archivo; en la tabla de i-nodos:

- Localización.
- Tamaño.
- Cuántos enlaces tiene el archivo.
- Propietario.
- Permisos de acceso.
- Identificación de si es o no archivo.
- Fecha de creación.
- Fecha de modificación.
- Fecha de último acceso.

Modo
Cuenta de enlace
ID. de usuario
ID. de grupo
Tamaño del archivo
Direcciones de bloque
Fecha de acceso
Fecha de modificación
Fecha de cambio de inodo

#### 5.f Contenido de un i-nodo.

Además, cada i-nodo contiene diez apuntadores directos con la dirección de los diez primeros bloques de datos, un puntero indirecto simple con la dirección de un bloque que contiene apuntadores directos, que a su vez apuntan a bloques de datos; un puntero indirecto doble que apunta a un bloque con apuntadores indirectos simples, que apuntan a bloques con apuntadores directos y éstos a bloques de datos; y un puntero indirecto triple. En la figura 5.g se ve un ejemplo gráfico de esta estructura.



**Figura 5.g Estructura de los apuntadores.**

#### 5.4 Archivos especiales

Además de los archivos normales de datos, dentro de la estructura de un sistema UNIX existen unos archivos especiales. Los nombres de estos archivos identifican los dispositivos físicos hardware del equipo:

- Discos magnéticos
- Cintas magnéticas
- Terminales
- Líneas de comunicaciones
- Etc.

Cada tipo de dispositivo tiene un controlador responsable de comunicarse con este dispositivo. Dentro del sistema hay una tabla que apunta a los diferentes controladores de los dispositivos. Todos los dispositivos son tratados como archivos. Existen archivos especiales para cada línea de comunicaciones, disco, unidad de cinta magnética, memoria principal, etc. Estos archivos en realidad están vacíos. Tienen por misión asociar entre sí los dispositivos y sus respectivos drivers. El sistema emplea dos números enteros, denominados número principal (14 bits) y número secundario (18 bits) y almacenados en el i-nodo del archivo, para acceder al dispositivo asociado.

El número principal identifica una clase de dispositivo (en realidad identifica al driver de dicha clase como pueden ser terminales, impresoras, discos, etc.) y el número secundario identifica a un elemento de dicha clase (un terminal específico, un disco concreto, ...).

Las ventajas de tratar las unidades de entrada / salida de esta forma son:

- Los archivos y dispositivos son tratados lo más similarmente posible.
- Los nombres de ambos son pasados a los programas de la misma forma.
- Los archivos especiales están sujetos al mismo tipo de protección de acceso que los archivos y directorios ordinarios.



#### 5.h Relación de un archivo especial.

## 6. Comandos Básicos de Usuario

### 6.1 man

Las páginas de man (abreviatura de manual) son documentos online que cubren el uso de las herramientas y sus correspondientes archivos de configuración. El formato del comando man es como sigue:

```
$ man nombre_programa
```

donde nombre\_programa identifica el programa de interés. Por ejemplo:

```
$ man ls
```

Dentro de la lectura acerca de LINUX o fuentes de información relacionadas con LINUX (grupos de noticias), se encuentran referencias a comandos seguidas por números entre paréntesis (por ejemplo, ls (1)). El número representa la sección de las páginas del manual. Cada sección cubre varias áreas de temas, para acomodarse al hecho de que algunas herramientas (tales como printf) son comandos del lenguaje de programación C, así como comandos de la línea de comandos.

Para referirse a una sección del man específica, simplemente especifique el número de sección como primer parámetro y el comando como segundo parámetro. Por ejemplo, para conseguir información de programación en lenguaje C sobre printf, debería escribirse esto:

```
$ man 3 printf
```

#### Secciones de página man

Sección	Tema
1	Herramientas de usuario.
2	Llamadas al sistema.
3	Llamadas a bibliotecas C.
4	Información de controladores de dispositivos.
5	Archivos de configuración.
6	Juegos.
7	Paquetes.
8	Herramientas del sistema.

Por defecto, se imprime primero el número de sección más bajo. Desafortunadamente, esta organización es a veces difícil de usar. Puede resultar útil usar la interfaz gráfica para esta biblioteca de documentación, desarrollada como parte del proyecto GNOME, llamada gnome-help-browser. El icono grande para esta interfaz se encuentra en la barra de herramientas de la parte inferior de la pantalla o en el menú de selección de

la esquina inferior izquierda de su pantalla. Si no existe este icono, siempre puede iniciarse desde la terminal con lo siguiente:

```
$gnome-help-browser
```

## **6.2 pwd (print working directory)**

Función: Imprime la ruta completa del directorio actual de trabajo.

Cuando a un usuario se le da de alta en el sistema, también se le crea un directorio propio, que es al que se acceso directamente al entrar en sesión. Para lograr ver la ruta de acceso a él se teclea el comando y ésta se imprime en pantalla:

```
$pwd  
/home/nancy  
$
```

## **6.3 ls (list)**

### **6.3.1 Función**

Imprime en pantalla los nombres de los archivos contenidos en el directorio especificado.

Si al ejecutar esta orden no se da un nombre de directorio, por default se ven los nombres de los archivos del directorio actual. Si en el directorio no hay archivos creados aún por el usuario, entonces, sólo aparecerá el prompt del sistema. En caso de que hayan archivos en el directorio probablemente se verá algo como esto:

```
$ls  
carri clase nuevo  
$
```

### **6.3.2 Opciones**

-l (large)

Descripción: Da información absoluta acerca de los archivos del directorio. Si se desea tener información detallada acerca de un archivo, esta opción es muy útil para lograrla. Al teclear el comando se ve lo siguiente:

```
$ls -l  
total 3  
-rw-r--r-- 1 nancy ptc 163 Mar 4 11:00 carri  
-rx-r--r-- 1 nancy ptc 379 Mar 4 10:24 clase  
drwxr-xr-x 2 nancy ptc 64 Mar 4 10:56 nuevo  
$
```



-a (all)

Descripción: Al utilizar esta opción de ls se le pide al sistema que muestre todo tipo de archivos, incluso los ocultos. Si se teclea el comando con esta opción combinada con la opción -l en el directorio anterior aparece esto:

```
$ ls -la
total 29
drwxr-xr-x 7 nancy ptc 1024 Mar 6 14:29
drwxr-xr-x 24 root ptc 1024 Feb 23 11:32
-rw-r--r-- 1 nancy ptc 163 Mar 4 11:00 carri
-rw-r--r-- 1 nancy ptc 379 Mar 4 10:24 clase
drwxr-xr-x 2 nancy ptc 64 Mar 4 10:56 nuevo
$
```

-x (across)

Descripción: Lista los nombres de archivo por línea en orden alfabético. Si el directorio contiene un número considerable de archivos, esta opción ayuda a ahorrar espacio, colocando un nombre tras de otro. Probemos esta opción en el directorio padre del actual:

```
$ls -x
antonio argos david doom falcon firebird gork gus
invent jess morfeo nancy navajo pcs ptcweb raq
rody seiya sergio skid steeleryumbe4
$
```

-c (change)

Descripción : Coloca los nombres de archivos del directorio especificado por orden alfabético en forma de columnas. Ejecutando la opción -c con el directorio anterior se obtiene el siguiente resultado:

```
$ls -C /users/ptc
antonio doom gork jess navajo raq sergio
argos falcon gus morfeo pcs rody skid
david firebird invent nancy ptcweb seiya steerer
$
```

-p

Descripción: Indica si un nombre de archivo es directorio o no. Esta opción es muy útil si no se quiere el listado largo del directorio si no sólo saber los subdirectorios que existen en él. Al ejecutar la opción se indica que es un subdirectorio señalándolo con un "/" al final del nombre del archivo:

```
$ls -p
carri clase nuevo/
$
```

-F

Descripción: Indica qué archivos son subdirectorios y también cuáles son ejecutables. Al usar esta opción, los archivos ejecutables y los directorios se señalan con un \* y con un /, respectivamente. Al igual que en la opción anterior, estos símbolos no forman parte del nombre del archivo.

-t (time)

Descripción: Ordena a los nombres de archivos por su fecha de última modificación. Tal vez al usar esta opción sea conveniente combinarla con -l. Al usarse en el directorio que se está trabajando muestra algo como esto:

```
$ls -lt
```

```
-rw-r--r-- 1 nancy ptc 163 Mar4 11:00 carri  
drwxr-xr-x 2 nancy ptc 64 Mar4 10:56 nuevo  
-rw-r--r-- 1 nancy ptc 379 Mar4 10:24 clase  
$
```

-u

Descripción: Al usar esta opción los archivos se listen por la última fecha en la que se accedió a ellos. Al combinar esta opción con -l se ve la siguiente pantalla:

```
$ls -lt
```

```
-rw-r--r-- 1 nancy ptc 163 Mar 5 15:36 carri  
-rw-r--r-- 1 nancy ptc 379 Mar 4 10:24 clase  
drwxr-xr-x 2 nancy ptc 64 Mar 7 13:06 nuevo  
$
```

-r

Descripción: Entrega el listado de archivos en orden inverso al normal. Cuando se utilice una opción como -t o -u, el orden en que aparecen los nombres de archivos es por la última fecha de modificación o acceso. Si se quisiera ordenarlos por primer modificación o acceso debe usarse esta opción. Este es el ejemplo:

```
$ls -ltr
```

```
-rwxr--r-- 1 nancy ptc 379 Mar 4 10:24 clase  
drwxr-xr-x 2 nancy ptc 64 Mar 4 10:56 nuevo  
-rw-r--r-- 1 nancy ptc 163 Mar 4 11:00 carri  
$
```

### 6.3.3 Campos del ls

-rw-r--r--	1	nancy	ptc	163	Mar 4 11:00	carri	
------------	---	-------	-----	-----	-------------	-------	--

Nombre del Archivo  
Fecha y Hora de la última modificación  
Tamaño del archivo  
Nombre del Grupo  
Nombre del dueño  
Número de ligas  
Permisos del archivo

En el primer campo se pueden ver los permisos que tiene el archivo para el dueño y los demás usuarios del sistema. En su primer columna se ve el si se trata de directorio o archivo. Sobre los permisos se hablará con más detalle cuando se hable de otro comando. En el segundo campo podemos ver el número de ligas que existen en el archivo en todo el sistema. El siguiente campo muestra el nombre del dueño del archivo. En este caso es nancy. El cuarto campo indica cuál es el grupo al que pertenece el usuario. En el ejemplo vemos que nancy pertenece al grupo ptc.

El quinto campo muestra el tamaño del archivo en bytes. Si se tratara de un archivo de texto, este número indica el número de caracteres que contiene. En el penúltimo campo se puede ver la fecha y hora en la cual se le hizo la última modificación al archivo cuando se trata de un archivo reciente. Si el archivo no fue creado en el año corriente, entonces en lugar de lo anterior se puede ver el año en el que se creó. En el último campo se indica el nombre del archivo.

### 6.4 cd (change directory)

Función: Sitúa al usuario en otro directorio de trabajo.

```
$cd /etc
$pwd
/etc
$
```

Es conveniente que cada vez que se haga un cd se verifique el directorio en el que se está para evitar problemas que podrían ser graves, como el borrar archivos que no se querían borrar. Cuando se encuentra en un directorio que no es el propio se puede regresar a él fácilmente tecleando sólo cd. Esta orden lo llevará ahí automáticamente:

```
$pwd
/etc
$cd
```

```
$pwd  
/users/ptc/nancy
```

## 6.5 Rutas absolutas y rutas relativas

Existen dos formas de acceder a un archivo o a un directorio, la primera es escribiendo el nombre de la ruta absoluta y la segunda es utilizando una ruta relativa. Al mencionar ruta absoluta se quiere decir que hay que especificar el nombre de los directorios y subdirectorios por los que hay que pasar para llegar a archivo especificado comenzando desde la raíz del sistema. La ruta relativa se refiere a cómo llegar a ese archivo desde el directorio en que se está trabajando. El siguiente ejemplo muestra cómo llegar al subdirectorio /users/local/bin haciendo uso de su ruta absoluta:

```
$pwd  
/users/ptc/nancy  
$cd /users/local/bin  
$pwd  
/users/local/bin  
$
```

En el próximo ejemplo se muestra cómo hacer uso de la ruta relativa para llegar a ese directorio:

```
$cd  
$pwd  
$/users/ptc/nancy  
$cd ../../local/bin  
$pwd  
/users/local/bin  
$
```

En el segundo ejemplo se ve cómo se hace referencia al directorio padre del actual (ptc), luego al padre del padre (users), que es donde está el subdirectorio local, y, por último se da el nombre del directorio al que se quería llegar. En este ejemplo tal vez parezca que es incómodo usar las rutas relativas, pero no sería así si al subdirectorio que se quisiera llegar estuviera en el directorio padre, por ejemplo.

## 6.6 cat

Función: Muestra el contenido de un archivo en pantalla.

Para que el resultado de la ejecución de este comando tenga significado, debe utilizarse sobre un archivo de texto. El siguiente ejemplo muestra el uso de cat sobre un archivo ya creado en el directorio /etc:

```
$cat /etc/termcap  
##### TERMINAL TYPE DESCRIPTIONS SOURCE FILE
```

```
#
# Version 11.0.1
# $Date: 2000/03/02 15:51:11 $
# termcap syntax
#
# Eric S. Raymond(current maintainer)
# John Kunze, Berkeley
# Craig Leres, Berkeley
#
...
```

Esta tabla muestra los valores de termcap todos los caracteres en notación octal. Cuando se teclea mal el nombre de un archivo o éste no existe, en la pantalla se ve un mensaje que dice:

```
$ cat cari
cat: cannot open cari: No such file or directory
$
```

## 6.7 Redireccionamiento

El comando cat no sólo puede mostrar el contenido de un archivo a la pantalla, sino que también lo puede mandar a otro archivo para añadirsele, o para sobre escribir el resultado en él. A este proceso se le llama redireccionamiento. Para redireccionar la salida de cat de un archivo a otro podemos hacer lo siguiente:

```
$ cat /usr/pub/ascii > ascii
$ls
carri clase nuevo ascii
$
```

ls se usó sólo para verificar la ejecución del comando anterior. En el ejemplo se puede ver, cat creó el archivo ascii, que no existía. Si el archivo hubiera existido, su contenido anterior habría desaparecido y la salida de cat se le habría copiado. Como LINUX maneja a los dispositivos como archivos, también se puede direccionar un resultado a una terminal:

```
$ tty
/dev/ttyp2
$cat /etc/termcap > /dev/ttyp2
```

```
##### TERMINAL TYPE DESCRIPTIONS SOURCE FILE
#
# Version 11.0.1
# $Date: 2000/03/02 15:51:11 $
# termcap syntax
#
```

```
# Eric S. Raymond(current maintainer)
# John Kunze, Berkeley
# Craig Leres, Berkeley
#
...
$
```

## 6.8 Metacaracteres

Antes de continuar es conveniente describir una de las características más potentes del Sistema V de LINUX, que es su capacidad de utilizar una notación abreviada para hacer referencia a conjuntos de caracteres. Esto se logra utilizando un conjunto de caracteres especiales llamados comodines o metacaracteres. Estos se utilizan para hacer una correspondencia con nombres de archivos o con partes de nombres de ellos. Su equivalencia es la que sigue:

- \*: Representa a toda una cadena de caracteres o a la cadena nula.
- ?: Representa a cualquier carácter simple.
- [abc]: La correspondencia es cualquier carácter encerrado entre los corchetes, es decir, una a, una b o una c, en este caso. Si las letras entre los corchetes se separan por un -, la correspondencia será cualquier carácter entre ese rango especificado. Por ejemplo, [0-9] corresponderá a cualquier carácter entre 0 y 9, inclusive.
- ^: Representa el inicio de línea.
- \$: Representa el fin de línea.
- !: Representa la negación.

```
$ ls -l /etc/x*
-rwxr--r-- 1 root sys 2729 Mar 07 1993 /etc/xd daemon.rc
-rwx----- 1 bin bin 2481 Mar 15 1993 /etc/xinstall
-rwx--x--x 2 bin bin 25716 Mar 10 1993 /etc/xnstatus
-r-xr-xr-x 1 bin bin 10926 Mar 12 1993 /etc/xntpd
-r-xr-xr-x 1 bin bin 98518 Mar 12 1993 /etc/xntpde
-r-xr-xr-x 1 bin bin 8618 Mar 12 1993 /etc/xntpres
-rw-r--r-- 1 root other 823 Mar 18 11:53 /etc/xtab
$
```

Este ejemplo muestra a todos los archivos de /etc cuyos nombres comienzan con la letra d. Si por ejemplo se deseara conocer los archivos de /etc que tienen una d al principio y una s al final usaríamos la siguiente orden

```
$ ls -l /etc/d*s
-rw-rw---- 1 bin terminal 0 Mar 15 1993 /etc/dialups
$
```

Si se quisiera listar a todos los archivos del directorio /etc simplemente hay que teclear `ls *y` ya está.

Para el uso del carácter ? supongamos que necesitamos conocer los archivos del /etc que comienzan con a y les sigue cualquier carácter y después una t con cualquier otra cadena, entonces se teclearíamos:

```
$ ls -l /etc/a?k*
-rwx----- 2 bin bin 2966 Mar 15 1993 /etc/asktime
-rwx----- 2 bin bin 2966 Mar 15 1993 /etc/asktimerc
$
```

Usar los corchetes tiene mucha ventaja, ya que con ellos podemos manejar rangos o selecciones. Por ejemplo, si queremos conocer el nombre de archivos y directorios que comienzan con las letras t, u, v, w, x, y o z de ese mismo directorio, se tendría que teclear:

```
$ ls -dx /etc/[qwertyui]*
/etc/esd.conf          /etc/exports          /etc/identd.conf
/etc/imrc              /etc/info-dir          /etc/init.d
/etc/initlog.conf      /etc/inittab           /etc/inputrc
/etc/ioctl.save        /etc/isapnp.gone       /etc/isdn
/etc/issue             /etc/issue.net         /etc/rc
/etc/rc.d              /etc/rc.local          /etc/rc.sysinit
/etc/rc0.d             /etc/rc1.d             /etc/rc2.d
/etc/rc3.d             /etc/rc4.d             /etc/rc5.d
/etc/rc6.d             /etc/redhat-release    /etc/resolv.conf
/etc/rmt               /etc/rpc               /etc/rpm
/etc/termcap           /etc/updatedb.conf     /etc/yp.conf
/etc/ypserv.conf
$
```

Esta misma tarea se hubiera podido hacer con estas otras ordenes:

```
$ ls -dx /etc/[q-z]*
/etc/rc                /etc/rc.d              /etc/rc.local          /etc/rc.sysinit
/etc/rc0.d             /etc/rc1.d             /etc/rc2.d             /etc/rc3.d
/etc/rc4.d             /etc/rc5.d             /etc/rc6.d             /etc/redhat-release
/etc/resolv.conf       /etc/rmt               /etc/rpc               /etc/rpm
/etc/samba             /etc/sane.d            /etc/screenrc          /etc/securetty
/etc/security          /etc/sendmail.cf       /etc/services          /etc/shadow
/etc/shadow            /etc/shells            /etc/skel              /etc/smrsh
/etc/sound             /etc/ssh               /etc/sysconfig         /etc/sysctl.conf
```

```

/etc/syslog.conf      /etc/termcap      /etc/updatedb.conf  /etc/xinetd.conf
/etc/xinetd.d        /etc/yp.conf      /etc/ypserv.conf
$ ls -dx /etc/[!a-s]*
/etc/CORBA           /etc/DIR_COLORS   /etc/Mutttrc        /etc/X11
/etc/termcap         /etc/updatedb.conf /etc/xinetd.conf     /etc/xinetd.d
/etc/yp.conf         /etc/ypserv.conf
$

```

Como se puede ver, la última orden selecciona no sólo a los archivos o directorios que empiezan con las minúsculas requeridas, sino también los que comienzan con esas letras, pero mayúsculas. El carácter ! dentro de los corchetes, como se puede ver, invierte la correspondencia de los caracteres.

## 6.9 Combinación de archivos

Otra utilidad del comando cat es que tiene una forma muy fácil de concatenar archivos. Por ejemplo, si se quisiera que en un archivo se copiara el texto de otros dos, se tendría que teclear algo semejante a esto:

```

$cat /usr/pub/ascii /usr/pub/eqnchar >> ascii
$

```

Esta línea copiará el archivo eqnchar del directorio /etc al final del archivo ascii, situado en el directorio actual.

## 6.10 Creación de archivos

Una forma muy fácil de crear archivos de texto es utilizando el comando cat. La forma de hacerlo es la siguiente:

```

$ cat > nuevo
Este es un archivo nuevo.
^D
$

```

cat toma del teclado, que es la entrada por default en el redireccionamiento, lo que se escriba en él y lo manda al archivo nuevo. Para guardar el archivo sólo se oprimen las teclas Ctrl-D en la última línea y el archivo se crea. Para agregar texto al final del archivo nuevo se teclearía:

```

$cat >> nuevo
Agrego más texto al archivo nuevo.
^D

```

Al teclear una línea en el archivo y pulsar enter, esa línea ya no se podrá modificar. Sólo se podrán borrar caracteres en la línea que se este escribiendo.



## 6.11 file

Descripción: Determina de que tipo es un archivo. Cuando se utilizan los comandos para ver el contenido de un archivo, si el escogido no es uno de texto, se verá algo inentendible en la pantalla. Para evitar esto se puede usar el comando file, que dice de qué tipo es un archivo. Si el archivo nombrado no existe o no se tienen permisos de lectura sobre el aparecerá un mensaje de error en la pantalla. En el siguiente ejemplo se aclara su función:

```
$file nuevo clase
nuevo: directory
clase: ascii text
$
```

## 6.12 mkdir (make dir)

Función: Permite crear contenedores de archivos (directorios). Cuando existe un usuario en un sistema LINUX a este se le da un directorio de trabajo. Cada usuario es responsable del contenido de su directorio y en él puede crear tantos archivos y subdirectorios como necesite. El comando indicado para crear directorios es el mkdir. En el próximo ejemplo se ve el uso de este comando:

```
$mkdir programs textos
$cd programs
$mkdir c pascal
$cd
$ls -l
```

Este proceso pudo hacerse también de la siguiente forma:

```
$mkdir programs textos programs/c programs/pascal
$ls -l
total 5
drwxr-xr-x  4 kyo      ptc  64 Mar 18 15:35  programs
drwxr-xr-x  2 kyo      ptc  32 Mar 18 15:35  textos
$
```

Si antes de crear el directorio programs se hubiera intentado crear los subdirectorios c y pascal, el sistema hubiera mandado un mensaje como este:

```
mkdir: cannot access programs/
```

Para evitar el problema anterior se puede utilizar la bandera -p, que indica al comando que es necesario generar aquellos directorios dentro del árbol que no existan aún. Debe tenerse cuidado al crearse directorios de estar en el directorio casa para evitar problemas con los permisos y confusiones.

### 6.13 cp (copy)

Función: Crea una copia del archivo indicado.

*-f Fuerza el movimiento. No pide verificación.*

*-i Copia interactiva. Antes de copiar cada archivo, verifica con el usuario.*

*-r Realiza un procedimiento recursivo, copiando la estructura con subdirectorios incluidos.*

Esta es una orden simple de ejecutar, por ejemplo, si se desea hacer una copia del archivo mensaje del directorio /user/ptc/kyo a uno de los subdirectorios que se acaba de crear con esa misma ruta se tendría que hacer lo siguiente:

```
$pwd
$/users/ptc/kyo
$cp hola textos
$
```

Y si se desea copiar ese mismo archivo con otro nombre en el mismo subdirectorio se haría esto:

```
$cp hola textos/mensaje
$ls-l textos
-rw-r--r-- 1 kyo ptc 6 Mar 18 15:41 hola
-rw-r--r-- 1 kyo ptc 6 Mar 18 15:41 mensaje
$
```

También podrían copiarse más archivos a un mismo directorio en una sola línea de comando:

```
$cp /usr/include/stdio.h /usr/include/math.h programs/c
$
```

Cuando se omite un nuevo nombre en el último argumento que es a donde se copiarán los archivos, éstos se copiarán con el que tenían:

```
$ls -l programs/c
-rw-r--r-- 1 kyo ptc 6685 Mar 18 15:47 math.h
-rw-r--r-- 1 kyo ptc 11700 Mar 18 15:47 stdio.h
$
```

Cuando se copia un archivo también se copian los permisos que tenía el de origen.

## 6.14 mv (move)

Función: Cambia de nombre o de directorio a un archivo. Este comando puede cambiar de nombre a un archivo, igual que cp, con la diferencia de que el archivo fuente será borrado. También puede cambiar de directorio a un archivo con la opción de conservar el nombre o de cambiarlo.

-f Fuerza el movimiento.

-i Movimiento interactivo.

En los siguientes ejemplos se muestra su uso:

```
$mv hola mensaje
$ls
$
```

En las líneas anteriores se cambió el nombre del archivo hola por el de mensaje. Como se ve hola desapareció.

```
$ mv mensaje textos/hola2
$ls textos hola hola2 mensaje
$
```

En la acción anterior se cambió de directorio y de nombre al archivo mensaje. Al igual que con el comando cp, la orden mv puede mover varios archivos a un mismo directorio en la misma línea de comando. El proceso es análogo al que se vio en cp. Es necesario decir sobre mv que si el nombre del archivo destino existe y tiene permisos de escritura, entonces será borrado sin preguntar y su contenido anterior se perderá. Si no, el sistema preguntara si se quiere seguir con el proceso.

## 6.15 rm (remove)

Descripción: Borra los archivos especificados. El uso de este comando es sencillo, sólo se da el nombre del archivo o archivos que se quieren borrar, delante de la orden y ya está, los archivos son suprimidos. Una cosa muy importante al usar este comando es saber con certeza el directorio actual de trabajo ya que si se da una orden de borrado de todos los archivos que se encuentren y no se está en el directorio en el que se pensaba, se pueden tener sorpresas desagradables. El siguiente ejemplo ilustra el uso de esta orden:

```
$pwd
/users/ptc/kyo
$cd textos
$ls
```

```
hole
hola2
mensaje
$rm *
$ls
$
```

el comando ha borrado todos los archivos de este subdirectorio, como se puede ver.

-i (interactive)

Función: Antes de borrar, el sistema pregunta para confirmar la acción. Si los archivos del directorio textos no se hubieran borrado y se utilizara ahora rm con la opción -i, se vería lo siguiente:

```
$cd
$rm -i textos/hola textos/hola2 textos/mensaje
textos/hola: ? n
textos/hola2: ? y
textos/mensaje: ? y
$
```

Lo que se obtiene como resultado de las líneas anteriores es que el archivo hole no se borra, mientras que el archivo hola2 y mensaje son suprimidos.

-f (forzar)

Función: Forza la eliminación de archivos, sin verificar aun cuando estos no tengan permisos de escritura.

-r (recursive)

Función: Esta opción busca hacia abajo en el árbol del directorio y va borrando archivos y subdirectorios desde el directorio actual en adelante. Al utilizar esta opción se debe tener cuidado para no borrar algo que no se deseaba suprimir.

### **6.16 rmdir (remove dir)**

Descripción: Con este comando se logra el borrado de los directorios especificados. Para que un directorio pueda ser borrado con este comando es indispensable que esté vacío, si no la operación no podrá ejecutarse. Por ejemplo, si se quisiera borrar el directorio programs/c sucedería lo siguiente:

```
$pwd
/users/ptc/kyo
$rmdir programs/c
rmdir: programs/c: Directory not empty
$
```

Para lograr lo que se quería en las líneas anteriores se debe hacer lo siguiente:

```
$cd programs/c
$rm *
$cd ..
$rm -r programs/c
$ls
pascal
$
```

Si se quiere borrar un archivo que no existe, se mandará un mensaje que dice:

```
$rm -r uno
rm: uno: Directory does not exist
$
```

### 6.17 Permisos de un archivo o directorio.

El sistema LINUX da una forma de permitir o restringir accesos a una archivo o directorio por parte de un usuario. Estos permisos sólo pueden ser dados por el creador del archivo o por el administrador del sistema. Para ver los permisos que tiene un archivo se utilice el comando `ls -l`, como ya se vio antes.

Los permisos de un archivo son de tres tipos y se les pueden otorgar a tres clases distintas de usuarios. Los tipos son de lectura, de escritura y de ejecución. Se pueden otorgar al propietario (quien creó el archivo), al grupo (usuarios con el mismo directorio padre del propio) y a otros (usuarios de otros grupos).

De lectura	De escritura	De ejecución
r	w	x

El permiso de lectura se refiere a que un usuario tiene acceso al archivo o directorio para ver su contenido. El de escritura sirve para poder modificar un archivo y en el caso de directorios para crear o borrar archivos en el caso que éstos tengan permisos de escritura. El de ejecución es para que se pueda usar el archivo como comando del sistema, y en el caso de directorios para poder copiar archivos de ese directorio en caso de que ellos tengan permiso de lectura.

Si un archivo carece de algún permiso, esto se note al listarlo con la opción `-l`. Si en lugar de la letra que indica el permiso aparece un `-`, esto significa que ese permiso está restringido para el bloque en el que aparece (para el usuario, para el grupo o para otros). Como ya se dijo, los permisos de un archivo sólo se pueden borrar o

modificar por su creador. Para el caso de cambiar permisos a un archivo se usa lo siguiente:

### 6.17.1 chmod (change mode)

Descripción: Sirve para modificar los permisos de un archivo o directorio.

Este comando acepta el cambio de permisos de dos formas, como un valor dado en un número octal o como lo que se denomina como modo simbólico. El comando cambia los permisos tecleando:

chmod permisos archivo

Un modo fácil de manejar los permisos en forma octal es ayudándose con el siguiente esquema:

r	w	x	r	w	x	r	w	x
400	200	100	40	20	10	4	2	1

Así, si se desea dar permiso de sólo lectura, a los tres bloques se suman los valores de dichos permisos:

$$400+40+4=444$$

por lo que el comando para cambiar a esos permisos a todos los archivos del directorio actual será:

```
$chmod 444 *  
$
```

Si se le quisiera cambiar los permisos a un archivo a lectura, escritura y ejecución a un usuario, lectura al grupo y lectura a otros se sumarían 744 y se teclearía:

```
$chmod 744 archivo  
$
```

El otro modo de otorgar los permisos es de manera simbólica y su notación es así:

- u -- (user) asigna permisos solo al propietario
- g -- (group) asigna permisos al grupo
- o -- (other) asigna permisos a otros
- a -- (all) asigna permisos a todos (usuario, grupo y otros)

- = asigna un permiso
- + agrega un permiso
- quita un permiso

Si por ejemplo, le se quisiera otorgar permiso de lectura a todos y además de escritura y ejecución al dueño de los directorios programs y textos, se tendría que teclear:

```
$chmod a=r,u+w,u+x programs textos
$ls -ld programs textos
drwxr--r-- 4 kyo ptc 64 Mar 18 15:35 programs
drwxr--r-- 2 kyo ptc 80 Mar 18 18:46 textos
$
```

Otra forma de hacer lo anterior hubiese sido tecleando estas líneas:

```
$chmod go=r,u+rwX programs textos
$
```

Nótese que para asignar varias órdenes de cambio a un archivo o archivos, se separan mediante una coma.

### 6.17.2 Permisos iniciales

Cuando un archivo es creado, se supone que tiene asignados permisos por default, dados por el sistema. Sin embargo, los permisos de un archivo al momento de su creación pueden modificarse, creando una nueva máscara de usuario.

#### **umask (user mask)**

Descripción: Sirve para cambiar los permiso iniciales de un archivo o directorio. Si se deseara ver los permisos iniciales actuales sólo se teclearía umask, sin argumentos :

```
$umask
000
$
```

Esto significa que los archivos al crearse tienen permisos rw-rw-rw (666) y que los directorios tendrán rwxrwxrwx(777). Si se quisiera cambiar los permisos iniciales a permisos de lectura y escritura para el usuario y sólo de lectura para los demás (644) tendría que hacer primero:

```
666 permisos actuales
644 permisos requeridos
022 modo final y después teclear el comando:
```

```
$umask 022
$
```

Con esto los nuevos archivos a crear adquirirán permisos rw-r--r-- (644) y los nuevos directorios rwxr-xr-x (755) La nueva máscara creada en una sesión será substituida por la que el sistema maneja por defecto al apagar la máquina, así que si se quiere

tener la máscara deseada cada vez que se entre al sistema, se deberá modificar el perfil de sesión.

### 6.17.3 chown (change owner)

Descripción: Permite hacer el cambio de propietario para un archivo o directorio. El proceso para poder hacer el cambio de dueño sólo se puede llevar a cabo por el propietario original del archivo, o por el administrador del sistema. Veamos un ejemplo de chown cambiar de dueño un directorio:

```
$mkdir pruebas
$ls -ld pruebas
drwxr-xr-x 2 kyo ptc 32 Mar 18 15:45 pruebas
$chown nancy /users/ptc/kyo/pruebas
$ls -ld pruebas
drwxr-xr-x 2 kyo ptc 32 Mar 18 15:45 pruebas
$
```

### 6.18 ln

Descripción: El comando ln le permite establecer enlaces duros y enlaces suaves. El formato general de ln es el siguiente:

```
$ln archivo_original archivo_nuevo
```

Aunque ln tiene muchas opciones, nunca deberá usarse la mayoría de ellas. La opción mas utilizada es :

**-s crea un enlace simbólico o suave en lugar de un enlace duro.**

```
$ln -s archivo_original enlace_nuevo
```

### 6.19 more

El comando more toma un flujo de entrada y lo muestra en una pantalla a la vez. El flujo de entrada puede venir de la entrada estándar o de un parámetro de la línea de comandos.

```
$ more /etc/passwd
```

### 6.20 du

Con frecuencia se necesita determinar dónde y cuánto espacio de disco está ocupado, el comando du permite determinar la utilización de disco en base a los directorios. Éstas son algunas de las opciones disponibles:

- a Muestra para cada archivo su tamaño.
- b Muestra el tamaño en bytes.



- c Muestra el total gastado al final de la ejecución
- h Imprime el tamaño de formatos legibles para la personas.
- k Imprime al tamaño en kilobytes en lugar de en bloques
- s Sumario. Imprime sólo una salida para cada argumento.

### 6.21 whereis

La herramienta whereis busca en el path y muestra el nombre del programa y su directorio absoluto, el archivo fuente(si está disponible) y la página del man para ese comando. Para encontrar la posición del programa, fuente y página del man del comando ls:

\$whereis ls

### 6.22 df

Descripción: El programa df muestra la cantidad de espacio libre, partición. Las particiones deben estar montadas para conseguir esta información. La información NFS se puede también conseguir se esta forma. Se muestra a continuación algunos parámetros de df:

- h Genera la cantidad de espacio libre en números legibles en lugar de en bloques libres.
- l Lista sólo los sistemas de archivos montados localmente. No muestra información por red.

### 6.23 who

Descripción: En los sistemas que permiten al usuario entrar en otras máquinas o en servidores, puede quererse saber quién está conectado. Puede generarse un informe usando el comando who.

- u Opción por defecto que informa sobre los usuarios actuales del sistema.
- T Informa sobre la posibilidad de enviar mensajes mediante write a los distintos usuarios que estén en el sistema.
- q Cambia el formato de salida de la información sobre usuarios a un formato más rápido.

### 6.24 Comandos emuladores de DOS (mcd, mcopy, mdir, mdel)

Utilidades para el manejo de un sistema de archivos ms-dos.

Descripción: Este grupo de comandos se encuadran dentro de las conocidas mtools, que no son sino un conjunto de herramientas que facilitan el manejo de un sistema de archivos tipo MS-Dos dentro de Linux. sin tener que montarlo expresamente. La utilidad de estos comandos radica sobre 1.0(k) en el manejo de disquetes formateados con MS-DOS, a los que se puede acceder con estos comandos sin tener que montar la unidad.

Ejemplos:

Copia todos los archivos \*.txt del directorio actual Linux, a un disquete formateado con ms-dos.

\$mcopy \*.txt a:

Borra todos los archivos \*.txt del directorio raíz del disquete ubicado en la unidad A.

\$mdel a:\\*.txt .

## 6.25 diff

Descripción: Básicamente, lo que realiza el comando diff es encontrar las diferencias entre dos archivos de texto. Sintaxis:

diff [opciones] archivo\_a archivo\_b

Opciones:

-a	Compara los archivos como de texto incluso si no parecen ser (le texto.
-b	Ignora los espacios en blanco en la comparación.
-d	Cambia el algoritmo para intentar encontrar un conjunto de cambios más pequeño.
-e	Genera un script para el editor ed. con los cambios necesarios para igualar los archivos.
-r	Opción recursiva para directorios.
archivo_a	Este es el archivo que se quiere que se compare. diff nos devuelve la información con las operaciones a realizar para que este archivo coincida con archivo_b.
archivo_b	Archivo maestro que se utiliza en la comparación para que diff nos indique que se ha de hacer en archivo a para que sea igual a éste.

Ejemplo:

\$diff tabla\_a.txt tabla\_b.txt

Compara los dos archivos indicados buscando las diferencias del primero con respecto al segundo.

## 6.26 env

Descripción: Permite cambiar el ambiente de ejecución, variables de ambiente, para la ejecución de un comando o programa sin variar el ambiente actual. Los cambios de las variables de ambiente sólo afectarán al comando que se quiere lanzar con env. Si se ejecuta sin algún parámetro sirve para ver el ambiente actual. Sintaxis:

env [opciones] comando

Opciones:

-	Considera el ambiente formado sólo por los valores que se especifican a continuación en el comando env.
-u nombre	Elimina la variable indicada con nombre.
-i	Inicializa el ambiente como vacío.
nombre= valor	Establece las variables de ambiente y sus valores para la ejecución del comando. Este parámetro se puede repetir tantas veces como se quiera.
comando	Comando o archivo con código ejecutable que se quiere ejecutar con el ambiente cambiado.

Ejemplos:

\$env HOME=/home/pepe cd

Ejecuta el comando cd con el ambiente cambiado, como se ha cambiado la variable de ambiente HOME el resultado cd está directamente afectado por ese cambio.

\$env

Muestra los valores del ambiente actual.

## 6.27 time

Descripción: Este comando facilita la información de tiempo que consume un comando. La información que facilita es de tres tipos: tiempo real que transcurre desde que se lanza el comando hasta que termina, tiempo de usuario que es el tiempo que realmente ha utilizado de CPU y, por último, tiempo de sistema que es el tiempo consumido en peticiones al sistema. Sintaxis:

\$ time comando

Ejemplo:

```
$ time ls -s /
```

Medirá el tiempo que tarde en ejecutarse un `ls -l` de todo el sistema de archivos.

### 6.28 clear

Descripción: Borra la pantalla completamente del terminal en el que se esté trabajando.

```
$ clear
```

### 6.29 date

Descripción: Este comando permite ver la hora del sistema y la fecha. Si se es el usuario root además permite cambiar día hora y fecha. El formato de salida tiene multitud de opciones y se recomienda consultar la página man correspondiente para verlas todas. Sintaxis:

```
$ date [MMDDhhmm] [CC][AA] [.ss] [+formato]
```

Siendo:

MM	El mes que vade 01 a 12.
DD	El día del mes que va de 01 a 31.
hh	La hora del día de 00 a 23.
mm	El minuto de la hora que vade 00 a 59.
CC	Es opcional y son los dos primeros dígitos del año.
AA	Opcional y son los dos últimos dígitos del año.
.ss	Se puede indicar opcionalmente los segundos.

Ejemplo:

```
$date 01081430 20 00
```

Representa la hora 4:30 del día 1 de agosto de 2000.

## 7. Redireccionamiento, filtro e interconexión de comandos.

### 7.1 grep

Descripción: Este comando busca expresiones regulares que se le pasan como parámetro dentro de archivos dando como salida donde aparecen dentro del archivo esas expresiones. Así, por ejemplo, este comando se puede utilizar para la búsqueda de archivos que contengan determinadas cadenas de caracteres o para la búsqueda dentro de archivos de determinadas expresiones. Sintaxis:

`$grep [opciones] expresión_regular archivos`

Opciones:

-v	Muestra las líneas del archivo que no contienen la expresión regular.
-c	Cuenta las líneas que contiene la expresión regular.
-l	Sólo muestra los nombres de los archivos que contiene al menos una coincidencia con expresión regular, de hecho, sólo busca en cada archivo la primera concordancia y después sigue por otro archivo.
-h	No muestra los nombres de los archivos en los que aparecen las expresiones regulares para cada línea encontrada.
-q	Sólo busca la primera concordancia.
-n	Se muestra cada línea donde aparece la expresión regular con su número de línea.
-i	Realiza la búsqueda con independencia de mayúsculas y minúsculas.
-e expresión_regular	Se utiliza para expresiones regulares que comiencen por guión -.
-f archivo	En archivo le pasamos las cadenas a buscar.
-s	No muestra los errores en los archivos que no encuentra nada.
expresión_regular	Una expresión regular describe un conjunto de cadenas de caracteres. Las expresiones regulares se construyen combinando expresiones más pequeñas mediante ciertos operadores para formar expresiones complejas.
[...]	Representa cualquier carácter de los encerrados entre los corchetes.
[^...]	Representa todos los caracteres que no están entre los corchetes.
^...	Representa las cadenas que empiezan por la cadena que sigue al signo ^.
...\$	Representa las líneas que terminan con la expresión que precede al signo \$.
\c	Se utiliza para representar el carácter c, con independencia que el carácter tenga significado para el comando grep. Así, se utilizará para, por ejemplo, buscar cadenas con el carácter especial \$, dentro del comando grep, para buscar líneas con este carácter se tendrá que expresar este carácter como \\$

	para evitar que grep lo interprete con su significado.
.	El carácter punto se utiliza para representar cualquier carácter.
archivos	Lista de archivos en los que se quiere buscar.

Ejemplos:

`$ grep "include" h*`

Busca las líneas en los archivos que comiencen por la letra h en las que aparece la cadena include.

`$grep "[aA]lbacete.*[1234]$"*`

Busca las líneas en cualquier archivo que contenga la palabra Albacete tanto comenzando por mayúscula como por minúscula y que al final de la línea aparezca uno de los números contenidos en la lista 1 2 3 4.

`$grep ma *.txt`

Busca las líneas en los archivos con extensión txt que contengan la cadena ma, valdrá la cadena en cualquier línea, así valdrá tanto comando como maleta.

## 7.2 sort

Este comando ordena alfabéticamente la entrada, línea por línea. en primer lugar ordena los espacios en blanco, luego las letras mayúsculas y después las minúsculas, por lo que no es un orden estrictamente alfabético.

Opciones:

-r	Invierte el orden normal.
-n	Clasifica en orden numérico.
-nr	Clasifica en orden numérico inverso
-f	Considera iguales las mayúsculas y minúsculas
+n	Clasifica comenzando en el campo n+1

Ejemplo:

Clasifica por conteo de bytes, primero el más grande.

`$ls -l | sort +3nr`

Clasifica por tiempo desde inicio de sesión, primero el más antiguo.

`$who | sort +4n`

### 7.3 paste

Descripción: Este comando genera en su salida varias columnas formadas por las líneas de varios archivos. Su utilidad radica en la posibilidad de unir varios archivos en uno solo, redireccionando la salida del comando, que estaría formado por tantas columnas como archivos procesa. Sintaxis

\$paste [opciones] archivos

Opciones:

-d separador	Especifica cuál va a ser el caracter separador de las columnas que va a generar en la salida, por defecto el tabulador.
-s	Archivos a unir en la salida.

### 7.4 head

Descripción: Visualiza en pantalla las primeras líneas de un archivo. Este comando es el complemento del anterior. Con él se ven las primeras líneas especificadas de un archivo. Si no se especifica el número de líneas, se verán las primeras diez, por default. El ejemplo es el siguiente:

```
$head -3 /usr/pub/eqnchar
.\ " @(#) eqnchar 23.1 91/08/29
.\ "
.\ " Copyright (C) 1986- 1991 The Santa Cruz Operation, Inc.
$
```

### 7.5 tail (cola)

Función: Permite ver las últimas líneas especificadas de un archivo. Si se desea ver sólo la parte final de un archivo, esta opción es la correcta ya que con indicarle el número de las últimas líneas que se quieren ver, estas son las únicas que aparecen. En el siguiente ejemplo se ven las funciones de este comando:

Opciones:

offset	Número desplazamiento dentro del archivo, en número de líneas, a partir de los cuales queremos que se muestra el archivo. Si el número va precedido del signo (+), se mostrará a partir de ese número de líneas; si va precedido del signo (-), se mostrará a partir de ese número de últimas líneas.
-f	Muestra el crecimiento de un archivo, sólo termina con la combinación de teclas [ctrl.][c].
archivo	Archivo que se quiere procesar.

Ejemplos:

\$tail +10 pro.txt

Muestra a partir de la línea 10 el archivo pro.txt.

\$tail -15 pro.txt

Muestra las últimas 15 líneas del archivo pro.txt.

## 7.6 cut

Función: Permite cortar partes de un flujo especificando un separador. Puede separarse caracteres, bytes o campos. El separador por defecto es el tabulador.  
Opciones:

-d	Especificar el carácter separador. \t para tabulador, \$ para fin de línea, ^ para comienzo de línea.
-c	Para especificar los caracteres.
-b	Para especificar los bytes.
-f	Para especificar los campos.
lista	Separada por guiones indica rango; separada por comas, elementos.

Ejemplos:

Mapeo de los UUIDs con los nombres de los usuarios:

\$cut -d: -f1,5 /etc/passwd

Establecer a la variable "name" con el valor del login:

```
$name=`who am i | cut -f1 -d' '`  
$echo $name
```

## 7.7 find

Descripción: Localiza archivos desde un directorio o directorios especificados, recorriendo el árbol hacia abajo. Esta poderosa herramienta del sistema permite escudriñar en cada directorio especificado excepto en los que no se tiene permiso de lectura.

Opciones:

**-name** Esta opción es la que permite dar el nombre del archivo que se desea encontrar.



**-print**      Se utilice esta opción para indicarle al comando que imprima en pantalla la ruta completa del resultado de la búsqueda. Si no se use esta opción, el sistema no dirá si se encontró el archivo.

Por ejemplo, si se está buscando un archivo que se llama eqnchar pero no se sabe en que subdirectorios está se podría escribir lo siguiente para conocer su ruta de acceso:

```
$ find / -name eqnchar -print
...
/usr/pub/eqnchar
...
$
```

En este ejemplo la búsqueda comienza desde la raíz y de ahí a todos los subdirectorios.

Si se quisiera encontrar los archivos que tuvieran por ejemplo, la cadena tex al principio de su nombre, en el directorio de base y en el directorio del grupo se teclearía:

```
$ find . -name "tex*" -print
```

**-perm**      Le indica al comando que busque los archivos que tienen el patrón de permisos establecidos en notación octal.

El siguiente ejemplo muestra el uso de esta opción:

```
$ find . -perm 700 -print
```

**-type**      Encuentra los archivos por tipo.

Por ejemplo, para encontrar a los que son directorios se teclea:

```
$ find . -type d -print
```

**-user**      Esta opción le indica al comando que restrinja la búsqueda a sólo los archivos que pertenecen al usuario especificado:

```
$ find . -type f -user nancy -print
```

**-size**      Encuentra a los archivos con la longitud especificada en bytes.

**-atime** Encuentra a los archivos que no han sido accedidos desde el plazo indicado, en días.

**-mtime** Encuentra a los archivos que hayan sido modificados en el plazo elegido, también en días.

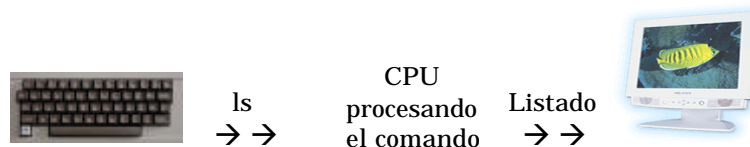
Algunas veces el proceso de búsqueda puede tardar demasiado, así que algo que se puede hacer es aprovechar la característica de UNIX de ser un sistema multitarea y ejecutar dicho proceso en background, es decir realizarlo como un trabajo asíncrono.

Una línea de comandos se puede ejecutar en background si al final de ella se le agrega un símbolo "&" (ampersand), por ejemplo:

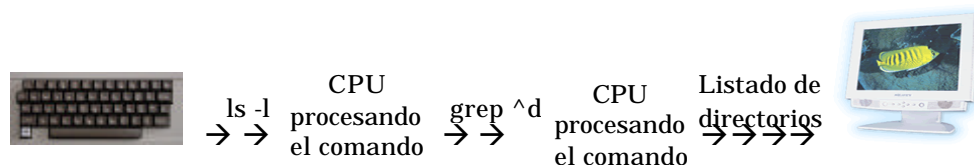
```
$find / -name "tex*" -print > busqueda &
```

## 7.8 Interconexión

Para la interconexión de comandos se utiliza el símbolo del "pipe": |. Este símbolo es indicador de una redirección de salida hacia una entrada especificada después del símbolo. En una PC, la salida estándar es normalmente el monitor y la entrada estándar es normalmente el teclado. De esta manera un comando promedio actúa de la siguiente manera:



No obstante, puede ser de utilidad para el usuario el indicar a un comando que su flujo de entrada deberá ser el flujo de salida del otro. De manera que se obtenga lo siguiente:



Lo anterior se logra con la secuencia de comandos:

```
$ls -l | grep ^d
```

La interconexión de comandos no tiene límite.

## 8. El Editor vi

El editor vi es un editor difícil de usar, y muy poco intuitivo. Sin embargo, al ser un editor que ocupa muy poco, suele estar instalado en todos los sistemas Linux, por lo que nunca es malo conocer algo sobre este tipo de editores. Aquí podemos ver una imagen de vi, al ejecutarlo en la línea de comandos:

```
$vi
```

Este editor trabaja en modo texto y en modo comando. Estos comandos pueden ser separados en comandos básicos y avanzados. El editor vi, posee tres posibles modos de funcionamiento, el modo órdenes, que es el modo inicial, modo inserción y modo de última línea.

- En el modo órdenes, vi espera comandos propios para trabajar con los archivos o cambiar a los otros modos. Prácticamente la mayoría de las órdenes que se pueden dar en este editor tienen sólo uno o dos caracteres, aunque existe alguno de más y muchos de ellos aceptan parámetros.
- En el modo inserción se insertará, se eliminará y en general se editará el texto, que en realidad es el modo más normal de trabajo. Para pasar del modo órdenes al modo inserción basta con dar una coma de edición como «a» de añadir o «i» de insertar. Del modo de inserción podemos salir en cualquier momento pulsando la tecla <Esc>.
- El modo de última línea es un modo especial que se utiliza para dar a vi una serie de órdenes específicas, y aparecen en la última línea de la ventana de edición. Estos comandos suelen ser de carácter general y de gran utilidad. Para entrar en este modo sólo tenemos que pulsar la tecla «:», tras lo que damos la orden y pulsamos <Enter> para ejecutarla. Entre estos comandos podemos encontrar órdenes como «q», que nos permite abandonar el vi o «q!» para salir del vi sin guardar los cambios.

El editor vi trabaja con un buffer en memoria, por lo que la información no se lleva a disco hasta que el usuario así lo indique. Por este motivo se recomienda usar el comando :w cada cierto tiempo para evitar perder todo el trabajo realizado durante una gran cantidad de tiempo de trabajo. Para empezar podemos dar un nombre al fichero mediante un argumento del vi:

```
$vi archivo_de_texto
```

## 8.1 Comandos básicos

Comandos del modo de última línea	Comando
Salir del vi	:q
Guardar cambios	:w
Guardar cambios en el archivo	:w nombre_de_archivo
Guardar cambios y sobrescribir el archivo	:w! nombre_de_archivo
Guardar cambios y salir	:wq
Guardar cambios y salir	ZZ
Guardar cambios y salir	:x
Salir sin guardar cambios	:q!

Movimiento de cursores	Comando
Una línea abajo	k
Una línea arriba	j
Un carácter a la derecha	h
Un carácter a la izquierda	l

Salto de línea	Retorno de carro
----------------	------------------

Avanza una palabra	w
Va al principio de la palabra actual	b
Va al final de la palabra actual	e
Mover el cursor al principio de la línea	0
Mover el cursor al final de la línea	\$

Ir a la primera línea de la pantalla	Mayús-h
Ir a la línea situada en mitad de la pantalla	Mayús-m
Ir a la última línea de la pantalla	Mayús-l
Ir a enésima línea	nG
Avanza una pantalla	ctrl.-f
Retrocede una pantalla	ctrl.-b

Comandos para añadir texto	Comando
Añadir texto después de la posición del cursor.	a
Se pone en modo entrada y añade texto al final de la línea actual.	A
Insertar texto delante de la posición del cursor.	i
Se pone en modo entrada e inserta texto al principio de la línea actual.	I
Inserta una línea por debajo de la línea actual para añadir texto.	o
Inserta una línea encima de la línea actual para añadir texto.	O

Comandos para borrar texto	Comando
Borrar un carácter en la posición del cursor	x
Borrar n caracteres en la posición del cursor	nx
Suprime desde la posición del cursos hasta el principio de la siguiente.	dw
Suprime n palabras el cursor hasta el principio de la siguiente.	ndw
Suprime texto desde la posición del cursor hasta el final de la línea.	d\$
Al igual que d\$, suprime el resto de la línea actual.	D
Suprime la línea actual entera	dd
Elimina n líneas	ndd
Comandos para buscar texto	Comando
Busca la cadena hacia delante.	/cadena
Busca la cadena hacia atrás.	?cadena
Vuelve a busca en la dirección actual.	n
Vuelve a buscar en a dirección opuesta	Mayús-n
Comandos para sustituir texto	Comando
Sustituir un carácter	r
Reemplaza una secuencia de caracteres.	R
Cambia la palabra actual desde el cursor hasta el final	cw
Palabra actual, desde el principio hasta el carácter anterior al cursor	ce
Acción modo texto	Comando
Abandonar modo texto	<Esc>
Deshacer último cambio	u
Deshacer todos los cambios de la línea	U

## 9. Proceso de arranque, inicio y cierre del sistema

Una de las características más importantes y poderosas de Red Hat Linux es el método abierto y configurable para el inicio y cierre del sistema operativo. Los usuarios son libres de configurar muchos aspectos del proceso de arranque, incluyendo qué programas se lanzarán al momento de arranque. De forma parecida, el cierre del sistema finaliza los procesos de forma organizada y configurable, aunque la personalización de este proceso casi nunca es necesaria.

Entender el funcionamiento del proceso de arranque y cierre no solo le permitirá personalizar fácilmente Red Hat Linux, sino que también le facilitará resolver problemas relacionados con el inicio y el cierre del sistema.

### 9.1 Proceso de arranque

A continuación se tienen las etapas básicas del proceso de arranque para un sistema x86:

1. La BIOS del sistema comprueba y lanza la primera etapa del gestor de arranque del MBR del disco duro primario.
2. La primera etapa del gestor de arranque se carga automáticamente en memoria y lanza la segunda etapa del gestor de arranque desde la partición `/boot/`.
3. La segunda etapa del gestor de arranque carga el kernel en memoria, lo cual en su momento carga los módulos necesarios y monta la partición root para sólo-lectura.
4. El kernel transfiere el control del proceso de arranque al programa `/sbin/init`.
5. El programa `/sbin/init` carga todos los servicios y herramientas de espacio del usuario y monta todas las particiones listadas en `/etc/fstab`.
6. El usuario se presenta con un intérprete de comandos de registro para el sistema Linux apenas arrancado.

### 9.2. Vista detallada del proceso de arranque

El inicio del proceso de arranque varía dependiendo de la plataforma de hardware usada. Sin embargo, una vez que se encuentra el kernel y se carga el sistema, el proceso de arranque por defecto es idéntico a través de todas las arquitecturas. Este capítulo se basa en la arquitectura x86.

### 9.2.1 La BIOS

Cuando una computadora x86 se carga, el procesador busca al final de la memoria del sistema por Basic Input/Output System o el programa BIOS y lo ejecuta. La BIOS controla no sólo el primer paso del proceso de arranque, sino que también proporciona una interfaz de bajo nivel para dispositivos periféricos. Por este motivo se escribe tan sólo en modo lectura, memoria permanente y está siempre disponible para el uso.

Otras plataformas usan programas diferentes para ejecutar tareas a bajo nivel equivalentes a aquellas de la BIOS en el sistema x86. Por ejemplo, los ordenadores basados en Itanium usan Interfaz de Firmware extensible (EFI) Shell, mientras que los sistemas Alpha usan SRM console.

Una vez que se ha cargado, la BIOS checa los periféricos y localiza un dispositivo con el que arrancar el sistema. En primer lugar se comprueba cualquier disquete y unidades de CD-ROM presente por los medios de arranque, y a continuación si esto falla, se echa un vistazo a las unidades de disco duro del sistema. El orden de las unidades necesario para arrancar puede ser controlado con una configuración de la BIOS. La BIOS carga en memoria cualquier programa que resida en el primer sector de este dispositivo, llamado Master Boot Record o MBR. La MBR sólo tiene 512 bytes de tamaño y contiene las instrucciones de código de máquina para el arranque del equipo, llama un gestor de arranque así como también la tabla de particiones. Una vez que la BIOS ha encontrado y cargado el gestor de arranque en memoria, le deja el control del proceso de arranque a éste.

### 9.2.2 El gestor de arranque

Esta sección revisa los gestores de arranque para la plataforma x86. Dependiendo de la arquitectura del sistema, el proceso de arranque diferirá ligeramente. Bajo Red Hat Linux están disponibles dos gestores de arranque: GRUB o LILO. GRUB es el gestor de arranque por defecto, pero LILO está disponible para los usuarios que lo necesiten o prefieran.

Los gestores de arranque de Linux para la plataforma x86 se dividen en dos etapas. La primera es un código binario de máquina pequeña en el MBR. Su única función es la de localizar el gestor de arranque de la segunda etapa y cargar la primera parte de éste en memoria.

GRUB es el gestor de arranque más nuevo y tiene la ventaja de ser capaz de leer particiones ext2 y ext3 y cargar su archivo de configuración – /boot/grub/grub.conf –al momento de arranque.

Con LILO, la segunda etapa del gestor de arranque es usar la información del MBR para determinar las opciones de arranque disponibles para el usuario. Esto significa que cada vez que se produzca un cambio en la configuración o actualice el kernel de

forma manual, debe ejecutar el comando `/sbin/lilo -v -v` para escribir la información apropiada al MBR.



### Sugerencia

Si se actualiza el kernel mediante el uso de **Agente de actualización de Red Hat**, el archivo de configuración se actualiza automáticamente.

Una vez que el gestor de arranque de la segunda etapa está en memoria, se presenta al usuario con la pantalla inicial gráfica de Red Hat Linux, se muestran los diferentes sistemas operativos o kernels para los que ha sido configurado para arrancar. En esta pantalla se puede usar las flechas direccionales para escoger el sistema operativo o kernel con el que se desea arrancar y se presiona la tecla [Enter]. Si no se presiona ninguna tecla, el gestor de arranque carga la selección predeterminada luego de un período de tiempo de espera (también configurable).



### Nota

Si se ha instalado el soporte para el kernel Symmetric Multi-Processor (SMP), se verá más de una opción la primera vez que se arranque el sistema. Bajo LILO se verá Linux, el cual es el kernel SMP, y Linux-up, el cual es para procesadores únicos. En GRUB se mostrará Red Hat Linux (<kernel-version>-smp), el cual es el kernel SMP, y Red Hat Linux (<kernel-version>), la cual es para procesadores únicos.

Si surge cualquier problema con el kernel SMP, trate de seleccionarse un kernel que no sea SMP antes de volver a arrancar.

Una vez que el gestor de arranque de la segunda etapa ha determinado qué kernel arrancar, se localizará el binario del kernel correspondiente en el directorio `/boot/`. El kernel binario se llama usando el siguiente formato `— /boot/vmlinuz-<kernel-version>` (donde <kernel-version> corresponde a la versión del kernel especificada en las configuraciones del gestor de arranque).

El gestor de arranque luego coloca la imagen apropiada de initial RAM disk, conocida como `initrd`, en la memoria. El `initrd` es usado por el kernel para cargar controladores necesarios para arrancar el sistema. Esto es muy importante si se posee unidades de disco duro SCSI o si se está usando el sistema de ficheros `ext3`.



### Aviso

No se elimine el directorio `/initrd/` del sistema de ficheros bajo ningún concepto. Si se elimina el sistema dará un mensaje de error de pánico en el momento de arranque.



Una vez que el kernel y la imagen `initrd` se cargan en memoria, el gestor de arranque controla el proceso de arranque para el kernel.

#### **9.2.2.1 Gestores de arranque para otras arquitecturas**

Una vez que el kernel de Red Hat Linux se arranca y se pasa el proceso de arranque al comando `init`, los mismos acontecimientos suceden en cada arquitectura exactamente en el mismo modo. La única diferencia entre el proceso de arranque de cada arquitectura está en la aplicación que se usa para encontrar y cargar el kernel.

Por ejemplo, en la arquitectura Alpha se usa el gestor de arranque `aboot`, mientras que en Itanium se usa el gestor de arranque `ELILO`.

#### **9.2.3. El kernel**

Cuando el kernel se carga, inmediatamente se inicializa y configura la memoria de la computadora y los diferentes hardwares conectados al sistema, incluyendo procesadores, subsistemas de entrada/salida y dispositivos de almacenamiento. A continuación se buscará la imagen `initrd` en una ubicación predeterminada en memoria, se descomprimirá, se montará y se cargará todos los controladores necesarios. A continuación se inicializa los dispositivos virtuales relacionados con el sistema de ficheros, tal como LVM o software RAID antes de desmontarse la imagen del disco `initrd` y liberarse toda la memoria que la imagen del disco ocupó anteriormente.

El kernel luego crea un dispositivo `root`, monta la partición `root` como sólo lectura y libera cualquier memoria no utilizada. Llegados a este punto, el kernel está cargado en memoria y operativo. Sin embargo, como no hay aplicaciones de usuario que permitan la entrada significativa de datos al sistema, no se puede hacer mucho más. Para configurar el entorno de usuario, el kernel inicia el programa `/sbin/init`.

#### **9.2.4 Programa `/sbin/init`**

El programa `/sbin/init` (también llamado `init`) coordina el resto del proceso de arranque y configura el ambiente del usuario.

Cuando el comando `init` se arranca, se vuelve el padre o abuelo de todos los procesos que comienzan automáticamente en el sistema Red Hat Linux. Primero, se ejecuta el script `/etc/rc.d/rc.sysinit`, que establece la ruta a otros programas, se activa el `swap`, se controla los sistemas de fichero y se encarga de todo lo que el sistema necesita tener hecho al momento de la inicialización. Por ejemplo, la mayoría de los sistemas usan un reloj, por lo tanto, en ellos, el `rc.sysinit` tendrá una referencia `/etc/sysconfig/clock` para inicializar el reloj. Otro ejemplo es si hay procesos en los puertos seriales especiales que deben ser inicializados, `rc.sysinit` ejecutará el archivo `/etc/rc.serial`.

El comando `init` luego ejecuta el script `/etc/inittab`, que describe cómo debería configurarse el sistema en cada nivel de ejecución de SysV `init`. Entre otras cosas, `/etc/inittab` configura el nivel de ejecución por defecto y establece que `/sbin/update` debería de ejecutarse cuando se arranque un nivel de ejecución en concreto.

A continuación, el comando `init` configura la librería de función de fuente, `/etc/rc.d/init.d/functions`, para el sistema. Esto indica el modo en que empezar o matar un programa y cómo determinar el PID del programa.

El programa `init` inicia todos los procesos de fondo buscando en el directorio apropiado `rc` por el nivel de ejecución especificado por defecto en `/etc/inittab`. Los directorios `rc` están numerados para corresponder al nivel de ejecución que represente. Por ejemplo, `/etc/rc.d/rc5.d/` es el directorio para el nivel de ejecución 5.

Cuando se arranca el nivel de ejecución 5, el programa `init` consulta el directorio `/etc/rc.d/rc5.d/` para determinar qué procesos iniciar o parar.

A continuación un ejemplo de listado del directorio `/etc/rc.d/rc5.d/`:

```
K05innd → ../init.d/innd
K05sasauthd → ../init.d/sasauthd
K10psacct → ../init.d/psacct
K12cWnn → ../init.d/cWnn
K12FreeWnn → ../init.d/FreeWnn
K12kWnn → ../init.d/kWnn
K12mysqld → ../init.d/mysqld
K12tWnn → ../init.d/tWnn
K15httpd → ../init.d/httpd
K15postgresql → ../init.d/postgresql
K16rarpd → ../init.d/rarpd
K20bootparamd → ../init.d/bootparamd
K20iscsi → ../init.d/iscsi
K20netdump-server → ../init.d/netdump-server
K20nfs → ../init.d/nfs
K20rstatd → ../init.d/rstatd
K20rusersd → ../init.d/rusersd
K20rwalld → ../init.d/rwalld
K20rwhod → ../init.d/rwhod
K24irda → ../init.d/irda
K25squid → ../init.d/squid
K28amd → ../init.d/amd
K34dhcrelay → ../init.d/dhcrelay
K34yppasswdd → ../init.d/yppasswdd
K35atalk → ../init.d/atalk
K35dhcpd → ../init.d/dhcpd
K35smb → ../init.d/smb
```

K35vncserver → ../init.d/vncserver  
K35winbind → ../init.d/winbind  
K40mars-nwe → ../init.d/mars-nwe  
K45arpwatch → ../init.d/arpwatch  
K45named → ../init.d/named  
K45smartd → ../init.d/smartd  
K46radvd → ../init.d/radvd  
K50netdump → ../init.d/netdump  
K50snmpd → ../init.d/snmpd  
K50snmptrapd → ../init.d/snmptrapd  
K50tux → ../init.d/tux  
K54pxe → ../init.d/pxe  
K55routed → ../init.d/routed  
K61ldap → ../init.d/ldap  
K65identd → ../init.d/identd  
K65kadmin → ../init.d/kadmin  
K65kprop → ../init.d/kprop  
K65krb524 → ../init.d/krb524  
K65krb5kdc → ../init.d/krb5kdc  
K70aep1000 → ../init.d/aep1000  
K70bcm5820 → ../init.d/bcm5820  
K74ntpd → ../init.d/ntpd  
K74ups → ../init.d/ups  
K74ypserv → ../init.d/ypserv  
K74ypxfrd → ../init.d/ypxfrd  
K84bgpd → ../init.d/bgpd  
K84ospf6d → ../init.d/ospf6d  
K84ospfd → ../init.d/ospfd  
K84ripd → ../init.d/ripd  
K84ripngd → ../init.d/ripngd  
K85zebra → ../init.d/zebra  
K90isicom → ../init.d/isicom  
K92ipvsadm → ../init.d/ipvsadm  
K95firstboot → ../init.d/firstboot  
S00microcode\_ctl → ../init.d/microcode\_ctl  
S05kudzu → ../init.d/kudzu  
S08ip6tables → ../init.d/ip6tables  
S08ipchains → ../init.d/ipchains  
S08iptables → ../init.d/iptables  
S09isdn → ../init.d/isdn  
S10network → ../init.d/network  
S12syslog → ../init.d/syslog  
S13portmap → ../init.d/portmap  
S14nfslock → ../init.d/nfslock  
S17keytable → ../init.d/keytable  
S20random → ../init.d/random

```
S24pcmcia → ../init.d/pcmcia
S25netfs → ../init.d/netfs
S26apmd → ../init.d/apmd
S28autofs → ../init.d/autofs
S44acpid → ../init.d/acpid
S55sshd → ../init.d/sshd
S56rawdevices → ../init.d/rawdevices
S56xinetd → ../init.d/xinetd
S80sendmail → ../init.d/sendmail
S80spamassassin → ../init.d/spamassassin
S84privoxy → ../init.d/privoxy
S85gpm → ../init.d/gpm
S90canna → ../init.d/canna
S90crond → ../init.d/crond
S90cups → ../init.d/cups
S90xfs → ../init.d/xfs
S95anacron → ../init.d/anacron
S95atd → ../init.d/atd
S97rhnsd → ../init.d/rhnsd
S99local → ../rc.local
S99mdmonitor → ../init.d/mdmonitor
```

Como se puede ver, ninguno de los scripts que inician y cierran los servicios está localizados en el directorio `/etc/rc.d/rc5.d/`. Casi todos los ficheros en `/etc/rc.d/rc5.d/` son enlaces simbólicos que apuntan a los scripts localizados en el directorio `/etc/rc.d/init.d/`. Los enlaces simbólicos se usan en cada uno de los directorios `rc` de manera que los niveles de ejecución puedan ser reconfigurados al crear, modificar y eliminar los enlaces simbólicos sin que se afecte a los scripts actuales a los que se refiere.

El nombre de cada enlace simbólico inicia con `K` o `S`. Los enlaces `K` son procesos eliminados en ese nivel de ejecución, mientras que aquellos que inician por `S` son procesos iniciados.

El comando `init` en primer lugar detiene todos los enlaces simbólicos de `K` en el directorio mediante la ejecución del comando `/etc/rc.d/init.d/<command> stop`, en el que `<command>` es el proceso a matar. A continuación se inicia todos los enlaces simbólicos `S` al ejecutar `/etc/rc.d/init.d/<command>. start`.



### **Sugerencia**

Una vez que el sistema ha acabado el arranque podrá registrarse como `root` y ejecutar los mismos scripts para iniciar y parar los servicios. Por ejemplo, el comando `/etc/rc.d/init.d/httpd stop` paralizará el servidor Web Apache.

Cada uno de los enlaces simbólicos se numera para dictaminar el orden de inicio. Puede cambiarse el orden en el que los servicios inician o paran al cambiarse el número. Mientras más bajo es el número, más rápido se arrancará. Los enlaces simbólicos con el mismo número se inician de modo alfabético.



#### **Nota**

Una de las últimas cosas que el programa init ejecuta es el archivo `/etc/rc.d/rc.local`. Este archivo es útil para la personalización del sistema.

Después que el comando init ha progresado a través del directorio adecuado rc para el nivel de ejecución, el script `/etc/inittab` bifurca los procesos `/sbin/mingetty` para cada consola virtual ubicada para el nivel de ejecución. Los niveles de ejecución del 2 al 5 obtienen todas las seis consolas virtuales, mientras que el nivel de ejecución 1 (modo usuario único) obtiene tan sólo uno y los niveles de ejecución del 0 al 6 no obtienen ninguno. El proceso `/sbin/mingetty` abre las rutas de la comunicación para los dispositivos tty, establece sus modos, imprime el indicador de inicio de sesión, toma el nombre del usuario, e inicia el proceso de inicio de sesión para el usuario.

En el nivel de ejecución 5, el `/etc/inittab` ejecuta un script llamado `/etc/X11/prefdm`. El script ejecuta `prefdm` ejecuta su gestor de pantalla preferido para X — `gdm`, `kdm`, o `xdm`, dependiendo de los contenidos del archivo `/etc/sysconfig/desktop`.

En este punto, el sistema está operando en el nivel de ejecución nivel 5 y mostrando la pantalla de inicio de sesión.

### **9.3 Ejecutar programas adicionales en el momento de arranque**

El script `/etc/rc.d/rc.local` lo ejecuta el comando init en tiempo de arranque, o cuando se cambien niveles de ejecución. El agregar comandos a este script es una forma fácil de realizar tareas necesarias como arrancar servicios especiales o inicializar dispositivos sin tener que escribir scripts complejos de inicialización en el directorio `/etc/rc.d/init.d/` y creando los enlaces simbólicos.

El script `/etc/rc.serial` es usado si se deben configurar puertos seriales en el momento de arranque. Este script ejecuta los comandos `setserial` para configurar los puertos seriales del sistema.

### **9.4 Niveles de ejecución de SysV Init**

El sistema de niveles de ejecución SysV init provee de un proceso estándar para controlar cuáles programas init lanza o detiene cuando se inicializa un nivel de ejecución. SysV init fue escogido porque es más fácil de usar y más flexible que el proceso tradicional init estilo BSD.

Los ficheros de configuración para SysV init están en el directorio `/etc/rc.d/`. Dentro de este directorio, se encuentran los scripts `rc`, `rc.local`, `rc.sysinit`, y, opcionalmente, los scripts `rc.serial` así como los siguientes directorios:

```
init.d/  
rc0.d/  
rc1.d/  
rc2.d/  
rc3.d/  
rc4.d/  
rc5.d/  
rc6.d/
```

El directorio `init.d/` contiene los scripts usados por el comando `/sbin/init` cuando se controlan los servicios. Cada uno de los directorios numerados representa los seis niveles de ejecución predeterminados configurados por defecto bajo Red Hat Linux.

#### 9.4.1 Niveles de ejecución

Los niveles de ejecución son un estado, o modo, definido por los servicios listados en el SysV directorio `/etc/rc.d/rc<x>.d/`, donde `<x>` es el número de nivel de ejecución.

La idea detrás de los niveles de ejecución de SysV init gira alrededor del hecho que sistemas diferentes se pueden usar de formas diferentes. Por ejemplo, el servidor corre de forma más eficiente sin tener que arrastrar recursos del sistema creados por el sistema X. Otras veces, el administrador del sistema puede necesitar operar el sistema en un nivel más bajo de ejecución para realizar tareas de diagnóstico, como reparar corrupción del disco duro, cuando no es posible que ningún otro usuario esté usando el sistema.

Las características de un nivel de ejecución dado determinan qué servicios son detenidos o iniciados por init. Por ejemplo, el nivel de ejecución 1 (modo único usuario) detiene cualquier servicio de red, mientras que el nivel 3 arranca estos servicios. Asignando servicios específicos a ser detenidos o arrancados en un nivel dado, init se puede fácilmente cambiar el modo de la máquina sin que el usuario tenga que manualmente arrancar o detener servicios. Los siguientes niveles de ejecución están definidos por defecto para Red Hat Linux:

- 0 — Parar
- 1 — Modo texto usuario único
- 2 — Sin usar (usuario-definible)
- 3 — Modo texto multiusuario completo
- 4 — Sin usar (usuario-definible)
- 5 — Modo gráfico multiusuario completo (con una pantalla de inicio de sesión basada en X)
- 6 — Volver a arrancar

Generalmente, los usuarios utilizan Red Hat Linux al nivel de ejecución 3 o nivel de ejecución 5 — ambos modos multiusuario. Ya que los niveles de ejecución 2 y 4 no son usados, los usuarios a veces personalizan estos niveles para cubrir necesidades específicas.

El nivel de ejecución por defecto para el sistema está listado en `/etc/inittab`. Para saber el nivel de ejecución por defecto de un sistema, se busca por la línea similar a la que se muestra abajo cerca de la parte superior de `/etc/inittab`:

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

El nivel de ejecución predeterminado en el ejemplo de arriba es cinco, como indica el número después del punto y coma. Para cambiarlo, se modifica `/etc/inittab` como usuario root.



#### Aviso

Se debe tener mucho cuidado cuando se esté modificando `/etc/inittab`. Errores simples de tipo pueden hacer que el sistema no arranque nuevamente. Si esto ocurre, deberá usarse un disquete de arranque, entrar a modo de usuario único o en modo de rescate y reparar el archivo.

Es posible cambiar al nivel de ejecución por defecto al momento de arranque modificando los argumentos pasados del gestor de arranque al kernel.

### 9.4.2 Utilidades de los niveles de ejecución

Una de las mejores formas de configurar los niveles de ejecución es usando `initscript` utility. Estas herramientas están diseñadas para simplificar las tareas de mantener archivos en la jerarquía del directorio SysV init y descargan a los administradores de sistemas de tener que directamente manipular numerosos enlaces simbólicos en los subdirectorios de `/etc/rc.d/`.

Red Hat Linux ofrece tres de tales utilidades:

- **`/sbin/chkconfig`** — Es una herramienta de línea de comandos sencilla para mantener la jerarquía del directorio `/etc/rc.d/init.d`.
- **`/sbin/ntsysv`** — Provee de una interfaz interactiva basada en texto, que muchos encuentran más fácil de usar que `chkconfig`.
- **Herramienta de configuración de servicios** — Es una utilidad flexible basada en GTK2 para la configuración de niveles de ejecución.

## 9.5 Apagar

Para apagar Red Hat Linux, puede ejecutarse el comando `/sbin/shutdown`. Las opciones más usadas son:

```
/sbin/shutdown -h now  
/sbin/shutdown -r now
```

Después de apagar todo, la opción `-h` detendrá la máquina, y la opción `-r` la reiniciará. Para el caso de los usuarios no root pueden usarse los comandos `reboot` y `halt` para apagar el equipo mientras se está en niveles de ejecución 1 hasta 5. Sin embargo, no todos los sistemas operativos Linux soportan esta característica. Si la computadora no se apaga, téngase cuidado de no apagar la computadora hasta que aparezca un mensaje indicando que el sistema ha sido detenido. No esperar por este mensaje puede significar que no todas las particiones de discos duros han sido desmontadas, y puede llevar a un sistema de archivos corrupto.



## 10. Disquetes y CD-ROMs

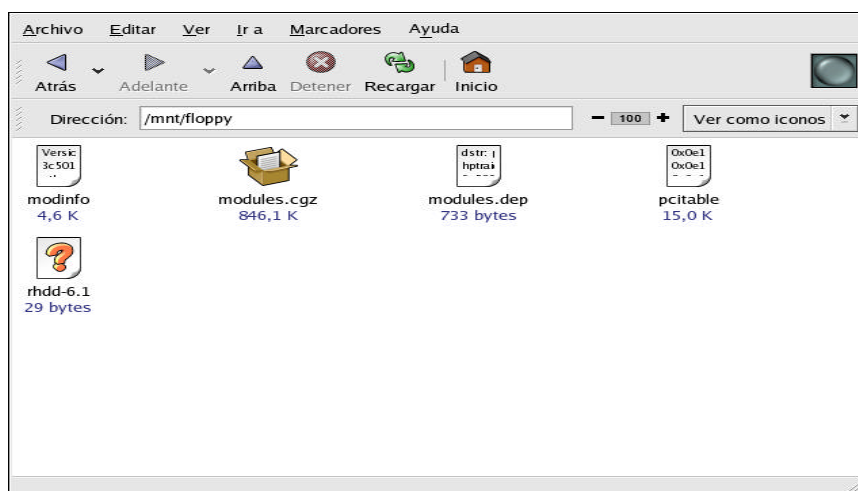
El uso de disquetes y discos CD-ROM con el sistema Red Hat Linux exige tener algunos conocimientos sobre medios extraíbles. En este capítulo se describe cómo leer y escribir archivos de/a disquetes, cómo dar formato a disquetes, y cómo leer y copiar datos de un disco CD-ROM. En este capítulo también se explica el uso de las unidades de CD de escritura y reescritura.

### 10.1 Uso de disquetes

Los disquetes son una de las soluciones de medio extraíble más antiguas que hay disponibles para la computadora personal (PC). Constituyen la solución de almacenamiento portátil ideal para archivos pequeños que se deben llevar físicamente a otros lugares. Por ejemplo, si dos PC no están en la misma red, los disquetes son una solución perfecta para transferir la información de una computadora a otro.

#### 10.1.1 Montaje y desmontaje de un disquete

Un disquete, antes de utilizarse, se debe montar. Se inserta un disquete en la unidad de disquetes y se escribe `mount /mnt/floppy/` en el indicador de comandos de shell. La luz de actividad de la unidad de disquetes parpadea mientras el sistema de archivos del disquete se monta en el directorio `/mnt/floppy`. Puede tenerse acceso al contenido del disquete cambiándose al directorio con el comando `cd /mnt/floppy/`. Como alternativa, también puede montarse un disquete si se hace click con el botón derecho del ratón en el escritorio y selecciona **Discos => Disquete**. Con esto se montará el disquete y se agregará el icono de escritorio sobre el que puede hacer doble click para explorar el contenido del disquete. Ahora se puede copiar información del mismo y escribirse información en él. Se puede abrir, guardar y copiar archivos al/del disquete del mismo modo que se haría con la unidad de disco duro o usar un explorador.



#### 10.a Ver los archivos de un disquete con la aplicación Nautilus

Cuando se ha terminado de usar el disquete, deberá desmontarse antes de expulsarlo de la unidad. Para ello, se cierra cualquier aplicación que pueda estar utilizando los archivos del disquete o explorando el contenido del disquete (por ejemplo, la aplicación **Nautilus** o **Konqueror**) y se escribe en el indicador de comandos de shell el siguiente comando:

```
umount /mnt/floppy/
```

Si está utilizándose GNOME, se desmonta el disquete haciendo clic con el botón



derecho del ratón en el icono y seleccionando **Desmontar el volumen** en el menú. Ahora puede expulsarse de forma segura el disquete de la unidad.

### 10.1.2 Colocar archivos Linux en un disquete MS-DOS

Para copiar archivos desde una máquina Linux a un disquete formateado bajo MS-DOS para que una máquina Windows lo pueda leer, deberá formatearse el disquete con un formato de sistema de archivos MS-DOS (FAT). Esto se puede hacer con el Windows OS o con **gfloppy**. Luego se monta en Linux como ya se describió. Se copia los archivos usando el comando siguiente (sustituyendo *filename* con el nombre del archivo que desea copiar):

```
Cp filename /mnt/floppy
```

Puede luego desmontarse el disquete y sacarlo de la unidad. El nuevo archivo en el disquete deberá ser ahora accesible desde la máquina Windows.

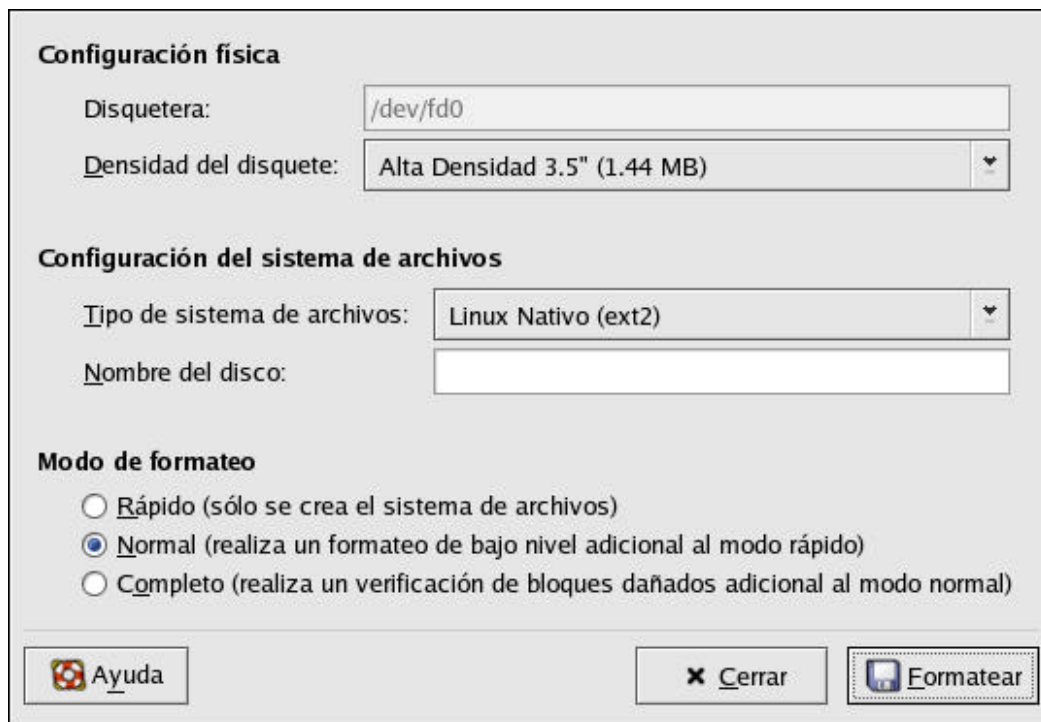
### 10.1.3 Formatear un disquete

Para usar un disquete específicamente con el sistema Red Hat Linux, deberá dársele formato utilizando el sistema de archivos ext2 o ext3, que son sistemas de archivos compatibles con Red Hat Linux. Una vez creado el sistema de archivos en el disquete, puede manipularse el contenido del mismo modo que se manipula los directorios y los archivos de la unidad de disco duro.

#### 10.1.3.1 Uso de gfloppy

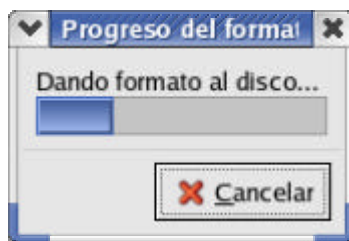
Para arrancar **gfloppy**, se selecciona **Menú principal → Herramientas del sistema → Formateador de disquetes**. En el indicador de comandos de shell, se escribe `/usr/bin/gfloppy`. Como se muestra en la Figura 10.b, la interfaz **gfloppy** es pequeña y tiene pocas opciones. La configuración por defecto es suficiente para satisfacer a la mayoría de los usuarios y las necesidades. Sin embargo, puede darse formato al disquete con el tipo de sistema de archivos de MS-DOS si se desea. También puede seleccionarse la densidad del disquete (si no se utiliza la alta densidad normal de los

disquetes de 3,5" de 1.44 MB). También se puede optar por dar *formato rápido* al disquete si anteriormente fue formateado con ext2 o ext3.



### 10.b gfloppy

Se inserta un disquete y se modifica la configuración en **gfloppy** para adaptarla a las necesidades específicas y, a continuación, se hace click en **Formatear**. El cuadro de estado aparecerá en la parte superior de la ventana principal y mostrará el estado del proceso de formato y verificación (Figura 10.c). Una vez terminado, puede expulsar se el disquete y cerrarse **gfloppy**.



### 10.c Cuadro de estado de gfloppy

### 10.1.3.2. Uso de mke2fs

El comando `mke2fs` es usado para crear el sistema de archivos de Linux `ext2` en un dispositivo tal como una partición de disco duro o (en este caso) un disquete. `mke2fs` esencialmente formatea el dispositivo y crea un dispositivo vacío compatible con Linux que puede luego ser usado para almacenar archivos y datos. Se inserta el disquete en la unidad y se ejecuta el comando siguiente en el intérprete de comandos:

```
/sbin/mke2fs /dev/fd0
```

En sistemas Linux, `/dev/fd0` se refiere a la primera unidad de disquete. Si la computadora tiene más de una unidad, la unidad primaria es `/dev/fd0`, la segunda unidad es `/dev/fd1` y así sucesivamente.

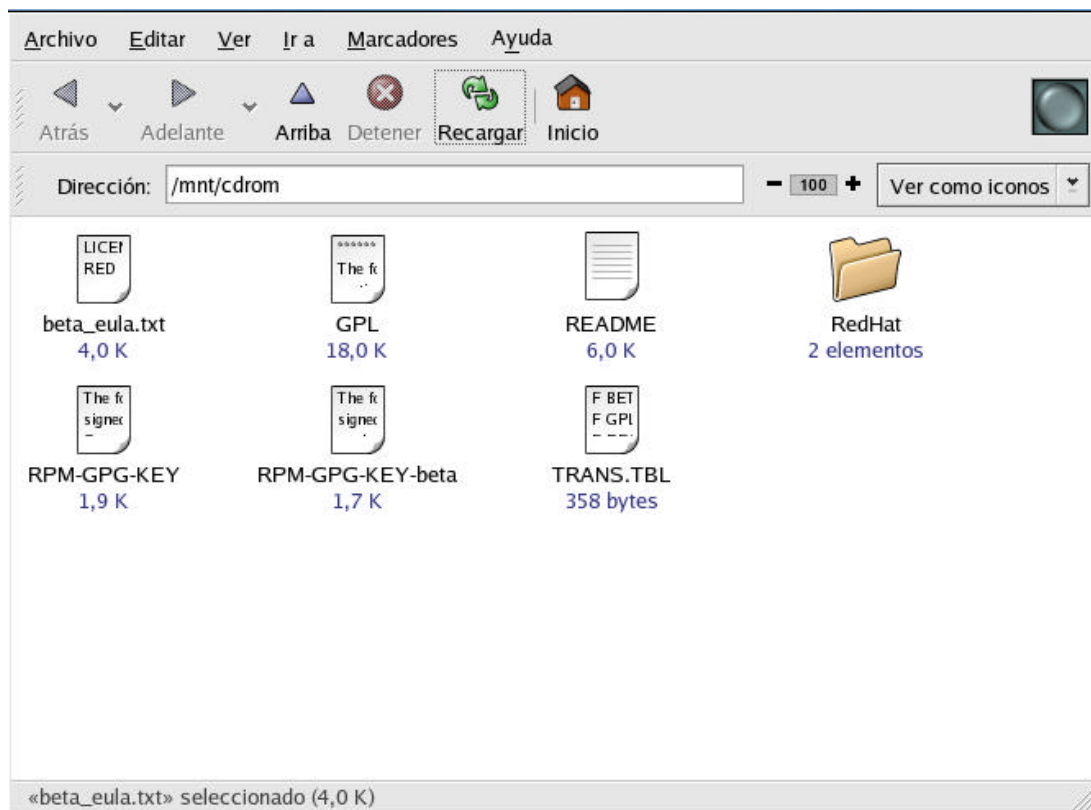
La utilidad `mke2fs` tiene varias opciones. La opción `-c` hace que el comando `mke2fs` verifique el dispositivo por sectores dañados antes de crear el sistema de archivos. Una vez que se haya creado el sistema de archivos `ext2` en el disquete, está listo para ser usado con su sistema Red Hat Linux.

## 10.2. CD-ROMs

El formato CD-ROM es un medio muy conocido para aplicaciones de software y presentaciones multimedia de gran tamaño. La mayoría del software que se puede adquirir en los comercios de productos se presentan en forma de CD-ROM. En esta sección se describe cómo utilizar los discos CD-ROM en el sistema Red Hat Linux.

### 10.2.1. Uso de discos CD-ROM con su administrador de archivos

Por defecto, el sistema Red Hat Linux detecta automáticamente si un disco CD-ROM se inserta en la unidad de CD-ROM. Se montará el disco y el administrador de archivos abrirá una ventana con los contenidos del CD. La Figura 10.d muestra los contenidos del CD-ROM dentro del administrador de archivos **Nautilus**.



#### 10.d Contenidos de un CD-ROM en Nautilus



Aparecerá un icono del CD en el escritorio, el cual puede utilizarse para desmontar y expulsar el CD-ROM después de usarlo. Con el botón derecho del ratón click en el icono para visualizar todas las opciones disponibles. Por ejemplo, para desmontar y expulsar el CD-ROM, se selecciona **Expulsar** desde el menú.

##### 10.2.2. Uso de CD-ROMs desde el indicador de comandos de shell

Se puede montar y desmontar de forma manual sus CD-ROMs desde un indicador de comandos del shell. Se introduce el CD en una unidad de CD-ROM, se abre un indicador de comandos del shell y se escribe el siguiente comando:

```
mount /mnt/cdrom
```

El CD-ROM estará ahora montado y disponible para usarse con el administrador de archivos. Puede accederse al CD-ROM haciendo click sobre el icono en el escritorio y escribiendo /mnt/cdrom en la barra de dirección.

Después de trabajar con el CD, debe desmontarse antes de expulsarlo de la unidad de CD-ROM. Debe cerrarse cualquier aplicación o los administradores de archivos que estén utilizando el CD-ROM y se escribe lo siguiente en el indicador de comandos del shell:

```
umount /mnt/cdrom
```

Ahora puede presionarse el botón de expulsión de la unidad de CD-ROM para recuperar el CD con seguridad.

## 10.4. Recursos adicionales

### 10.4.1. Documentación instalada

- Página del manual de `cdrecord` — Se describe cómo quemar discos CD-ROM de datos, audio y modo mezclado. Ofrece todas las opciones y los comandos de forma detallada, incluidos algunos comandos de ejemplo para las tareas de quemar CD-R(W) más comunes.
- `/usr/share/doc/cdrecord-<version>` (donde `<version>` es la versión de `cdrecord` instalada en su sistema) — Se incluyen muchos archivos de documentación con información sobre su uso y sobre la licencia.
- Página del manual de `mkisofs` — Detalles completos sobre la utilidad, incluyendo algunas advertencias sobre ciertos tipos de imágenes ISO. Ofrece todas las opciones en detalle, incluyendo algunos comandos de ejemplo para la creación de archivos de imagen ISO comunes.
- `/usr/share/doc/mkisofs-<version>` (donde `<version>` es la versión instalada de `mkisofs` en su sistema) — Varios archivos de documentación son incluidos con información de uso general y licenciamiento.
- `/usr/share/doc/xcdroast-<version>` (donde `<version>` es la versión de **X-CD-Roast** instalada en su sistema) — Contiene opciones de línea de comando muy útiles e información sobre el uso de esta aplicación gráfica.
- `/usr/share/doc/dvdrrecord-<version>` (donde `<version>` es la versión de `dvdrrecord` instalada en su sistema) — Para usuarios que tienen dispositivos DVD-R(+W), este grupo de documentación le ayudará a iniciarse en la creación de DVD-ROMs originales para respaldo de datos y presentaciones multimedia.

#### 10.4.2. Sitios Web útiles

- **<http://www.xcdroast.org/>** — El sitio oficial del proyecto **X-CD-Roast**.
- **<http://freshmeat.net/projects/cdrecord/>** — La página del proyecto cdrecord en Freshmeat se actualiza periódicamente con las versiones más recientes, noticias y comentarios de los usuarios.
- **<http://www.freesoftware.fsf.org/dvdrtools/>** — El sitio oficial del proyecto dvdrtools, lo que incluye la utilidad dvdrecord para la escritura de discos DVD-R(+W).

## 11. Manipulación del Sistema de Archivos

La parte más importante de cualquier sistema de cómputo son los datos, es decir, la información que los programas almacenan y manipulan. Linux tiene un sistema de archivos cuyo trabajo es conservar toda la información que se almacene en la computadora, incluyendo programas, documentos, bases de datos, textos, etcétera.

En Linux el término "archivo" se refiere a cualquier fuente de entrada o destino de salida, no sólo a un depósito de datos. El espacio visible a los usuarios se basa en una estructura de árbol, con la raíz en lo alto. Los archivos y los directorios se colocan debajo de la raíz.

En realidad, algunos de los directorios en el árbol de archivos están físicamente ubicados en diferentes particiones del disco, sobre diferentes discos y eventualmente en diferentes computadoras. Cuando una de estas particiones del disco está ligado con el árbol de archivos en el directorio conocido como punto de montaje (mount point); al punto de montaje y a todos los directorios inferiores se les conoce como el **sistema de archivos**.

Linux tiene ocho tipos de archivos y hablaremos de 3: los ordinarios, los directorios y los especiales. Los archivos ordinarios contienen datos y se almacenan en un disco. Un directorio se almacena en disco y contiene información que se usa para organizar y permitir el acceso a otros archivos.

Los archivos especiales o de dispositivo, son la representación interna de un dispositivo físico. Para poder montar el sistema de archivos en Linux, primero debemos tener una partición física en el disco, un CD-ROM, o un floppy.

### 11.1 Montaje y desmontaje

#### 11.1.1 mount

Linux usa el comando 'mount' para montar el sistema de archivos. La sintaxis es:

```
$ /bin/mount [opciones] dispositivo punto-de-montaje
```

Donde opciones son las banderas que soporta el comando mount, dispositivo es el nombre del archivo de dispositivos (device file) de modo bloque que se desea montar y punto-de-montaje es el directorio donde se montará el sistema de archivos. Cabe mencionar que el punto-de-montaje es un sub-directorio que debe estar vacío, si éste sub-directorio tiene alguna información; ésta quedará oculta cuando se monte el sistema de archivos. El comando tiene varias banderas, estos son:

-f	Termina todos los procesos excepto la actual llamada al sistema.
-v	mount provee información adicional sobre lo que se trata de hacer.



-w	El sistema de archivos se montará con permisos de lectura y escritura.
-r	El sistema se montará con permisos de solo lectura.
-n	Se monta sin escribir las entradas en el archivo /etc/mtab
-t tipo	Especifica el tipo de sistema de archivos a instalar. Los tipos validos son: minix, ext, ext2, ext3, xiafs, msdos, hpfs, proc, nfs, umsdos, sysv e iso9660.
-a	Intenta montar todo el sistema de archivos, descritos en /etc/fstab
-o lista de opciones	El argumento -o le dice a mount que aplique las opciones que son especificadas para el sistema de archivos que se montará.

### 11.1.2 Montaje automático

Puede conseguirse que este tipo de montajes se realicen automáticamente cuando el sistema entra a modo multiusuario, editando el archivo de configuración llamado: /etc/fstab. Un archivo común es como el siguiente:

```
#
# /etc/fstab
# <device> <mountpoint> <filesystemtype> <options> <dump> <fsckorder>
/dev/hda2      /          ext2          defaults      1 1
/dev/hdb1      /disco     ext2          defaults      0 0
/dev/fd0       /mnt/floppy ext2          defaults,users,noauto 0 0
/dev/hda1      none       msdos         defaults      0 0
/proc          /proc      proc          defaults
/dev/hda3      none       swap          sw
cronos:/reas   /users     nfs           rw
brahm:/alum1   /users/alum1 nfs           rw
brahm:/alum2   /users/alum2 nfs           rw
```

### 11.1.3 umount

El proceso contrario de montar es desmontar un sistema de archivos. Al igual que con el comando 'mount' tenemos tres distintas formas muy usuales del comando 'umount', estas son:

```
$ /bin/umount [opciones] dispositivo punto-de-montaje
```

donde dispositivo es el nombre del dispositivo físico a desmontar y punto-de-montaje es el nombre del directorio donde fue montado. Solamente se necesita especificar una u otra opción. El comando tiene dos parámetros adicionales: -a desmonta todo el sistema de archivos, y -t fs-tipo actúa sólo en los sistemas de archivos especificado por fs-tipo.

## 11.2 fdisk

Eventualmente será necesario conectar un nuevo disco duro a la computadora. Al hacerlo, se necesitará hacer la partición del disco para poder montarlo. La partición se realiza con el comando 'fdisk'.

```
$ fdisk drive
```

```
[root@perseo dan]# fdisk /dev/hda
```

The number of cylinders for this disk is set to 9729.  
There is nothing wrong with that, but this is larger than 1024,  
and could in certain setups cause problems with:

- 1) software that runs at boot time (e.g., old versions of LILO)
- 2) booting and partitioning software from other OSs  
(e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): m

Command action

- a toggle a bootable flag
- b edit bsd disklabel
- c toggle the dos compatibility flag
- d delete a partition
- l list known partition types
- m print this menu
- n add a new partition
- o create a new empty DOS partition table
- p print the partition table
- q quit without saving changes
- s create a new empty Sun disklabel
- t change a partition's system id
- u change display/entry units
- v verify the partition table
- w write table to disk and exit
- x extra functionality (experts only)

En el ejemplo anterior se observa la pantalla inicial que genera el comando. Al teclear la letra m, se despliega un menú de ayuda con las opciones disponibles.

## 11.3 Creación y verificación de un sistema de archivos

### 11.3.1 Creación

Una vez que se ha realizado una partición en el disco con el comando 'fdisk', debe crearse el sistema de archivos antes de proceder a ocupar el disco para introducir los datos. Esto puede realizarse con el comando 'mkfs'. La forma de usar el comando es la siguiente:

```
# mkfs [-V] [-t fs-tipo] [fs-opciones] sist_archivo [bloques]
```

sist_archivo	Es el nombre especial del sistema de archivos que se desea construir tal como /dev/hda1.
-V	Especificando esta opción inhibe más de una ejecución del sistema de archivos.
-v	verbose. Información extra.
-t fs-tipo	Especifica el tipo de sistema de archivo a construir. Ver el manual en línea de: fsck(8), mkfs.minix(8), mkfs.ext(8), mkfs.ext2(8), mkfs.xiafs(8). El tipo por omisión es minix.
fs-opciones	Las opciones con las que se creará el actual sistema de archivos.
-c	Verifica el dispositivo para los bloques malos antes de construir el archivo.
bloques	Especifica el número de bloques a usar por el sistema de archivos.

### 11.3.2 Verificación

En muchas ocasiones el sistema de archivos se daña. La causa más común es apagar la PC sin antes haber desmontado los sistemas de archivos. En los sistemas Linux se ofrece el comando 'fsck' (que a su vez es una liga a e2fsck) para reparar el sistema de archivos dañado. En Linux el comando tiene la siguiente sintaxis:

```
# fsck [-A] [-V] [-t fs-tipo] [-a] [-l] [-r] [-s] sis_arch
```

-A	Va a través del archivo /etc/fstab y trata de verificar todo el sistema de archivos en una pasada.
-V	Imprime información adicional acerca de lo que 'fsck' va haciendo.
-t fs-tipo	Especifica el tipo de sistema de archivo a verificar.
-a	Automáticamente repara cualquier problema que encuentra en el sistema de archivos sin preguntar. Use esta opción con cuidado
-l	Lista todos los nombres de archivos en el sistema de archivos.
-r	Pregunta la confirmación antes de reparar el sistema de archivos.
-s	Lista el superblock antes de verificar el sistema de archivos.
sis_arch	Especifica el sistema de archivos a ser verificado.

El programa e2fsck está diseñado para correr rápidamente si es posible. Desde el chequeo del sistema de archivos tienden que ser un disco atado, esto fue acabado para optimizar los algoritmos usando el e2fsck así esta estructurando el sistema de archivos no son repetidos accediendo desde el disco. En suma, el orden en el cual los i-nodos y los directorios son verificados, ordenados por el número de bloques reducidos al

importar el tiempo de discos buscados. Algunas de estas ideas son originalmente exploradas por [ Bina y Emrath 1989 ] por esto tiene que ser refinados por los autores.

**Paso 1 :** Toma la longitud de tiempo a ejecutar, desde que todos los i-nodos son leídos dentro de la memoria y verificados. Reduce el tiempo de I/O necesario en futuros pases. La información del sistema de archivos crítica se intercambia en memoria. El mas importante ejemplo de esta técnica es la localización en el disco de todos los bloques en el sistema de archivos. Esto obviamente necesita la re-lectura de la estructura de los i-nodos de directorios durante el paso 2.

**Paso 2 :** Revisa directorios como objetos sin conexión. Desde el directorio de entrada resuelve bloques de disco. Cada bloque de directorio puede verificarse individualmente sin referencia a otros bloques de directorios. Esto permite que el e2fsck ordene todos los bloques de directorios por el número de bloques y revise el directorio de bloques en orden ascendente. Así decrece la búsqueda en el disco. Los bloques de directorios son verificados para asegurar las entradas de directorios que son válidos y contienen referencia a los números de los i-nodos que están en uso (como se determinó en el paso 1). Desde el primer directorio de bloque en cada i-nodo del directorio, el "." y ".." de entrada son verificados asegurando que este existe y el número del i-nodo de la entrada del "." observando en el directorio actual (el número del i-nodo de la entrada ".." no es verificada hasta el paso 3). El paso 2 también verifica la información concerniente al directorio padre en cual cada directorio es ligado. Si un directorio es referenciado por más que en un directorio, la segunda referencia del directorio es negociado como una liga dura ilegal y es removida. Los pasos restantes del e2fsck están altamente ligados al CPU y toman un rango del 5-10% tiempo total de ejecución.

**Paso 3 :** La conectividad del directorio es verificado. El e2fsck recorre cada directorio a partir de la raíz, usando la información fue intercambiada durante el paso 2. En este momento, la entrada ".." de cada directorio es también verificada para asegurar que es válida. Los directorios que no pueden ser ligados hasta la raíz son ligados al directorio /lost+found.

**Paso 4 :** e2fsck revisa la referencia la cuenta de todos los i-nodos, por interacción sobre todos los i-nodos y compara las ligas contadas. Todos los archivos sin borrar con un contador de liga cero son también ligados al directorio /lost+found durante este paso.

**Paso 5 :** e2fsck verifica la validación del sumario de la información del sistema de archivos. Compara el bloque y el bitmap del i-nodo que fue construido durante los pasos previos contra los bitmaps en el sistema de archivos, y corrige las copias en disco si es necesario.

## 12. Espacio Swap

### 12.1 ¿Qué es el espacio Swap?

El Espacio swap en Linux es usado cuando la cantidad de memoria física (RAM) está llena. Si el sistema necesita más recursos de memoria y la memoria física está llena, las páginas inactivas de la memoria se mueven al espacio swap. Mientras que el espacio swap puede ser de ayuda para las máquinas con poca memoria RAM, no debería considerarse como algo que pueda sustituir a más RAM. El espacio Swap se encuentra en discos duros, que tienen un tiempo de acceso más lento que la memoria física.

El espacio Swap puede ser una partición swap dedicada (recomendable), un archivo swap o una combinación de particiones y archivos swap. El tamaño de su espacio swap debería ser igual o dos veces mayor que la memoria RAM de su ordenador, o 32 MB, la cantidad que sea más grande de estas dos, pero no más de 2048 MB (o 2 GB).

### 12.2 Añadir el espacio Swap

A veces es necesario añadir más espacio swap después de la instalación. Por ejemplo, puede actualizar la cantidad de RAM en su sistema de 64 MB a 128 MB, pero hay tan sólo 128 MB de espacio swap. Sería conveniente aumentar la cantidad de espacio swap hasta 256 MB sobre todo si lleva a cabo operaciones de uso intensivo de memoria o si ejecuta aplicaciones que requieran gran cantidad de memoria.

Tiene dos opciones: añadir una partición swap o un archivo swap. Se recomienda que añada una partición swap, pero a veces no resulta fácil si no cuenta con espacio libre disponible.

Para añadir una partición swap ( asumiendo que `/dev/hdb2` es la partición que quiere agregar):

1. El disco duro no puede estar en uso (no puede tener particiones montadas, y no se puede tener activado el espacio swap). El modo más fácil para lograr esto es iniciar su sistema de nuevo en modo de rescate. Cuando le pida montar el sistema de archivos, seleccione **Skip**. Por otro lado, si la unidad no contiene ninguna partición en uso, puede desmontarlas y eliminar todo el espacio swap del disco duro con el comando `swapoff`.
2. Cree la partición swap usando `parted` o `fdisk`. Usar `parted` es más fácil que `fdisk`; por esto es que sólo se explica el uso de `parted`. Para crear una partición swap con `parted`:
  - o En el intérprete de comandos del shell, como usuario `root`, escriba el comando `parted /dev/hdb`, donde `/dev/hdb` es el nombre del dispositivo para el disco duro con espacio libre.

- En el prompt de (parted), escriba **print** para ver las particiones existentes y la cantidad de espacio disponible. Los valores de comienzo y fin están en megabytes. Determine cuánto espacio libre hay en el disco duro y cuánto quiere dedicar a la nueva partición swap.
- En el indicador (parted), escriba **mkpartfs tipo-particion linux-swap inicio fin**, donde *tipo-particion* es primaria, extendida, o lógica, *inicio* es el punto de comienzo de la partición, y *fin* es el punto donde termina la partición.



#### Aviso

Los cambios tomarán efecto de inmediato. Tenga cuidado con lo que escribe.

- Salga de parted escribiendo **quit**.
3. Ahora que tiene la partición swap, use el comando `mkswap` para configurar la partición swap. En el indicador de comandos shell como root, escriba lo siguiente:

```
mkswap /dev/hdb2
```

4. Para activar la partición swap inmediatamente, escriba el comando siguiente:

```
swapon /dev/hdb2
```

5. Para activarlo cuando se arranca, edite `/etc/fstab` para incluir:

```
/dev/hdb2      swap          swap defaults    0 0
```

6. La próxima vez que se arranque el sistema, activará la nueva partición swap.
7. Después de añadir la nueva partición swap y de haberla activado, asegúrese de que está activa visualizando el resultado del comando `cat /proc/swaps` o `free`.

Para añadir un archivo swap:

1. Determine el tamaño del nuevo archivo swap y multiplique por 1024 para determinar el tamaño de bloque. Por ejemplo, el tamaño de bloque de un archivo swap de 64 MB es 65536.
2. En un indicador de comandos shell como root, escriba el siguiente comando con `count` lo que equivale al tamaño de bloque deseado:

```
dd if=/dev/zero of=/swapfile bs=1024 count=65536
```

3. Configure el archivo swap con el comando:

```
mkswap /swapfile
```

4. Para activar el archivo swap inmediatamente pero no automáticamente cuando se arranca:

```
swapon /swapfile
```

5. Para activarlo cuando se arranca, edite /etc/fstab para incluir:

```
/swapfile          swap          swap    defaults    0 0
```

6. La próxima vez que se arranque el sistema, se activará el nuevo archivo swap.
7. Después de haber añadido el nuevo archivo swap y de haberlo activado, asegúrese de que está activado visualizando el resultado del comando `cat /proc/swaps` o `free`.

### 12.3 Eliminar el espacio Swap

Para eliminar una partición swap:

1. El disco duro no puede estar en uso (no puede haber particiones montadas, y el espacio swap no puede estar activado). El modo más fácil para lograr esto es arrancar el sistema en modo de rescate. Cuando se le pida que monte el sistema de archivos, seleccione **Skip**.

Por otro lado, si la unidad no contiene ninguna partición en uso, puede desmontarlas y eliminar todo el espacio swap del disco duro con el comando `swapoff`.

2. En un indicador de comandos shell como `root`, ejecute el comando siguiente para asegurarse de que la partición swap está desactivada (donde `/dev/hdb2` es la partición swap):

```
swapoff /dev/hdb2
```

3. Elimine su entrada desde `/etc/fstab`.
4. Elimine la partición usando `parted` o `fdisk`. Aquí sólo se va a discutir `parted`. Para eliminar la partición usando `parted`, haga lo siguiente:

- En un indicador de comandos shell como root, escriba el comando `parted /dev/hdb`, donde `/dev/hdb` es el nombre del dispositivo para el disco duro con la partición swap a ser eliminada.
- En el indicador (`parted`), escriba **print** para visualizar las particiones existentes y determine el número menor de la partición swap que desea borrar.
- En el indicador (`parted`), escriba **rm MINOR**, donde *MINOR* es el número menor de la partición a eliminar.



#### **Aviso**

Los cambios se efectúan inmediatamente; debe escribir el número menor correcto.

- Escriba **quit** para salir de `parted`.

Para eliminar un archivo swap:

1. En un indicador de comandos shell como usuario root, ejecute el comando siguiente para desactivar el archivo swap (donde `/swapfile` es el archivo swap):

```
swapoff /swapfile
```

2. Elimine su entrada de `/etc/fstab`.
3. Elimine el archivo actual:

```
rm /swapfile
```

## **12.4 Mover el espacio Swap**

Para mover el espacio swap de un emplazamiento a otro, siga los pasos para eliminar el espacio swap y a continuación los pasos para añadir el espacio swap.



## 13. Sistema de archivos ext3

Con la versión Red Hat Linux 7.2, el sistema de archivos por defecto cambia del formato ext2 al sistema de archivos journaling ext3.

### 13.1 Características de ext3

Básicamente, el sistema de archivos ext3 es una versión mejorada de ext2. Las mejoras introducidas proporcionan las siguientes ventajas:

#### Disponibilidad

Tras un corte eléctrico o una caída inesperada del sistema (también se denomina *cierre no limpio del sistema*), se debe comprobar con el programa `e2fsck` cada sistema de archivos ext2 montado en la máquina para ver si es consistente. El proceso de comprobación lleva mucho tiempo y puede prolongar el tiempo de arranque del sistema de un modo significativo, especialmente si hay grandes volúmenes que contienen un elevado número de archivos. Durante este proceso, no se puede acceder a los datos de los volúmenes.

Con la característica journaling del sistema de archivos ext3 ya no es necesario realizar este tipo de comprobación en el sistema de archivos después de un cierre no limpio del sistema. En el sistema ext3, únicamente se realiza una comprobación de consistencia en los casos puntuales en los que se producen determinados errores de hardware, como, por ejemplo, fallos en el disco duro. El tiempo empleado para recuperar un sistema de archivos ext3 tras un cierre no limpio del sistema no depende del tamaño del sistema de archivos ni del número de archivos, sino del tamaño del *journal* (diario), utilizado para mantener la consistencia en el sistema. Por defecto, la recuperación del tamaño del "journal" tarda alrededor de un segundo, según la velocidad del hardware.

#### Integridad de los datos

El sistema de archivos ext3 proporciona una integridad superior de los datos si se produce un cierre no limpio del sistema. El sistema de archivos ext3 le permite seleccionar el tipo y el nivel de protección de los datos. Por defecto, Red Hat Linux 9 configura los volúmenes ext3 para que el nivel de consistencia de los datos sea elevado en relación con el estado del sistema de archivos.

#### Velocidad

El sistema de archivos ext3, aparte de permitir escribir datos más de una vez, en la mayoría de los casos tiene un rendimiento superior al que proporciona ext2 porque los "journals" de ext3 optimizan el movimiento de los cabezales de los discos duros. Se pueden seleccionar tres modos de journaling para optimizar

la velocidad, pero, como contrapartida, la integridad de los datos se verá afectada.

### Fácil transición

La migración de ext2 a ext3 es muy sencilla y se pueden aprovechar las ventajas de un sólido sistema de archivos con journaling sin tener que volver a dar formato al sistema. Consulte la [Sección 1.3](#) para más información sobre como realizar esta tarea.

Si realiza una instalación nueva de Red Hat Linux 9, el sistema de archivos por defecto que se asigna a las particiones Linux del sistema es ext3. Si realiza una actualización a partir de una versión de Red Hat Linux con particiones ext2, el programa de instalación le permitirá convertir estas particiones a ext3 sin perder los datos. Consulte el apéndice titulado *Actualización del sistema actual* en el *Manual de instalación de Red Hat Linux* para obtener más detalles.

## 13.2 Creación de un sistema de archivos ext3

A menudo es necesario, después de la instalación, crear un nuevo sistema de archivos ext3. Por ejemplo, si añade un nuevo disco duro al sistema Red Hat Linux puede desear particionar el disco duro y usar el sistema de archivos ext3.

Los pasos para crear un sistema de archivos ext3 son los siguientes:

1. Cree la partición utilizando parted o fdisk.
2. Dé formato a la partición con el sistema de archivos ext3 usando mkfs.
3. Etiquete la partición usando e2label.
4. Cree el punto de montaje.
5. Añada la partición a /etc/fstab.

## 13.3 Conversión a un sistema de archivos ext3

El programa tune2fs permite añadir un journal a un sistema de archivos ext2 existente sin modificar los datos en la partición. Si el sistema de archivos ya está montado mientras se realiza la migración, el journal estará visible como .journal en el directorio raíz del sistema de archivos. Si el sistema de archivos no está montado, el journal se ocultará y no aparecerá en el sistema de archivos.

Para convertir un sistema de archivos ext2 a ext3, conéctese como root y escriba:

```
/sbin/tune2fs -j /dev/hdbX
```

En el comando anterior, reemplace `/dev/hdb` con el nombre del dispositivo y `X` con el número de partición. Una vez realizado esto, asegúrese de cambiar el tipo de partición de ext2 a ext3 en `/etc/fstab`.

Si está migrando el sistema de archivos raíz, tendrá que usar una imagen `initrd` (o disco RAM) para arrancar. Para crear una, ejecute el programa `mkinitrd`. Para obtener más información sobre el uso del comando `mkinitrd`, escriba `man mkinitrd`. Asegúrese también de que la configuración LILO o GRUB carga el archivo `initrd`.

Aunque no consiga realizar este cambio, el sistema se arrancará, pero el sistema de archivos se montará como ext2 en vez de como ext3.

### 13.4 Volver al sistema de archivos ext2

Puesto que ext3 es relativamente nuevo, algunas utilidades de disco todavía no son compatibles con este sistema. Por ejemplo, tal vez deba reducir el tamaño de una partición con `resize2fs`, que todavía no es compatible con ext3. En estos casos, deberá volver temporalmente al sistema de archivos ext2. Para revertir una partición, primero deberá desmontar la partición conectándose como root y escribiendo:

```
umount /dev/hdbX
```

En el comando anterior, sustituya `/dev/hdb` por el nombre del dispositivo y `X` con el número de la partición. En el resto de esta sección, los comandos de ejemplo utilizarán `hdb1` para estos valores.

A continuación, cambie el tipo del sistema de archivos a ext2. Para ello, escriba el comando siguiente como root:

```
/sbin/tune2fs -O ^has_journal /dev/hdb1
```

Compruebe si la partición tiene errores. Para ello, escriba el comando siguiente como root:

```
/sbin/e2fsck -y /dev/hdb1
```

A continuación, vuelva a montar la partición como sistema de archivos ext2. Para ello, escriba:

```
mount -t ext2 /dev/hdb1 /mount/point
```

En el comando anterior, sustituya `/mount/point` por el punto de montaje de la partición.

Luego, quite el archivo `.journal` del nivel root de la partición cambiando el directorio donde está montado y escribiendo:

```
rm -f .journal
```

Ahora tendrá una partición `ext2`.

Si cambia definitivamente la partición a `ext2`, recuerde que debe actualizar el archivo `/etc/fstab`.

## 14. Control de Procesos

### 14.1 ps

Descripción: Este comando se utiliza para ver en un instante dado, cuál es el estado de los procesos que se están ejecutando en el sistema. Si no se especifica ninguna opción da información sobre los procesos del intérprete de comandos en el que se ejecuta. Este comando es muy útil para averiguar el pid de procesos que se quieren matar con el comando kill. La información que facilita el comando ps sobre los procesos, se presenta en formato de columnas y dependiendo de las opciones nos muestra la siguiente información, en sus correspondientes columnas. Sintaxis:

ps [opciones]

#### 14.1.1 Leyenda

ADDR	Dirección de memoria del proceso.
F	Flags del proceso
1	Proceso trazado.
2	El proceso está asignado al procesador.
3	El proceso no está asignado.
PID	identificador del proceso.
PPID	identificador del proceso padre que lo creó.
PRI	Prioridad del proceso.
SIZE	Espacio en número de páginas que ocupa el proceso en memoria.
RSS	Tamaño del proceso en kilobytes que está realmente en memoria en este instante.
WCHAN	Evento sobre el que el proceso está esperando.
-	El proceso no existe.
S	Durmiendo.
W	En espera de algún evento.
R	En ejecución.
I	Recién creado.
Z	Zombi.
T	Trazado.
TTY	Terminal del proceso
PAGEIN	Cantidad de faltas de página que ha causado.
TIME	Tiempo de CPU consumido por el proceso.
UID	identificador del usuario que lanzó el proceso.
SWAP	Cantidad de kilobytes del proceso en la memoria SWAP

#### 14.1.2 Opciones

-a	Muestra los procesos de otros usuarios.
-l	Listado largo de los procesos.
-u	Listado con información relacionada con el usuario

-v	Muestra los procesos con la información sobre la memoria virtual.
-m	Muestra los procesos con la información sobre la memoria.
-ttyxx	Muestra los procesos asociados al terminal xx.
-S	Muestra información sobre tiempos de CPU y faltas de página.
-s	Muestra los procesos con la información de las señales.
-r	Muestra sólo los procesos en ejecución.
-w	Muestra en formato ancho.
-x	Muestra los procesos sin terminal asociado.

Ejemplos:

\$ ps

Muestra los procesos del intérprete de comandos donde se ejecuta el comando ps.

\$ ps -aux

Muestra los procesos en formato de usuario, tanto los propios como los de otros usuarios así como los procesos sin terminal asociado.

#### 14.2 kill

Descripción: Comando utilizado generalmente para detener procesos que se están ejecutando, comúnmente se llama matar a un proceso a esta acción. En general, la acción que se realiza con el comando kill es mandar señales a los procesos en ejecución. Sólo podemos matar al proceso si se es el usuario que lanzó el proceso o el usuario roca. Sintaxis:

\$kill [-señal] pid

Opciones:

<b>-signal</b>	Señal que se quiere enviar a proceso, generalmente es la señal SIGKILL o señal 9, que es la que mata al proceso sin remisión.
<b>pid</b>	Número identificador del proceso al que se pretende mandar la señal.
<b>-l</b>	Muestra una lista con las señales que podemos mandar a los procesos.

Ejemplo:

\$kill -9 425

Eliminará al proceso cuyo pid es el 425, sólo podemos matar al proceso si se es el usuario que lanzó el proceso o el usuario root.

## 15. Utilerías

### 15.1 Respaldo

#### 15.1.1 tar

Descripción: La función fundamental de este comando es la de realizar copias de archivos hacia un dispositivo de cinta. Pero también se utiliza el comando para generar copias de seguridad de varios archivos en uno solo. Muchos paquetes y programas vienen en archivos tar por su comodidad de distribución. Sintaxis:

\$ tar acción [opciones] archivos

Opciones:

<b>c</b>	Crea un nuevo archivo .tar
<b>r</b>	Copia los archivos al final del archivo Lar indicado.
<b>t</b>	Muestra la tabla de contenidos del archivo de cinta especificado.
<b>u</b>	Actualiza el archivo Lar con los especificados, los nuevos los pone al final del archivo y los que ya están sólo se sustituyen si han sido modificados o actualizados.
<b>x</b>	Extrae del archivo Lar los archivos indicados.
<b>f</b>	Especifica el nombre del archivo destino del archivo Lar que se va a crear. Puede ser un archivo normal o un archivo especial de dispositivo como una cinta.
<b>v</b>	Muestra el nombre de los archivos conforme se van procesando
<b>w</b>	Modo interactivo hace que Lar pregunte antes de realizar cualquier acción.
<b>F archivo</b>	Se especifican los archivos a procesar en el archivo de texto archivo.
<b>z</b>	Pasa el archivo de cinta por el comando gzip antes que por tar, en el caso de extraer, para descomprimir el archivo Lar previamente comprimido con gzip. Para la creación, después de procesar con Lar pasa el resultado por gzip para comprimir. Generalmente estos archivos toman la extensión .tgz
<b>archivos</b>	Lista de archivos que se van a procesar m, pueden ser directorios que se procesarán recursivamente.

Ejemplos:

\$ tar zxvf prueba.tgz

Extrae el contenido del archivo prueba.tgz, que está comprimido con gzip después de crear el tar, en el directorio actual, mantiene la estructura de directorios original.

\$ tar cvf backup.tar \*

Crea el archivo backup.tar con todos los archivos del directorio actual.

### 15.1.2 gzip

Descripción: El programa gzip es un compresor descompresor de archivos, reduce el tamaño de los archivos mediante un algoritmo de compresión. gzip genera archivos con extensión .gz. por omisión. Si se comprime un archivo gzip, lo reemplaza por otro ya comprimido manteniendo sus permisos y derechos. Este comando tiene su complementario, aunque realmente no hace falta, que es el comando gunzip que descomprime un archivo. Pero gzip es capaz de devolver los archivos a su estado original. Generalmente este comando se usa en combinación con el comando tar. Sintaxis:

gzip [opciones] archivos

Opciones:

<b>-c</b>	Comprime pero dejando intactas los archivos originales, así tendremos tanto la copia comprimida como la original.
<b>-d</b>	Descomprime.
<b>-f</b>	Fuerza la compresión sin preguntar antes de borrar o sobrescribir.
<b>-l</b>	lista los siguientes campos, de un fichero comprimido.
<b>compressed</b>	Tamaño del fichero comprimido.
<b>uncompr</b>	Tamaño del fichero descomprimido.
<b>ratio</b>	Relación de compresión (0.0% si no se conoce).
<b>uncompressed_name</b>	Nombre del fichero descomprimido.
<b>-q</b>	Suprime los avisos por errores.
<b>-r</b>	Recorre la estructura de directorios recursivamente. Si algún archivo de los facilitados es un directorio con esta opción lo recorre recursivamente comprimiendo todos sus archivos.



<b>-t</b>	Comprueba la integridad del fichero comprimido.
<b>archivos</b>	Lista de archivos a procesar, se permiten los caracteres comodín.

Ejemplos:

`gzip *`

Comprime todos los archivos del directorio actual.

`gzip -d documento.gz`

Descomprime el archivo documento.gz dejándolo en su estado original

`gzip -r /tmp`

Comprime todos los archivos del directorio /tmp.

## 15.2 Comunicación remota

La ventaja principal que puede obtener de su computadora con el trabajo en red es la capacidad de compartir recursos e información, así como la posibilidad de acceder a dicha información desde localizaciones remotas. Linux proporciona un robusto conjunto de herramientas para ello. Si la World Wide Web permite acceder a cientos de informaciones en formato hipertexto, existen herramientas adicionales que le permitirán conectarse a computadoras remotas, transferir archivos y ejecutar comandos remotos.

### 15.2.1 Telnet

El comando Telnet es la herramienta básica para conexiones remotas bajo Linux. Con Telnet podrá mantenerse sesiones como una terminal de la computadora remota, ejecutando comandos como si se estuviera conectado localmente.

Para conectarse a una computadora vía Telnet, debe disponerse de un nombre de usuario y de una contraseña válida para la computadora remota. Aunque algunos sistemas proporcionan funciones de conexión para invitados, esto no es muy corriente por razones de seguridad. Cuando se permiten conexiones de invitados, éstas casi siempre colocan al usuario en un shell o sistema de menú sujeto a restricciones. La idea en que se basan estos entornos de invitado es proporcionar seguridad informática y proteger el sistema de usuarios maliciosos o descuidados. Un shell restringido impide que el usuario pueda ejecutar determinados comandos y un sistema de menú sólo permite seleccionar entre un conjunto previamente determinado de menús, bloqueando completamente el acceso al shell.

Telnet también permite a los usuarios conectarse a sus propias computadoras desde localizaciones remotas con sólo introducir su nombre de usuario y contraseña. Esto permite a los usuarios comprobar el correo electrónico, editar archivos y ejecutar programas en su computadora normal como si estuvieran conectados a nivel local. Una desventaja es que, para ello, debe trabajarse en un entorno basado en terminal en lugar del sistema X Windows, puesto que Telnet sólo proporciona emulación de terminal para terminales corrientes como, por ejemplo, el VT-100 de DEC, que no soporta entornos gráficos como X Windows.

La sintaxis básica de telnet es la siguiente:

\$telnet [nombre del servidor]

**nombre del servidor:** Es el nombre de una computadora remota. Si no especifica una computadora remota, Telnet se inicia en su modalidad de comando interactiva.

Por el contrario, si proporciona un nombre de servidor remoto, Telnet intenta iniciar una sesión inmediatamente. Telnet acepta varios argumentos de línea de comandos, tal y como muestra la siguiente Tabla.

Argumentos de línea de comandos del comando telnet

<b>-d</b>	Desactiva la depuración.
<b>-a</b>	Intenta una conexión automática.
<b>-n archivo_de_rastreo</b>	Activa el rastreo y guarda los datos del mismo en el archivo de <b>rastreo</b> .
<b>-e carácter_de_escape</b>	Establece carácter de escape como el caracter de escape que debe utilizarse en esta sesión. Si se omite el carácter de escape del argumento, no habrá carácter de escape para esta sesión Telnet.
<b>-l usuario</b>	Envía el nombre de usuario al sistema remoto para conexión automática. Este argumento implica la utilización del argumento -a.
<b>puerto</b>	Indica el número de puerto al que debe realizarse la conexión en el sistema remoto. Se utiliza para especificar distintos programas de red. Si no llegara a especificar, Telnet se conecta al puerto Telnet predeterminado.

### 15.3 FTP

El Protocolo de transferencia de archivos (WFP) es un método sencillo y efectivo de transferir archivos entre computadoras conectadas a una red TCP/IP FIP permite a los usuarios transferir archivos ASCII y binarios. Durante una sesión WFP, es posible conectarse a otra computadora utilizando el programa cliente FTP. A partir de este momento, podrá desplazarse por el árbol de directorios, listarse el contenido de

directorios, copiarse archivos desde la computadora remoto a su propia computadora y transferirse archivos desde la computadora al sistema remota. Los mecanismos de protección de archivos son también aplicables en este caso, por lo que no podrá obtenerse un archivo o colocarlo en el sistema remoto, si no tiene los permisos correspondientes.

Para utilizar los archivos de transferencia FTP, deberá disponerse de un nombre de usuario y contraseña válidos para la computadora remota. Esta combinación de nombre de usuario y contraseña se utiliza para validar la sesión WFP y determinar el tipo de acceso que tiene para transferir los distintos archivos. Además, deberá conocerse el nombre de la computadora con la que desea mantener una sesión FTP.

Debe tenerse en cuenta que los clientes WFP tienen conjuntos diferentes de comandos, dependiendo del sistema operativo del que se trate. Cabe la posibilidad de que los comandos que espera el sistema remoto sean diferentes, dependiendo de la versión. Sería muy raro que los sistemas FTP fueran totalmente incompatibles entre sí. Lo más probable es que los comandos sean ligeramente diferentes o que no se encuentren disponibles.

#### **15.3.1 FTP como usuario anónimo**

Debido al crecimiento explosivo de Internet, muchas organizaciones han establecido enormes bancos de información accesibles vía WFP. Estos sitios WFP contienen todo tipo de elementos, desde archivos de texto a software de cualquier naturaleza. Existe una convención estándar en Internet que permite acceso FTP como invitado a depósitos de archivos para que los usuarios puedan efectuar transferencias de archivos. Este acceso como invitado recibe el nombre de FTP como usuario anónimo. Para poder utilizar FTP como usuario anónimo, deberá iniciarse una sesión WFP en el sistema remoto y utilizar el nombre de usuario anonymous y alguna dirección de correo electrónico como contraseña.

Muchos sitios no permiten FTP como usuario anónimo por razones de seguridad. En los casos en que no se permite FTP como usuario anónimo, el comando ftp da un mensaje de error del tipo Login failed -User "anonymous" unknown. Los sitios que sí permiten FTP como usuario anónimo normalmente sitúan al usuario en un árbol de directorios sujeto a restricciones con acceso sólo de lectura. Si se permite colocar archivos en la computadora remota, normalmente sólo podrán ubicarse en un directorio.

#### **15.3.2 Comandos FTP**

El comando ftp de Linux proporciona un conjunto detallado de opciones de comando en modalidad interactiva. Como se mencionó anteriormente, algunos sistemas remotos podrían no admitir todos estos comandos, pero de todas formas probablemente no tendrá que utilizar muchos de ellos. La tabla siguiente relaciona la lista de comandos disponibles en la modalidad interactiva de WFP.

* !	Escapa al shell.
\$	Ejecuta una macro.
account	Envía el comando de cuenta al servidor remoto.
* append	Añade a un archivo.
* ascii	Establece el tipo de transferencia de archivos en la modalidad ASCII.
* bell	Emite una señal acústica cuando se completa un comando.
* binary	Ajusta el tipo de transferencias de archivos a la modalidad binaria.
* bye	Finaliza la sesión FTP y sale.
case	Conmuta mget entre asignación de nombre de archivos con mayúsculas o minúsculas.
* cd	Cambia el directorio de trabajo en la computadora remota.
cdup	Cambia el directorio de trabajo remoto al directorio matriz.
chmod	Modifica los permisos de archivo del archivo remoto.
* close	Finaliza la sesión FTP
cr	Conmuta el filtrado de retornos de carro cuando se recibe un archivo ASCII.
* delete	Borra archivos remotos.
* debug	Borra archivos remotos.
* dir	Lista el contenido del directorio remoto.
* disconnect	Finaliza la sesión FTP (igual que close).
exit	Finaliza la sesión FTP y sale.
form	Establece el formato de transferencia de archivos.
* get	Obtiene un archivo de la computadora remota.
* glob	Conmuta la expansión de comodín de nombres locales de archivo.
* hash	Conmuta la impresión del carácter para cada memoria intermedia transferida.
* help	Imprime información de ayuda local.
idle	Obtiene o fija el temporizador en la computadora remota.
image	Establece el tipo de transferencia de archivo al modo binario (lo mismo que binary).
* lcd	Cambia al directorio local de trabajo.
* ls	Lista el contenido del directorio remoto (proporciona el tamaño y el tipo de permiso asignado).
macdef	Define una macro.
* mdelete	Borra archivos múltiples en la computadora remota.
* mdir	Lista el contenido de directorios remotos múltiples.
* mget	Obtiene archivos múltiples de la computadora remota.
* mkdir	Crea un directorio en la computadora remota.
* mls	Lista el contenido de directorios remotos múltiples.
mode	Establece la modalidad de transferencia de archivos.
modtime	Muestra la hora de la última modificación del archivo remoto.
* mput	Envía archivos múltiples a la computadora remota.
newer	Obtiene el archivo remoto si éste es más reciente que el archivo local correspondiente.
nmap	Establece plantillas para asignación predeterminada de nombres de archivos.
niist	Lista el contenido del directorio remoto.

ntrans	Establece la tabla de traducción para la asignación predeterminada de nombres de archivos.
* open	Establece conexión con el sitio FTP remoto.
passive	Establece el modo de transferencia pasivo.
* prompt	Fuerza indicaciones interactivas de comandos múltiples.
proxy	Emite comandos en conexión alternativa.
* put	Envía un archivo a la computadora local.
* pwd	Imprime el directorio de trabajo en la máquina remota.
* quit	Finaliza la sesión FTP y sale.
* quote	Envía un comando ftp arbitrario.
* recv	Recibe un archivo.
reget	Obtiene un archivo volviendo a empezar al final del archivo local.
rstatus	Muestra el estado de la máquina remota.
rhelph	Obtiene ayuda del servidor remoto.
* rename	Asigna un nuevo nombre al archivo.
reset	Despeja la cola de respuestas de comandos.
restart	Reinicia la transferencia de un archivo a partir de la cuenta de bytes especificada.
* rmdir	Elimina un directorio de la máquina remota.
runique	Cuando se recuperan archivos múltiples con el mismo nombre de archivo al mismo directorio, runique asigna un nombre de archivo exclusivo a cada archivo recibido.
* send	Envía un archivo a la computadora remota.
site	Envía un comando específico de sitio al servidor remoto, uno entre umask, idie, chmod, help, group, gpass, newer o minfo.
size	Muestra el tamaño del archivo remoto. status Muestra el estado actual.
* status	Muestra el estado actual.
struct	Establece la estructura de transferencia de archivos. system Muestra el tipo de sistema remoto.
sunique	Cuando se envían archivos múltiple con el mismo nombre de archivo al mismo directorio, sunique asigna un nombre exclusivo de archivo a cada archivo enviado.
tenex	Establece el tipo de transferencia de archivos tenex.
tick	Conmuta el tamaño del contador de byte de impresión durante las transferencias.
* trace	Conmuta el rastreo de paquetes.
* type	Establece el tipo de transferencia de archivos.
* user	Envía nueva información de usuario.
umask	Obtiene o establece el umask en la computadora remota.
* verbose	Conmuta la modalidad detallada.
* ?	Imprime información de ayuda local.

## 15.4 Uso de los comandos r-

Además de ftp y telnet, existen otros comandos que permiten acceder a computadoras remotas e intercambiar archivos en una red. Estos comandos reciben colectivamente el nombre de comandos r-.

Los comandos r- merecen una atención especial, puesto que una de sus características puede comprometer seriamente la seguridad del sistema si no se tiene cuidado. Cuando se introduce un comando r-, el sistema remoto comprueba un archivo llamado /etc/hosts.equiv para comprobar si el sistema local se encuentra dentro de la lista. Si no encuentra el sistema local, busca un archivo llamado rhosts en el directorio de usuario en la máquina remota. El comando r- comprobará entonces si el nombre de servidor local se encuentra en el archivo rhosts. Si el servidor local se encuentra en alguno de estos lugares, se ejecutará el comando sin comprobar la contraseña.

Aunque es muy cómodo no tener que teclear la contraseña cada vez que se necesita acceder a una computadora remota, está claro que también podría comprometerse seriamente la seguridad del sistema. Por ello, conviene considerar cuidadosamente las implicaciones, en términos de seguridad, que conllevan los comandos r- antes de establecer archivos /etc/hosts.equiv y .rhosts en el sistema local.

### 15.4.1 rlogin

El comando rlogin es muy parecido al comando telnet, ya que permite iniciar una sesión de comandos interactiva en un sistema remoto. La sintaxis de este comando es la siguiente:

```
rlogin [-8EKLdx] [-e char] [-k realm] [-l nombre_usuario] nombre_servidor
```

Aunque la utilización más común es sencillamente:

```
$ rlogin nombre_servidor
```

Opciones de línea de comandos para el comando rlogin:

- 8 Esta opción Permite una ruta de datos de entrada de 8 bits en todo momento, con lo que se permite el envío de caracteres ANSI formateados y otros códigos especiales. Si no se utiliza esta opción, se filtrarán los bits de paridad, salvo que los caracteres de parada y arranque sean distintos de <Ctrl-s> y <Ctrl-q>.
- E Impide el reconocimiento de cualquier carácter como carácter de escape. Proporciona una conexión completamente transparente cuando se utiliza con la opción -8.

- K Desactiva la autenticación Kerberos. Se utiliza sólo cuando se está realizando una conexión con un sistema que usa el protocolo de autenticación Kerberos.
  
- L Permite la ejecución de la sesión rlogin en modalidad litout.
  
- d Desactiva la depuración de los zócalos TCP utilizados para establecer la comunicación con el sistema remoto.
  
- e Se utiliza para establecer el carácter de escape para la sesión rlogin. El carácter de escape es ~ por defecto. Puede especificarse un carácter literal o un valor octal en el formato \nnn.
  
- k Esta opción solicita a rlogin que obtenga permisos Kerberos para el servidor remoto en el dominio especificado en lugar del dominio del servidor remoto determinado por krb\_realmofhost.
  
- l Permite la especificación del nombre remoto. Se utiliza la autenticación Kerberos, si se encuentra disponible.
  
- x Activa el encriptado DES para todos los datos pasados durante la sesión rlogin. Esto suele aumentar en el tiempo de respuesta y la utilización de la CPU, pero proporciona un mayor de seguridad.

#### **15.4.2 rsh**

El comando rsh es una abreviatura de shell remoto. Este comando inicia un shell en el sistema remoto especificado y ejecuta el comando que haya especificado en la línea de comandos rsh, si ha especificado alguno. Si no especifica ninguno, se le conecta a la máquina remota utilizando rlogin. La sintaxis de este comando es:

```
rsh [- Kdnx] [- k realm] [-l nombre -usuario] nombre _ servidor [comando]
```

Aunque la forma más común es:

```
rsh nombre-servidor [comando]
```

El argumento comando puede ser cualquier comando de Linux que pueda escribirse en el indicador del shell. Opciones de la línea de comandos para el comando rsh:

- K Desactiva la autenticación Kerberos. Se utiliza sólo cuando se está realizando una conexión con un sistema que utiliza Kerberos.
- d Desactiva la depuración de los zócalos TCP utilizados para establecer la comunicación con el sistema remoto.
- k Esta opción solicita a rsh que obtenga tickets Kerberos para el sistema remoto en el dominio especificado, en lugar del dominio del sistema remoto determinado por krb\_realmofhost.
- l Permite la especificación del nombre remoto. Se utiliza la autenticación Kerberos, si se encuentra disponible, y la autorización se determina como en el comando rlogin.
- n Redirige entradas desde el dispositivo especial /dev/null.
- x Activa el encriptado DES para todos los datos transferidos. Podría influir en el tiempo de respuesta y la utilización de la CPU, pero proporciona mayor nivel de seguridad.

Linux lleva la entrada estándar al comando rsh y la copia en la entrada estándar del comando ejecutado remotamente. Después copia la salida estándar del comando remoto a la salida estándar para rsh. También copia el error estándar remoto al descriptor local del archivo de error estándar. Las señales de salida, finalización e interrupción se envían a la computadora remota. De igual forma, cualquier carácter especial de shell que no vaya entrecomillado, como en ">>", se manipula localmente. Si van entre comillas, estos caracteres se manipulan por medio del comando remoto.

### 15.4.3 rcp

El comando rcp, que significa copia remota, es el último de los comandos r- que necesita examinarse. Se utiliza para copiar archivos entre computadoras. rcp puede utilizarse para copiar archivos desde una computadora remota a otra, sin que la máquina local sea la fuente o el destino. El comando rcp tiene dos formatos. El primero se utiliza para copiar un archivo a otro archivo. El segundo se utiliza cuando se copian archivos o un directorio a otro directorio. La sintaxis del comando rcp es:

```
rcp [-px] [-k realm] nombre-archivo1 nombre-archivo2
rcp [-px] [-r] [-k realm] directorio archivo(s)
```

Cada argumento de archivo o directorio es un nombre de archivo remoto o local. Los nombres de archivos remotos tienen el formato rname@rhost:path, donde rname es el nombre del usuario remoto, rhost es la computadora remota y path es la ruta para acceder al archivo. El nombre de archivo debe contener dos puntos. Argumentos de línea de comandos para el comando rcp:

- r Copia de forma recursiva el árbol del directorio fuente al directorio de destino.



Observe que el destino debe ser un directorio para poder utilizar esta opción.

- p Trata de conservar los tiempos y la modalidades de los archivos fuente, ignorando umask.
- k Esta opción solicita a rcp la obtención de tickets Kerberos para el sistema remoto en el dominio especificado, en lugar de en el dominio del sistema remoto determinado por krb\_realmofhost(3).
- x La opción -x activa el encriptado DES para todos los datos transferidos por rcp. Esto podría influir en el tiempo de respuesta y en la utilización de la CPU, pero proporciona mayor nivel de seguridad.

Si la ruta especificada en el nombre de archivo no está completa, se interpretará como perteneciente al directorio de conexión del usuario especificado en la computadora remota. Se utilizará el nombre actual de usuario si no se proporciona nombre de usuario remoto. Si una ruta en un servidor remoto contiene caracteres especiales de shell, puede indicarse utilizando \, " o ' según corresponda. Esto hace que todos los meta caracteres del shell se interpreten remotamente.

NOTA: rcp no solicita la introducción de contraseñas. Realiza las copias utilizando el comando rsh.

## 15.5 SSH

SSH (abreviatura de la expresión inglesa "secure shell" o shell seguro), como el comando rsh, es un programa que permite conectar con la máquina remota y ejecutar comandos en dicha máquina remota. SSH está diseñado para reemplazar rsh y rlogin ya que permite definir una sesión encriptada entre sistemas poco fiables en una red insegura. Uno de los problemas que plantea telnet es que, cuando el usuario se conecta al sistema remoto, la contraseña se envía como texto ASCII a la red. De esta forma, si otro usuario comprueba los paquetes Ethernet, puede detectar el nombre de conexión y contraseña del usuario original para el sistema remoto. SSH evita este tipo de problemas al utilizar una autenticación basada en RSA. Debido a sus prestaciones en materia de seguridad, los administradores de sistemas utilizan SSH de forma habitual. Los clientes de SSH también pueden utilizarlos otros sistemas operativos, como Macintosh y Windows.

El comando SSH es muy parecido al comando telnet ya que permite iniciar una sesión de comandos interactiva en un sistema remoto. SSH tiene la sintaxis siguiente:

```
SSH [-a] [-c idea | blowfish | des | 3des | arcfour | tss | none] [-e escape_char] [-I identity file] [-l login_name] [-n] [-k] [-V] [=o option] [-p port] [-q] [-P] [-t] [-v] [-x] [-c] [-L port] host:hostport ] [-R port:host:hostport] hostname [comand]
```

Opciones de línea de comandos para el comando SSH:

- a Desactiva el envío del agente de autenticación.
- c Selecciona el cifrado que debe utilizarse para encriptar la sesión. Idea es el cifrado por defecto; arcfour es el más rápido; y none equivale a utilizar rlogin o rsh (sin encriptación).
- e Determina el carácter de escape para la sesión.
- f Establece SSH en segundo plano una vez determinados la autenticación y el envío.
- i Selecciona el archivo de identidad desde el que se lee la clave privada para la autenticación de RSA.
- k Desactiva el envío de tickets Kerberos.
- l Determina el nombre de conexión que debe utilizarse en la máquina remota.
- h Redirecciona stdin desde /dev/nulls que se utiliza cuando SSH se ejecuta en segundo plano.
- o Utilizado para las opciones definidas por el usuario, después del formato en el archivo de configuración.
- p especifica el puerto al que debe conectarse en el servidor remoto.
- q Activa el modo silencioso, que se encarga de borrar todos los mensajes salvo los que informan de errores graves.
- P Utiliza un puerto sin privilegios.
- t Fuerza la asignación pseudo-tty.
- v Activa el modo detallado (útil para depurar errores).
- x Desactiva el envío X11.
- C Requiere la compresión de todos los datos.
- L Especifica el puerto local que debe enviarse al puerto y al servidor remoto designados.
- R Especifica el puerto remoto al que deben enviarse el servidor local y el puerto designado.