Aprendiendo a Aprender Linux

Guías para colegios con plataforma de referencia S-Helio 1.1

Vladimir Támara

Jaime Irving Dávila

Pablo Chamorro

Igor Támara

Aprendiendo a Aprender Linux: Guías para colegios con plataforma de referencia S-Helio 1.1

por Vladimir Támara

por Jaime Irving Dávila

por Pablo Chamorro

por Igor Támara

Este trabajo se ha mejorado con correcciones hechas por voluntarios, si desea obtener las fuentes puede visitar en Internet: http://structio.sourceforge.net/guias/AA_Linux_colegio/. Para mantenerse informado de otros desarrollos de Structio puede suscribirse a la lista de anuncios (bajo tráfico): http://lists.sourceforge.net/lists/listinfo/structio-anuncio. Puede reportar fallas en estas guías usando la categoria S-Helio en: https://sourceforge.net/tracker/?group_id=4503&atid=104503 Si desea comunicarse con los desarrolladores de Structio, puede escribir por correo electrónico a <structio-devel@lists.sourceforge.net>

Tabla de contenidos

Agradecimientos

- A Dios por todo.
- A los desarrolladores de software de libre redistribución.
- A los desarrolladores de la distribución Debian http://www.debian.org
- A los integrantes de la comunidad educativa del colegio colombiano *Gimnasio Fidel Cano* por ser la motiviación inicial de estas guías y por permitir su implementación http://www.tamarapatino.com/gfc. Y especialmente a Igor Támara quien ha estado impulsando la implementación en este colegio.
- Por sus correcciones, gracias a: Gustavo Ospina `<gos@users.sourceforge.net>`, Melissa Giraldo `<melagira@yahoo.com>`, Jhair Tocancipá `<jhair_tocancipa@gmx.de>`, Martín Soto `<soto@iese.fhg.de>`, Juan Carlos Cubillos `<jcr@euclides.uniandes.edu.co>`, Freddie Valenzuela `<freval@gratis1.com.co>`, Isamary Garcia `<is-garci@uniandes.edu.co>`, Wilfredo Pachon `<wilfred_com@usa.net>`, Jaime Jimenez `<jajim@jajim.com>`, Juan Pablo Romero Bernal `<berome@tutopia.com>`.
- Por sus escritos y correcciones gracias a: Jaime Irving Dávila `<jadavila@uniandes.edu.co>`, Pablo Chamorro `<pchamorro@ingeomin.gov.co>`, Igor Támara `<igor@tamarapatino.com>`, Vladimir Támara `<vtamara@informatik.uni-kl.de>`.

Introducción

El principal objetivo de este documento es lograr que el lector adquiera la capacidad de aprender a usar, instalar y administrar una red de computadores con Linux en un colegio.

Para fijar un marco de referencia preciso, hemos diseñado una red de computadores apropiada para colegios, basada en el sistema operativo Linux y la utilización exclusiva de software de libre redistribución. A este diseño lo hemos llamado plataforma de referencia S-Helio 1.1. Ver Plataforma de referencia.

Para lograr el objetivo hemos escrito varios grupos de guías para diversos perfiles de usuarios, que documentan diferentes aspectos de la plataforma de referencia S-Helio 1.1. El diseño de las guías, el ambiente de aprendizaje que sugerimos y algunas recomendaciones para su uso se presentan en un ápendice (Ver Apéndice B).

El resto de este documento está organizado así: Los capítulos 1 y 2 son grupos de guías para presentar Linux a usuarios nuevos. El capítulo 3 consta de guías para usuarios avanzados que quieren aprovechar mejor un sistema Linux. Los capítulos 4,5 y 6 son grupos de guías para los administradores de la red. El cuarto capítulo es una introducción para ellos, el quinto presenta administración e instalación de un sistema Linux aislado de una red y el sexto capítulo presenta instrucciones para instalar y administrar varios computadores en una red de área local con servicios como correo electrónico, web, `DNS`, `ftp`, `ssh`, `NIS`, `NFS`.

Al igual que todos los componentes de la plataforma de referencia S-Helio 1.1 (hardware, Linux, software de libre redistribución), este documento está en constante proceso de actualización. El escrito que usted está consultando fue actualizado por última vez en Febrero de 2003, si la fecha resulta muy antigua por favor obtenga una versión más reciente en el sitio de distribución:

http://structio.sourceforge.net/guias/AA_Linux_colegio/

Las voluntarias y voluntarios que ayudamos a escribir estas guías no somos expertos en los temas tratados aunque nos hemos esforzado por investigarlos y resumirlos, sólo tenemos algo de experiencia que queremos brindar desinteresadamente. Sin mala intención pudimos haber consignado errores (de ortografía, redacción o contenido), si encuentra alguno por favor informenos —si le es posible y lo desea, sugiriendo una mejora. Si lo desea también puede completar a volver a escribir guías o porciones y liberarlas al dominio público para adjuntarlas a este escrito. Puede enviar sus comentarios y mejoras por correo electrónico a `<structio-guias@lists.sourceforge.net>` o si lo desea puede suscribirse a una lista de voluntarios, para colaborar de manera continua en el desarrollo: (http://lists.sourceforge.net/lists/listinfo/structio-guias)

Capítulo 1. Primera exploración de Linux desde la perspectiva de un usuario

Público y logros

Estudiantes, profesores y personal administrativo del colegio, quienes se acercan por primera vez como usuarios a la plataforma de referencia S-Helio 1.1 —suponemos que la plataforma de referencia ya está instalada y bien configurada.

- Conoce el espíritu de solidaridad tras Linux y el software de libre redistribución.
- Administra su espacio personal, su login y su clave.
- Configura el escritorio Gnome y se familiariza con algunas herramientas que tiene disponibles.
- Administra archivos en su espacio personal con el administrador de archivos de Gnome y en algunos casos desde un interprete de comandos.
- Emplea disquetes y CDROMs.
- Busca y consulta documentación que le permite aprender por si mismo.

1.1. Software de libre redistribución

Indicadores de Logros

- Distingue algunas licencias de software.

1.1.1. Lectura: Software de libre redistribución

Cuando se libera una obra intelectual al público, el autor [1] debe especificar los derechos que quienes reciban la obra tendrán sobre la misma, normalmente estos derechos se consignan en una licencia que acompaña a la obra y sus copias.

Un programa es una obra intelectual y por tanto debe acompañarse de una licencia cuando se libera al público. [2] Un programa es de libre redistribución si su licencia por lo menos da derecho a los usuarios para emplearlo, copiarlo y redistribuir copias.

Un buen programa es una herramienta que le facilita a algunas personas la realización de una tarea específica. Hacer un buen programa y mantener su calidad no es tarea fácil, porque un programa es como una compleja maquinaria con muchas partes que exige de sus desarrolladores:

- Estudio constante de teoría relacionada con el programa, de otros programas similares y de las necesidades de los usuarios.

1. En realidad el propietario de los derechos de reproducción es quien especifica la licencia
2. Note que en general si una obra intelectual no tiene una licencia visible, la redistribución y la copia NO son permitidas

- Disciplina para completar el programa, experimentar cambios y aplicarlos si resultan apropiados.
- Comprensión y paciencia para atender a los usuarios y sus reclamos o requerimientos, así como para comunicarse con otros desarrolladores.
- Constancia para continuar el desarrollo.

Como ejemplo podemos considerar el trabajo de los desarrolladores del "*kernel*" de Linux —el *kernel* es una de las principales partes del sistema operativo. Ellos se comunican por correo electrónico, en la lista de desarrollo "linux-kernel@vger.kernel.org", donde hay más de 3000 personas inscritas, y donde se intercambia diariamente más de 150 mensajes, con el fin de mejorar cada parte del kernel (el kernel es una parte compuesta de pequeñas partes). En esos mensajes algunos desarrolladores envían resultados de experimentos o propuestas de modificaciones para mejorar cierta parte. Las propuestas son juzgadas por otros miembros, y los experimentos son repetidos hasta que logran llegar a un acuerdo —tal vez varios días o incluso meses después para mejorar una sola de las partes.

Dada esta gran dificultad para desarrollar software de buena calidad, ¿Por qué el software de libre redistribución no exige contraprestación?

Cada desarrollador tiene sus motivaciones propias, pero tal vez algo común es el deseo de colaborar, el deseo de ayudar desinteresadamente. Esto no quiere decir que los desarrolladores de software de libre redistribución no tengan necesidades económicas. Claro que las tienen y deben buscar formas para pagar sus gastos; nosotros como usuarios podemos tomar una posición activa:

- Podemos continuar la ola de solidaridad ayudando desinteresadamente (hay muchos proyectos en los cuales trabajar, en particular podemos ayudar a desarrollar software de libre redistribución).
- Podemos donar a proyectos o personas que ayuden desinteresadamente (en particular a personas que desarrollan software de libre redistribución).

El software de libre redistribución es entonces una oportunidad para colaborar desinteresadamente.

1.1.2. Lecturas recomendadas: Software de libre redistribución

- Para conocer diversas motivaciones para lib1erar obras bajo licencias que permitean libre redistribución, puede consultar diversos puntos de vista:
  - Motivaciones del movimiento de fuentes abiertas en: http://www.opensource.org
  - Motivaciones del movimiento de software libre en: http://www.gnu.org/philosophy
  - Motivaciones para ceder obras al dominio público en: http://www.tamarapatino.com/vladimir/dominio_publico.shtml
- Para conocer más sobre el desarrollo del kernel de Linux, puede consultar el FAQ de la lista de desarrollo en: http://www.tux.org/lkml
- Si desea consultar más sobre licencias concretas de libre redistribución para programas y escritos: http://ingenieria.sanmartin.edu.co/cgi-bin/slec/slec.cgi/item.msg.view?uid=33

1.1.3. Ejercicios: Software de libre redistribución

1. ¿El software de libre redistribución puede venderse?

2. Consiga alguna(s) licencias de software y determine si permiten libre uso y libre redistribución.

3. Averigüe nombres de algunos desarrolladores de software de libre redistribución, así como mecanismos para donarles.

1.2. Linux

Indicadores de Logros

- Identifica algunos sucesos en la historia de Linux.

1.2.1. Lectura: Linux

Un sistema operativo es un programa que facilita el empleo del hardware de un computador, lo facilita presentando abstracciones de cada dispositivo. Por ejemplo al guardar información en un disquete, el usuario puede pensar en archivos, y el sistema operativo se encarga de los detalles de acomodar y organizar la información en el disquete a nivel físico. Linux es un sistema operativo tipo Unix de libre redistribución. Unix es el nombre de un sistema operativo concebido en los 70's junto con el lenguaje de programación C por Kenneth Thompson, Dennis Ritchie y sus colegas de laboratorios Bell.

Unix se basa en varias ideas muy buenas que aún están en plena vigencia:

- Para los creadores de Unix, un sistema operativo debería ser un conjunto de herramienta y conceptos básicos que le permitan al usuario construir rápida y fácilmente sus propias herramientas para automatizar procesos. Unix es como un juego de herramientas que puede aprovecharse al máximo con algo de estudio.
- Otra gran idea que se ha desarrollo junto con Unix ha sido Internet, porque en este sistema se realizaron las primeras implementaciones de los protocolos de comunicación en los que se basa Internet, y aún hoy son sistemas Unix los que mantienen Internet en funcionamiento [3].
- Otra buena idea ha sido usar y promover estándares abiertos. Hoy en día existen grupos que mantiene estándares que definen lo que es un sistema tipo Unix: Posix http://standards.ieee.org/regauth/posix/index.html y *The Open Group* http://www.opengroup.org —de hecho la palabra Unix es una marca registrada de X/Open.

Dado que sacar copias de un programa no es costoso ni difícil, y dado el carácter de Unix, desde sus comienzos este sistema ha promovido la colaboración y muchas personas han entregado al público las herramientas que han realizado para permitir su libre redistribución. En palabras del mismo Dennis

3. Por ejemplo el principal servidor de nombres en Internet es una máquina Unix(BSD), puede ver detalles sobre este servidor en http://www.isc.org/services/public/F-root-server.html

Ritchie “"Lo que queríamos preservar no era sólo un buen ambiente en el cual programar, sino también un sistema alrededor del cual se formara fraternidad"” [4].

Hay incluso varios sistemas operativos tipo Unix cuyas fuentes permiten la libre copia y libre redistribución —la mayoría se basan en Unix BSD [5]—, algunos son:

NetBSD

: Es un descendiente de 386BSD (que a su vez es un descendiente de BSD). NetBSD se centra en portabilidad, es decir que funciona en diversos tipos de computadores. http://www.netbsd.org.

OpenBSD

: Descendiente de NetBSD, centrado en seguridad y criptografía http://www.openbsd.org.

FreeBSD

: Descendiente de 386BSD, centrado en facilidad para el usuario final http://www.freebsd.org.

Linux

: El kernel de este sistema ha sido desarrollado de forma independiente de BSD desde 1992 por Linus Torvalds y un gran grupo de voluntarios (http://www.kernel.org). Otros componentes del sistema fueron tomados de BSD y otros fueron aportados por diversas organizaciones: *Free Software Foundation*, *XFree86*, *Apache Foundation*, etc.

Los sistemas tipo Unix han sido empleados tradicionalmente por programadores, por esto cuenta con compiladores, editores y herramientas para facilitar la programación —especialmente en lenguaje C. En el caso de Linux tales herramientas han sido desarrolladas por la *Free Software Foundation* http://www.fsf.org. Así mismo la FSF aportó la licencia que cubre al kernel (llamada GPL) y muchos componentes de un sistema Linux.

Al igual que otros sistemas operativos Unix/Linux cuenta con un sistema de ventanas que, a diferencia de otros sistemas operativos, es independiente de Unix/Linux —es decir puede ponerse, quitarse o cambiarse—. Tal sistema de ventanas se llama X-Window y la versión particular que se emplea en Linux se llama XFree86 http://www.xfree86.org.

Para facilitar la adopción de Unix por parte de usuarios sin experiencia con este sistema, hay varios proyectos en marcha, los dos más conocidos son GNOME (http://www.gnome.org) y KDE (http://www.kde.org), que funcionan sobre Linux y X-Window. Cada uno brinda un escritorio gráfico y aplicaciones de oficina como a las que están acostumbrados los usuarios de otros sistemas operativos (e.g. procesador de texto, hoja de cálculo, etc).

Hay muchos otros programas que funcionan sobre Linux y que satisfacen diversas necesidades. Dada tal cantidad y variedad de programas para Linux, hay grupos de personas dedicadas a configurar distribuciones. Una distribución de Linux es una selección de programas junto con herramientas para administrarlos e instalarlos con facilidad. Algunos nombres de distribuciones son Redhat, Mandrake, Suse, Debian. Nuestra plataforma de referencia (ver Plataforma de referencia) se basa en la distribución Debian que es de alta calidad técnica y que es realizada por voluntarios del mundo entero desde 1993.

---

4. Cita tomada de un artículo escrito por Dennis Ritchie sobre la historia de Unix, disponible en http://cm.bell-labs.com/cm/cs/who/dmr/hist.html
5. BSD es el nombre de una versión de Unix desarrollado en la universidad de Berkeley (Estados Unidos) desde los 70’s

1.2.2. Lecturas recomendadas: Linux

- Dennis Ritchie ha escrito una corta autobiografía, que está disponible en: http://cm.bell-labs.com/cm/cs/who/dmr/bigbio1st.html.
- La evolución del sistema Unix, como diagrama puede consultarse en http://perso.wanadoo.fr/levenez/unix/, allí mismo se encuentra una buena recopilación de enlaces a diversas versiones de la historia de Unix.
- Un muy buen libro en el cual se habla de la historia de Unix es *A Quarter Century of UNIX* de Peter H. Salus.
- La evolución de BSD puede consultarse en: http://www.oreilly.com/catalog/opensources/book/kirkmck.html.
- Puede consultarse más sobre la forma de desarrollo del kernel de Linux en el FAQ del mismo: http://www.tux.org/lkml.
- Hay un breve resumem de la historia de Linux, Unix y los movimientos de fuentes abiertas y software libre en la sección 2.1 de "*Secure Programming for Linux and Unix HOWTO*", escrito por David A. Wheeler: http://www.linuxdoc.org/HOWTO/Secure-Programs-HOWTO/history.html.

1.2.3. Ejercicios: Linux

1. Investigue más sobre las diferencias entre Linux y otros sistemas de libre redistribución como OpenBSD o FreeBSD.

2. ¿En que consisten los movimientos de fuentes abiertas y de software libre, y cual es su relación con Linux?

1.3. Sistema de usuarios y manejo de clave

Indicadores de Logros

- Entra y sale de su espacio personal.
- Se desplaza entre consolas virtuales y la sesión gráfica.
- Puede cambiar su clave.

1.3.1. Lectura: Sistema de usuarios y manejo de clave

Al igual que Unix, Linux es un sistema operativo multiusuario y multitarea. Multitarea significa que se puede realizar más de una tarea a la vez. Multiusuario significa que el sistema puede ser empleado por más de un usuario. Por ser multiusuario, Unix debe diferenciar a los usuarios y su información, para lograrlo a cada usuario se le asigna una identificación única (en inglés *login*), un grupo, una clave de acceso (en inglés *password*) y un espacio independiente del espacio de los demás usuarios (ese espacio se llama directorio personal y permite organizar información del usuario en archivos). Unix brinda también un esquema de permisos, con el que un usuario puede conceder o quitar permisos para ver o modificar sus archivos a otros usuarios o a usuarios de su mismo grupo. Gracias, al sistema de usuarios y al esquema de permisos no es posible que un usuario normal desconfigure el sistema completo o lo infecte con virus —la configuración general del sistema sólo puede cambiarla el administrador del sistema, cuyo login es `root`.

Como usuario de Linux, usted tendrá un login, un grupo principal, un espacio y una clave. Para evitar mala utilización de los recursos por parte de otra persona a nombre suyo, su clave de acceso debe ser **secreta**. Además por seguridad, es recomendable que cada usuario cambie su clave de acceso al menos cada dos meses, empleando el comando **passwd** desde un intérprete de comandos [6].

Puede entrar a una sesión con su login y clave bien desde un entorno gráfico (con un programa llamado gdm) o bien desde una interfaz tipo texto. Su sistema Linux puede brindarle simultáneamente varias interfaces tipo texto —llamadas consolas virtuales— junto con el entorno gráfico. Desde el entorno gráfico puede pasar a una consola virtual presionando **Ctrl-Alt-F1** [7], y desde una consola virtual puede retornar a la sesión gráfica con **Ctrl-Alt-F7**

Cuando inicie una sesión desde una consola virtual, entrará a un intérprete de comandos donde podrá dar comandos o ejecutar programas. En una sesión gráfica podrá emplear el ratón, por ejemplo para elegir opciones de menús, aún cuando en Unix es usual emplear un intérprete de comandos incluso desde sesiones gráficas (porque es más efectivo trabajar con 10 dedos en el teclado que con 2 sobre el ratón). Los programas iniciados desde una consola virtual, generalmente pueden interrumpirse con la secuencia de teclas **Control-C**, para reiniciar la sesión gráfica puede emplear la secuencia **Ctrl-Alt-Backspace**.

Tenga en cuenta salir de su sesión cuando termine de trabajar y recuerde no apagar el computador sin antes haber "cerrado" el sistema operativo (si apaga el computador antes de cerrar Linux puede que pierda información). Para salir de una sesión con un intérprete de comandos se emplea el comando **exit**, para salir de una sesión gráfica con Gnome busque entre los menús la opción Logout o Salir. La labor de "apagar" depende de la política que el administrador haya escogido, eventualmente usted podrá hacerlo desde los menús de gdm —el administrador puede apagar el sistema con el comando **halt** (ver Apagado y encendido de una red).

Con respecto al administrador(es) del sistema (quien maneja la cuenta *root*), recuerde que el también es una persona, es alguién que está sirviendole y ayudando a mantener la red y los recursos de su institución (que también son suyos). Seamos amables, respetuosos y humildes con él y con los demás usuarios,

6. Intérprete de comandos: en inglés *shell*
7. En su sistema puede haber varias consolas virtuales, puede pasar de una a otra con **Ctrl-Alt-F1** o la secuencia análoga con **F2**, **F3**, **F4** o **F5**.

ayudemos a cuidar la infraestructura que tenemos (un día podría estar a nuestro cargo y nos gustaría que los usuarios ayudarán a cuidarla).

1.3.2. Lecturas recomendadas: Sistema de usuarios y manejo de clave

- Como complemento a esta lectura puede consultar las secciones 3.2.1 a 3.2.6 del libro "*Linux Installation and Getting Started*", escrito por Matt Welsh y otras personas. Está disponible en Internet en http://www.linuxdoc.org/LDP/gs/node5.html#SECTION00500000000000000000.

1.3.3. Ejercicios: Sistema de usuarios y manejo de clave

1. Averigüe su login y su clave con el administrador del sistema, uselos para entrar a una sesión gráfica con el escritorio GNOME y después salga.

> **Aviso**
>
> No olvide salir de su sesión cada vez que deje de usarla. Si la deja abierta alguien podría actuar a nombre suyo.

2. Pase a consolas virtuales y regrese a modo gráfico. Practique el uso de la secuencia de teclas **Ctrl-Alt-Backspace**.

> **Aviso**
>
> Nuevamente le recordamos no apagar el computador con el interruptor o con el botón *reset* sin antes haber salido de Linux.

3. Empleando una consola virtual abra una sesión con su login, cambie su clave (con el comando

passwd), salga de la cuenta y regrese a su escritorio GNOME.

Aviso

Después de abrir una consola virtual o de entrar a una sesión del intérprete de comandos con su login, al terminar no olvide salir de la sesión con el comando **exit**. Para evitar problemas de seguridad es recomendable que su clave tenga más de 6 caracteres (algunos sistemas sólo permiten hasta 8), que no sea una palabra que aparezca en diccionario alguno (de ningún idioma), tampoco debe ser su nombre, ni el nombre de su mascota, ni de un ser querido, ni su dirección, ni el nombre de bandas de rock (trate de imaginar que clave usaria otra persona y NO use esa). Idee una palabra fácil de recordar sólo para usted, no la anote en un papel y emplee preferiblemente signos de puntuación y números (pero no **Backspace**), aún mejor si idea un esquema que le permita cambiarla con frecuencia. Otra importante recomendación con respecto a la clave: *no la olvide!*

1.4. Escritorio y herramientas Gnome

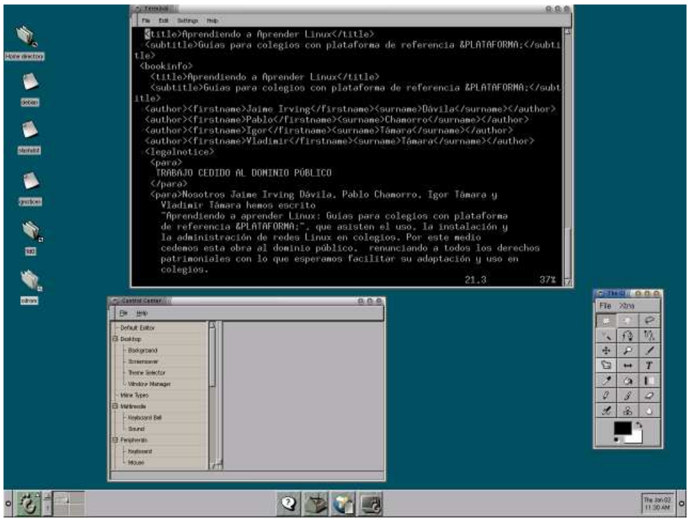
Indicadores de Logros

- Emplea y configura el escritorio Gnome.

1.4.1. Lectura: Escritorio y herramientas Gnome

Cuando inicie una sesión gráfica verá el escritorio Gnome. Este escritorio cuenta con iconos para acceder rápidamente a aplicaciones y con menús que le permiten iniciar programas. También le permite usar varios espacios de trabajo [8], cada uno como un escritorio independiente de los demás —aunque es fácil pasar aplicaciones de un espacio de trabajo a otro pues basta arrastrarlas con el ratón.

8. Espacios de trabajo: del inglés *workspaces*



El botón izquierdo del ratón normalmente permite elegir una opción de un menú o activa un icono. El botón derecho tiene diversas aplicaciones de acuerdo al contexto —por ejemplo sobre los iconos permite configurarlos—, el botón del centro permite pegar el texto que se hubiera seleccionado con el ratón, si su ratón sólo tiene 2 botones, puede "emular" el botón del centro oprimiendo simultáneamente el izquierdo y el derecho. Para seleccionar un texto se pasa por encima del mismo con el puntero del ratón mientras se mantiene presionado el botón izquierdo.

El escritorio Gnome es bastante configurable, usted puede configurar los menús, los iconos, el tipos de letra, el fondo, el protector de pantalla, el tema, el administrador de ventanas, sonidos, la interacción con las ventanas y muchos otros detalles de acuerdo a su gusto. Para hacer algunas de las configuraciones puede emplear opciones de los menús Gnome, el ratón —por ejemplo para administrar los iconos que hay sobre el escritorio—, el programa "Centro de control Gnome" y eventualmente los archivos de configuración de X-Window. [9].

El administrador de ventanas que emplee decorará cada ventana con botones que le permitirán cerrar, maximizar o minimizar. Las ventanas minimizadas se verán en la parte inferior del escritorio Gnome —más precisamente en el panel que no necesariamente esta en la parte inferior, porque puede reubicarse de acuerdo a su gusto.

Entre los menús de Gnome, encontrará diversas herramientas: calculadora gcalc; editor de texto sencillo gnotepad; calendario gnomecal, procesador de palabra, hoja de cálculo, graficadores, reproductor de sonido y vídeo y muchas otras que puede aprender a usar consultando la documentación —como se explica más adelante. Entre las aplicaciones debe haber una terminal, se trata de un intérprete de comandos en una ventana (para iniciarla puede haber varias opciones entre los menús de Gnome: Regular xterm, color xterm, GNOME terminal), este intérprete se opera con teclado y es la herramienta integradora del juego de herramientas Unix.

9. En sus experimentos de configuración, puede que le sirva saber que X-Window se encarga de poner en la pantalla toda imagen que vea y eventualmente si usted usa una terminal gráfica o accede remotamente un computador, X-Window maneja los protocolos de comunicación. El administrador de ventanas indica a X-Window cómo manejar los bordes, botones e interacción con cada ventana y así mismo administra espacios de trabajo. Gnome controla tanto a X-Window como al administrador de ventanas, los menús, los íconos y presta servicios a diversas aplicaciones gráficas

1.4.2. Lecturas recomendadas: Escritorio y herramientas Gnome

- Como complemento a esta lectura puede consultar la sección "Primera visita a GNOME" en la "*Guía de usuario de GNOME*": http://www.croftj.net/~barreiro/spanish/gnome-es/gnome-users-guide/quickstart.html

1.4.3. Ejercicios: Escritorio y herramientas Gnome

1. Abra un editor de texto, escriba algo de texto, y después copielo y peguelo empleando el ratón (para pegar emplee el boton del centro).

2. Ponga como fondo de su escritorio una imagen (Ayuda: Emplee el Centro de Control GNOME).

3. Cambie el tema de su escritorio.

4. Explore posibilidades de su gestor de ventanas.

5. Abra cada una de las herramientas que encuentre en los menús de GNOME y explore por unos minutos las posibilidades de cada una.

6. Para cada herramienta piense una aplicación que pueda darle a corto o mediano plazo y comience a usarla.

1.5. Directorios y archivos

Indicadores de Logros

- Emplea el administrador de archivos.
- Emplea los comandos **cd**, **ls** y **pwd** desde un intérprete de comandos.

1.5.1. Lectura: Directorios y archivos

La primera vez que inicie una sesión, posiblemente además del escritorio se iniciará un programa que le permite administrar sus archivos empleando el ratón (esta aplicación puede iniciarse también desde los menús de Gnome bajo el nombre Administrador de Archivos o gmc o bien Nautilus). Este programa y el intérprete de comandos son importantes porque le permiten organizar la información que se almacena en discos duros, disquetes y otros dispositivos.

Un sistema Linux instalado puede tener más de 1’000.000 de archivos que afortunadamente se identifican por nombres que pueden organizarse dentro de directorios. En el sistema de archivos de Linux, los directorios se organizan a manera de árbol (un directorio puede contener otros directorios distintos) a partir de un directorio principal llamado directorio raiz y que se denota por '`/`'. Cada directorio puede contener otros directorios o nombres de archivos. Un archivo es información de un mismo tipo localizada en algún dispositivo de almacenamiento junto con atributos como: fecha de creación, fecha de última actualización, login del dueño, grupo al que pertenece y permisos. Cada archivo puede tener diversos nombres en diversos directorios, cada nombre es como un enlace a la información del archivo.

Como cada programa tiene su propia forma de almacenar información, para facilitarnos la identificación de un archivo, suele agregarse un postfijo al nombre del archivo –tales postfijos suelen llamarse extensiones. Algunos postfijos comunes se presentan a continuación junto con el programa que puede emplearse para verlos/editarlos:

`.txt`

Textos planos, pueden editarse con un editor de texto.

`.jpg, .xpm, .gif, .png`

Gráficas en diversos formatos, pueden editarse por ejemplo con gimp.

`.gz, .Z, .zip`

Información comprimida, pueden descomprimirse con **gzip -d**, **uncompress** y **unzip** respectivamente. Para comprimir en estos formatos se usa **gzip**, **compress** y **zip**. Por ejemplo si en el directorio de trabajo exsite un archivo `cartalarga.txt` el comando **gzip cartalarga.txt** producirá en el mismo directorio el archivo comprimido `cartalarga.txt.gz`.

`.html`

Hipertextos HTML que pueden verse con un navegador y editarse con un editor de texto –también hay herramientas especializadas para editar HTML.

`.tar .tgz .tar.gz`

Archivo con varios archivos empaquetados, pueden desempaquetarse con el programa **tar** (si están además comprimidos el sufijo puede ser `.tar.gz` o `.tgz`). Para crear un empaquetado `tareas.tar` a partir del contenido del directorio `/home/pepe/tareas` puede usarse: **tar cvf tareas.tar /home/pepe/tareas**. Para desempaquete el archivo `juego.tar` en el directorio de trabajo se usa: **tar xvf juego.tar**. Para desempaquetar y descomprimir `llevar.tgz` (o `llevar.tar.gz`) puede descomprimir primero con **gzip** y el resultado desempaquetarlo con **tar** o puede emplear la opción `z` de **tar**: **tar xvfz llevar.tgz**. De forma análoga para empaquetar el directorio llevar y comprimir el empaquetado puede emplear **tar cvfz llevar.tgz llevar**.

Los programas (e.g el calendario, un editor de texto, etc) son también archivos. Los nombres de los programas ejecutables usualmente no tienen un postfijo común. Para ejecutarlos puede teclear el nombre desde un interprete de comandos. Para crearlos se emplea algún lenguaje de programación y un compilador [10].

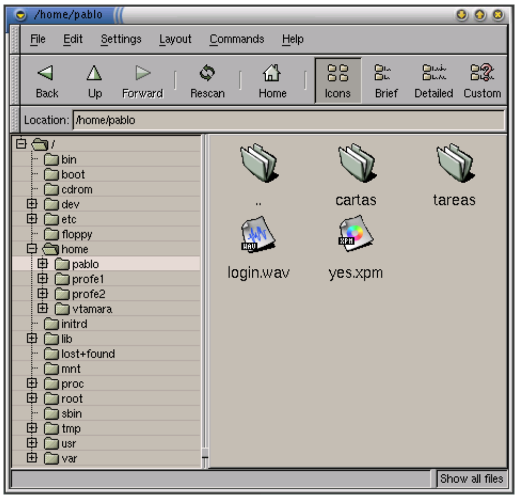
`.sh`

Script para el intérpete de comandos. Puede ejecutarse tecleando el nombre desde un interprete de comandos o con el comando **source**.

Como un archivo puede referenciarse con un nombre en un directorio, es posible localizarlo en el árbol completo de directorios indicando la secuencia de directorios que se debe pasar para llegar al directorio donde está el nombre. A esa secuencia se le llama **ruta**, se escribe comenzando con `/` (para denotar el directorio raiz) y separando un directorio de otro con un `/`. Cuando interactúe con el sistema o con programas podrá referenciar archivos anteponiendoles su ruta, por ejemplo un archivo con nombre `a`

10. Un compilador transforma un programa escrito en un lenguaje de programación a lenguaje de máquina que puede ser ejecutado por el computador –un computador sólo puede ejecutar programas escritos en lenguaje de máquina.

ubicado en el directorio `n` que a su vez está en el directorio `j`, él cual es un directorio en el directorio raiz se identifca con: `/j/n/a`.



Desde un administrador de archivos podrá ver una representación gráfica del arbol de directorios y puede emplear el ratón para pasar de un directorio a otro. En un interprete de comandos siempre estará ubicado en un directorio llamado directorio de trabajo y que cuando inicie una sesión corresponderá a su directorio personal. Puede examinar su directorio de trabajo con el comando **pwd**, puede emplear el comando **cd** para cambiarse de un directorio a otro y puede examinar el contenido de un directorio con el comando **ls**. Por ejemplo el comando **cd /** lo ubicara en el directorio raiz, si teclea entonces **pwd** verá `/`, al teclar **ls** verá algo como:

```
bin   cdrom  etc     home    lib         mnt   root  tmp  var
boot  dev    floppy  initrd  lost+found  proc  sbin  usr  vmlinuz
```

que corresponde al contenido del directorio raíz –en este caso son sólo directorios. Si teclea **cd home** pasará al directorio `home` —también puede pasar a ese directorio con **cd /home** [11].

Empleando el administrador de archivos, verá que puede organizar los nombres de sus archivos en directorios. Puede mover un archivo de un directorio a otro seleccionándolo con el puntero del ratón y arrastrándolo, es decir moviendo el ratón manteniendo presionado el botón izquierdo; arrastrando directorios o nombres de archivos creará nuevos iconos. Puede copiar archivos de la misma forma como los mueve pero manteniendo presionada la tecla Control, o alternativamente puede emplear opciones de los menús —por ejemplo la opción "Copiar" y posteriormente la opción "Pegar". Puede borrar un nombre de archivo o un directorio seleccionándolo y presionando la tecla **Supr** (o **Del** si su teclado es en inglés) o alternativamente después de seleccionar puede escoger una opción apropiada de un menú —por ejemplo "Cortar". Para renombrar un archivo o un directorio después de seleccionarlo presione nuevamente el botón izquierdo o alternativamente busque la opción "Renombrar" entre los menús del administrador de archivos o emplee el botón derecho una vez el puntero esté sobre el archivo. Al emplear el administrador de archivo descubrirá que sólo puede escribir o leer de ciertos directorios, esto ocurre por un sistema de permisos que se explicará más adelante, por lo pronto basta que sepa que su directorio

11. La diferencia entre usar **cd home** y **cd /home** es que la primera forma indica un directorio relativo al directorio en el cual está ubicado, mientras la segunda especifica la ruta completa. Asi el archivo `/home/pepe/e.txt` puede referenciarse como `pepe/e.txt` desde el directorio `/home` o simplemente como `e.txt` desde el directorio `/home/pepe`

personal (donde usted tiene permiso para leer, escribir y cambiar permisos) es un directorio dentro del directorio `/home`, es el directorio cuyo nombre es su login —el nombre completo de su directorio es entonces de la forma `/home/sulogin`.

1.5.2. Lecturas recomendadas: Directorios y archivos

A manera de complemento, de la "*Guía de usuario de GNOME*" puede consultar las secciones "Archivo y nombre de Archivos" y "Directorio y Rutas":

http://www.croftj.net/~barreiro/spanish/gnome-es/gnome-users-guide/new-file.html
http://www.croftj.net/~barreiro/spanish/gnome-es/gnome-users-guide/new-dirs.html

1.5.3. Ejercicios: Directorios y archivos

1. Abra `gmc`, explore los menús y las posibilidades de este programa. Si lo desea puede leer la ayuda de `gmc` (con el sistema de ayuda de GNOME puede consultar la ayuda en `file:/usr/share/gnome/help/users-guide/C/gmc.html`).

2. Cree un directorio `graficas` en su espacio personal.

3. Busque una gráfica en el directorio `/usr/share/pixmaps` y cópiela en el directorio `graficas` que creó –si lo desea puede veral bien con doble clic desde el administrador de archivos o con el comando **gimp graf** remplazando graf por el nombre de la gráfica.

4. Borre la gráfica y el directorio que había creado.

5. Intente borrar una gráfica del directorio `/usr/share/pixmaps` (no podrá a menos que usted sea el administrador del sistema, aún así, compruébelo).

1.6. Disquetes y CDROM

Indicadores de Logros

- Puede montar y desmotar diskettes y transferir información empleando GNOME.
- Puede montar y desmontar diskettes desde la línea de comandos.

1.6.1. Lectura: Disquetes y CDROM

Para transferir información de un computador a otro, o para tener copias de respaldo de alguna información que tenga en su espacio personal, puede emplear disquetes. En los sistemas Unix antes de poder usar un disquete es necesario "montar" el sistema de archivos del diquete en el árbol de directorios; para realizar esta operación puede bien usar el icono de disquete que encuentra en su escritorio y el botón derecho del ratón o desde un intérprete de comandos teclear:

```
mount /floppy
```

De forma análoga antes de sacar un disquete debe desmontar la unidad, esto también lo podrá hacer con el icono del disquete y el botón derecho del ratón o desde un intérprete de comandos con el comando

```
umount /floppy
```

Una vez montado un disquete puede leer o escribir información en el mismo en el directorio `/floppy` —note que cuando desmonta el disquete no se ve información en ese directorio.

Linux puede leer y escribir disquetes formateados para otros sistemas operativos, así que puede usar disquetes que emplea por ejemplo en Windows. Si desea formatear un disquete con un formato que pueda emplear en Windows o DOS emplee el comando **mformat** desde un intérprete de comandos.

> **Aviso**
>
> Al formatear un disquete se pierde la información que antes hubiera.

Si la distribución de Linux que está usando es Debian, para poder emplear la unidad de disquete, el administrador debe otorgarle permiso —agregándolo al grupo `floppy`. Sin tal permiso, las instrucciones presentadas en esta guía no le serán de utilidad.

Si su computador cuenta con unidad de CDROM y su login tiene permiso para usarlo —lo tiene si pertenece a los grupos `disk` y `cdrom`— puede montar CDROMs con datos y escuchar CDs con música (claro puede escucharlos mientras tenga tarjeta de sonido configurada, parlantes o audifonos y su login esté en el grupo `audio`). Para montar un CD con datos emplee el comando **mount /cdrom** y consulte la información en el directorio `/cdrom`. Si desea escuchar la música de un CD busque un programa apropiado entre los menús de Gnome.

1.6.2. Lecturas recomendadas: Disquetes y CDROM

Como complemento a esta lectura puede consultar de la "*Guía de usuario de GNOME*" la sección "Montando y desmontando dispositivos":

http://www.croftj.net/~barreiro/spanish/gnome-es/gnome-users-guide/new-mount.html.

1.6.3. Ejercicios: Disquetes y CDROM

1. Averigüe el formato de disquetes que puede usar en el sistema Linux que tiene disponible y consiga un disquete con tal formato.

2. Ponga el disquete en la unidad, montela y copie algún archivo (por ejemplo una gráfica).

3. Desmonte la unidad y saque el disquete.

> **Aviso**
>
> No olvide desmontar la unidad, antes de sacar un disquete o de lo contrario puede perder información.

4. Opcional. Existe otra forma de emplear disquetes que no requiere montarlos ni desmotarlos: con las herramientas mtools. Consulte la documentación de estas herramientas con el comando **info mtools** desde un intérprete de comandos y úselas.

1.7. Búsqueda y consulta de documentación

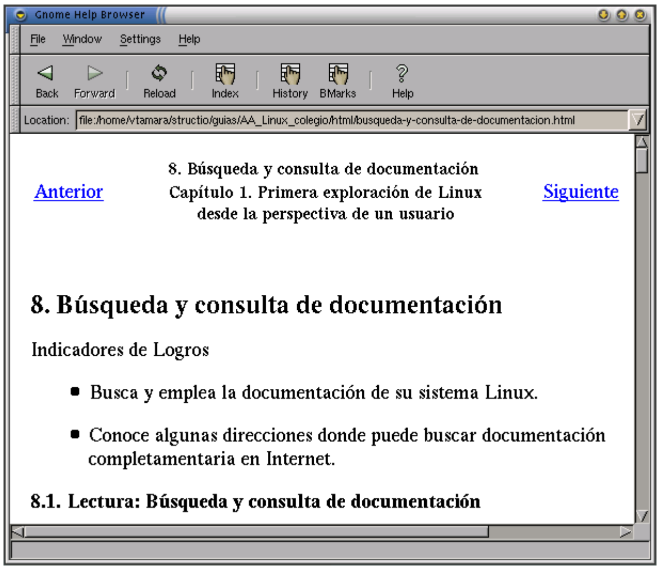
Indicadores de Logros

- Busca y emplea la documentación de su sistema Linux.
- Conoce algunas direcciones donde puede buscar documentación completamentaria en Internet.

1.7.1. Lectura: Búsqueda y consulta de documentación

Al emplear su sistema Linux notará que debe leer la documentación de cada programa y del sistema. Es importante que lo haga por varias razones:

- Linux es un sistema que evoluciona muy rápido con la ayuda de muchas personas en el mundo (nosotros también podemos ayudar!), por eso algunas particularidades pueden cambiar de una versión a otra –aunque los conceptos fundamentales de Unix permanecen como sustento.
- Sólo consultando la documentación de cada programa podrá aprovecharlo al máximo.
- Al buscar y consultar documentación por su cuenta, está aprendiendo por si mismo.



Una dificultad que puede encontrar al consultar documentación para Linux es el idioma, pues la documentación más actualizada está disponible sólo en Inglés. Aún así, generalmente, son documentos técnicos sencillos que podrá entender con unos conocimientos mínimos de inglés (además de que podrá poner en práctica ese idioma!).

Para buscar documentación, no necesita estar conectado a Internet, su sistema Linux incluye gran cantidad de información que le permitirá aprender, solucionar sus problemas y construir sus herramientas. En particular puede consultar los siguientes tipos de documentación:

- Estas guías. Si las tiene instaladas en su computador puede consultarlas de una de las siguientes formas:
  - con un navegador puede localizarlas como páginas HTML en el directorio `/usr/local/share/doc/AA_Linux_colegio`
  - en el mismo directorio encuentra una versión PostScript que puede ver e imprimir con el comando **gv /usr/local/share/doc/AA_Linux_colegio/AA_Linux_colegio/AA_Linux_colegio.ps**.

- Páginas del manual de Unix. Los desarrolladores de Unix, simultáneamente con el sistema operativo, concebieron un sistema documentación para editar, imprimir y ver en pantalla documentos. El manual del usuario de Unix está escrito con tal sistema. Cada página de este manual tiene un nombre con el que se identifica, y que puede emplearse con el comando **man** (desde una línea de comandos teclee **man** seguido del nombre de la página). Hay páginas para cada programa disponible, para los comandos que acepta el intérprete de comandos, páginas con convenciones de los archivos, y páginas para programadores que deseen emplear el sistema operativo. Cada página tiene además del nombre un resumen, una descripción, opciones de uso, archivos relacionados, referencias a documentación relacionada, errores y los nombres de los autores. El programa **man** permite, entre otras, ver la página solicitada, moverse sobre la misma y realizar búsquedas —para buscar presione **/** y después teclee la palabra que busca.
- Manuales info. Son libros o manuales que documentan algunos programas. Para consultarlos en pantalla puede emplear el programa **info** seguido del nombre del manual que desea consultar. Si

emplea el comando **info** sólo, entrará a un índice de los manuales disponibles.

- Gnome cuenta con un sistema de ayuda, que le permite navegar por las páginas del manual, por documentos info y por documentación en HTML de Gnome. Puede entrar a este sistema buscando "Sistema de ayuda" entre los menús de Gnome. Desde el intérprete de comandos lo puede hacer con **gnome-help-browser** (Solamente funciona en el entorno gráfico).
- Varios programas pueden mostrar una corta ayuda cuando se invocan desde un intérprete de comandos seguidos de un espacio y la opción `--help`.
- En los directorios `/usr/doc` y `/usr/share/doc`, podrá encontrar ayuda para varios programas, en diversos formatos. Los formatos de cada documento varían y usualmente están comprimidos, si lo están el nombre terminará en `.gz` y en ese caso, antes de consultar el documento comprimido debe copiarlo a su directorio y descomprimirlo con el programa **gzip** seguido de `-d` y a continuación el nombre del documento. Si el documento es un archivo de texto (el nombre suele termina en `.txt`) puede emplear un editor de texto para consultarlo, o desde un intérprete de comandos el programa **more** o el programa **less** (también puede usar **zless** para consultar un archivo tipo texto comprimido). Si el archivo es tipo `.dvi` puede emplear el programa `xdvi` para consultarlo. Si el archivo es Postscript (`.ps` o `.eps`) emplee el programa gv para verlo o imprimirlo. Si es un documento HTML (`.html`) consúltelo con un navegador, con el sistema de ayuda de Gnome o desde un intérprete de comandos con el programa **lynx** o con **w3m**. Si el documento está en formato PDF (`.pdf`) puede emplear bien gv o bien xpdf.
- Entre los documentos del directorio `/usr/doc` hay unos documentos tipo texto que describen de manera informal cómo administrar o configurar diversos componentes de un computadores con Linux, tales guías se llaman *HOWTO*s, y son un buen punto de referencia para quienes desean administrar un sistema Linux.

Si tiene acceso a Internet, encontrará un mar de información sobre Linux —tanta que podrá perderse o confundirse. Una vez encuentre información, le sugerimos revisar que la versión del programa al que se refiere la documentación concuerde con la versión del programa que usted emplea (para ver la versión de algunos programas puede dar el nombre del programa seguido de la opción `--version`). A continuación presentamos algunos URLs de sitios donde puede consultar información sobre Linux:

- Estas guías pueden consultarse y obtenerse en http://structio.sourceforge.net/guias/AA_Linux_colegio.
- El proyecto LuCAS se dedica a recolectar información sobre Linux en español: http://lucas.hispalinux.es/
- Puede consultar algunas páginas **man** en: http://www.delorie.com/gnu/docs/ o en http://www.openbsd.org/cgi-bin/man.cgi
- Varios de los manuales de usuario tipo **info** están disponibles en: http://www.gnu.org/manual/manual.html
- La guía del usuario GNOME en español está disponible en: http://www.croftj.net/~barreiro/spanish/gnome-es/users-guide/
- El *Debian Documentation Project* produce documentación para la distribución Debian Linux. Los libros y documentos de este proyecto están disponibles en: http://www.debian.org/doc/ddp
- Hay una vasta colección de información para diversas distribuciones de Linux, así como guías *HOWTO* en el sitio web de *The Linux Documentation Project*: http://www.tldp.org

1.7.2. Lecturas recomendadas: Búsqueda y consulta de documentación

- El sistema de documentación de Unix se basa en el formateador **troff**, así como pre y postprocesadores para el mismo. Con ayuda de algunos paquetes de macros pueden editarse ecuaciones **eqn**, tablas **tbl** y figuras **pic** y con postprocesadores el resultado puede presentarse o imprimirse en diversos dispositivos. El tema es tratado en detalle en "*Document Formatting and Typesetting on the Unix System*" de Narain Gehani, Silicon Press. En línea puede consultar el manual de Groff –la versión de troff de Linux– en http://www.cs.pdx.edu/~trent/gnu/groff/groff_toc.html.
- Los manuales info están escrito con el sistema de documentación Texinfo. Texinfo es una extensión a un sistema llamado TeX que permite escribir documentos de matemáticas. Puede consultar más sobre Texinfo en http://texinfo.org/
- Pueden resultar de interés para algunas personas estos enlaces a la primera edición del manual de Unix (1971): http://cm.bell-labs.com/cm/cs/who/dmr/1stEdman.html

1.7.3. Ejercicios: Búsqueda y consulta de documentación

1. Desde una interfaz de comando teclee **man man** y consulte la explicación sobre el programa **man** y el manual del sistema. En particular averigüe en esa página para que es el comando **whatis**.

2. Consulte la documentación de GNOME, en particular "*Guía de Usuario GNOME*", "*Lo básico sobre Applet: ¿Qué son Applets?*" [12]. Con la información que lea agregue el applet de su preferencia en el panel.

3. Abra bash y teclee **info**. Déle un vistazo a la tabla de contenido de la documentación que tiene disponible. Algunos de estos documentos son libros completos. Entre a la documentación de emacs (un editor de texto) y revísela por encima. Opcional: presione la tecla **h** para aprender a usar mejor el programa `info` o la tecla **?** para ver un resumen de las teclas que puede usar.

4. Estando en un intérprete de comandos teclee **mv --help**. ¿Para qué sirve el comando **mv**?

5. Revise el directorio `/usr/doc`, liste algunos de los directorios y el tipo de información que allí encuentre.

6. Revise el directorio `/usr/doc/HOWTO`. Revise alguno de los documentos de ese directorio. ¿Qué documento revisó? ¿Qué información encontró?

7. Opcional. Si tiene conexión a Internet consulte http://www.linuxdoc.org ¿En qué consiste el *Linux Documentation Project*?

12. Posiblemente la encontrará en `file:/usr/share/gnome/help/users-guide/C/gnome-applets.html`

Capítulo 2. Linux desde la perspectiva de un usuario con experiencia

Público y logros

Estudiantes, profesores, secretarias y personas que tienen alguna experiencia con Linux.

- Entiende mejor como se organizan y usan archivos, directorios y permisos.
- Sabe mantener su información pública y puede consultar la de otros usuarios.
- Aprovecha la red para comunicarse con otros usuarios.
- Emplea los servicios de la intranet de la institución: **telnet**, X-Window, **ftp** y web.
- Edita textos con vi o emacs.

2.1. Archivos y permisos

Indicadores de Logros:

- Maneja los permisos de sus archivos.
- Emplea comandos para manejo de directorios y archivos desde un intérprete de comandos.

2.1.1. Lectura: Archivos y permisos

En esta sección se introducen diversos comandos para administrar archivos y permisos. Si bien estas operaciones pueden hacerse con una herramienta gráfica como Nautilus o el administrador de archivos de Gnome (gmc), resulta más rápido emplear comandos desde bash, y en algunas situaciones es la única opción (por ejemplo si el modo gráfico falla o si emplea un sistema que no tenga X-Window).

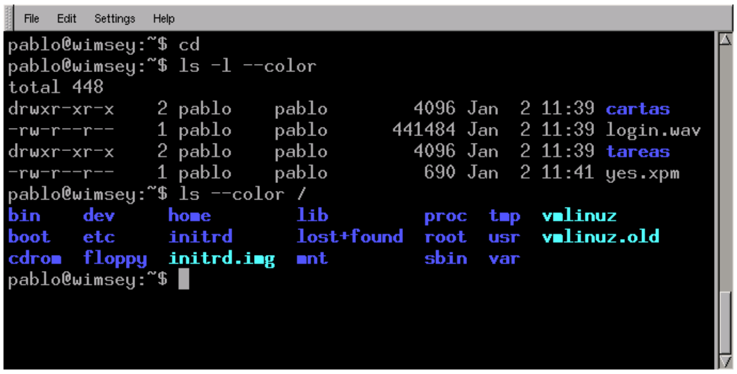
Un comando usualmente recibe opciones y parametros que especifican que operación realizar; algunos parametros son indispensables mientras que las opciones modifican el comportamiento por defecto de un comando (y por tanto son opcionales). En la ayuda presentada por el comando **man** y en estas guías emplemos como notación los paréntesis cuadrados `[ ]` para encerrar parametros opcionales y opciones. Por ejemplo

```
ls [-l] [ruta]
```

indica que el comando **ls** puede recibir la opción `-l` y una ruta como parámetro (las opciones suelen comenzar con uno o dos guiones). Puede descubrir que hace tal opción y el parámetro tecleando **ls / -l** y comparando con **ls -l**, **ls /**, y **ls**.

2.1.1.1. Administración de archivos, directorios y enlaces

El programa **ls** por defecto presenta los nombres de archivos que no comienzan con el caracter '.' y que están en el directorio de trabajo del usuaurio. Los archivos cuyos nombres comienzan con el caracter punto suelen llamarse archivos de configuración, si desea verlos también al usar el comando **ls**, puede emplear la opción `-a` (i.e. teclear **ls -a**). Si desea ver los archivos de un directorio diferente puede dar como argumento el nombre del directorio, por ejemplo **ls /usr/bin**. Además de `-a` y `-l`, el comando **ls** tiene muchas otras opciones que le permiten especificar que mostrar y como mostrarlo.



A continuación se presentan algunos programas para administrar archivos, junto con una breve descripción y unas pocas opciones —Puede consultar todas las opciones que recibe un programa con **man**:

ln `[-s]` *`fuente destino`*

: Crea un nuevo enlace para el archivo referenciado por *`fuente`* pero con el nombre *`destino`*. Por ejemplo si en el directorio de trabajo hay un archivo enlazado con el nombre `carta.txt` puede crearse otro enlace llamado `diario.txt` con:

```
ln carta.txt diario.txt
```

: Después de hacerlo tanto `carta.txt` como `diario.txt` serán nombres que enlazarán la misma información. La opción `-s` indica que el enlace debe ser símbolico, lo cual en general es más flexible que la opción por defecto [1].

mkdir `[-p]` *`ruta`*

: Crea un directorio con la ruta especificada. Sin la opción `-p` todos los directorios de la ruta excepto el último deben existir (el último es el nuevo directorio por crear). Con la opción `-p` el comando **mkdir** creará todos los directorios necesarios para construir la ruta.

mv *`fuente destino`*

: Renombra o mueve el archivo fuente en la localización/nombre destino.

---

1. El comportamiento por defecto de **ln** es crear enlaces "duros", es decir asociar el nombre del enlace con la información del archivo fuente. Un enlace suave es una referencia al nombre de un archivo —no a la información

rm `[-i] [-rf]` *`archivo`*

Borra un archivo.

> **Aviso**
>
> Una vez se borra un archivo no hay un método sencillo para recuperarlo.

La opción `-i` confirma antes de borrar cada archivo, las opciones `-rf` permiten borrar directorios enteros junto con todos los subdirectorios que contengan (la opción `-r` por si sóla borra en subdirectorios, la opción `-f` borra sin preguntar al usuario y sin producir fallas cuando no haya archivos por borrar).

cp `[-rf]` *`fuente destino`*

Copia del archivo fuente al destino especificado. Si se usa la opción `-rf` también se copiaran subdirectorios.

df `[-h]`

Para examinar espacio disponible en los dispositivos de almacenamiento, en particular el de las particiones cuyos sistemas de archivos estén montados (ver Montaje y desmontaje de sistemas de archivos). Puede emplearse con la opción `-h` para obtener datos en unidades más conocidas (en Kilobytes, Megabytes y Gigabytes) [2].

du `[-s] [`*`ruta`* `[`*`ruta`*`] ... ]`

Para examinar espacio empleado por cada una de las *`ruta`* y sus archivos y subdirectorios. Si no se especifica ruta alguna, este comando da información sobre el directorio de trabajo. La opción `-s` presnta totales de cada ruta (sin incluir los detalles de cada archivo o subdirectorio).

Al usar la opción `-a` con **ls**, por lo menos verá dos directorios más: `.` y `..`, el primero denota el directorio que examinó y el segundo denota el directorio en el cual está el directorio examinado. Por ejemplo si su directorio de trabajo es `/home/pepe` el caracter `.` abrevia `/home/pepe` mientras que `..` abrevia `/home`.

En la mayoría de programas y en el intérprete de comandos podrá emplear tanto `.` como `..` para referenciar archivos y subdirectorios relativos al directorio de trabajo. Así mismo los archivos y directorios que esten en su directorio de trabajo puede referenciarlo sin prefijo alguno, por ejemplo **du .** es equivalente a **du** porque muestra la utilización de disco del directorio de trabajo. Si en su directorio de trabajo tiene un directorio notas que tiene un archivo `2000.txt` puede verlo con el comando **less notas/2000.txt**.

También podrá emplear ~ que abrevia su directorio personal (i.e `/home/sulogin`). Por ejemplo si desde un intérprete de comandos teclea **cd ~**, después **cd ../..** y después **pwd** verá que estará ubicado en el directorio raiz.

Otra facilidad que brinda el interprete de comandos es la expansión de nombres de archivos con los comodines * y ?. ? indica una letra mientras que * indica una cantidad cualquiera de letras. Por ejemplo **ls /var/lo?** lista todos los archivos del directorio `var` de tres letras que comiencen con `lo`, mientras que **cp *gz ~** copia todos los archivos con sufijo gz en el directorio personal del usuario.

---

2. Un byte corresponde a un caracter, puede pensar en Kilobyte como en 1000 bytes –en realidad son 1024 bytes–, en Megabyte como 1'000.000 de bytes o 1000 Kilobytes –en realidad 1024^2– y en un Gigabyte con 1000'000.000 de bytes o 1000 Megabytes –en realidad 1024^3.

2.1.1.2. Permisos

Para brindar algo de privacidad y protección cada archivo o directorio tiene asociados permisos diferentes para el dueño, para el grupo y para los demás usuarios. En el caso de archivos los permisos que pueden darse o quitarse son: (r) lectura, (w) escritura y (x) ejecución. En el caso de directorios los permisos son: (r) para listar los archivos, (w) para escribir, crear o borrar archivos y (x) para acceder a archivos del directorio. [3].

Desde un administrador de archivos, puede ver los permisos de un archivo con el botón derecho del mouse cuando el puntero está sobre el archivo, escogiendo la opción apropiada del menú que aparece. Desde un interprete de comandos puede emplear el comando **ls** con la opción `-l`. Un ejemplo del resultado de este comando se presenta a continuación:

```
drwxr-xr-x    5 pepe   users        4096 Feb 21 06:31 graficas
-rw-r---    1 pepe   users       62561 May 13 18:13 c.tar.gz
lrwxrwxrwx    1 pepe   users          12 Nov 12  2000 a -> /etc/hosts
```

La primera línea presenta un directorio (la `d` al principio de la línea lo indica), la segunda presenta un archivo (el guión inicial lo indica) y la tercera un enlace. El nombre del directorio es `graficas` tiene 5 archivos, fue modificado por última vez el 21 de Febrero del año en curso a las 6:31AM, el dueño es `pepe`, el grupo es `users` y el tamaño es 4096 bytes —en realidad el tamaño cobra sentido sólo en el caso de archivos como `c.tar.gz` cuyo tamaño es 62561 bytes. Los tres caracteres `rwx` que siguen a la `d` inicial indican los permisos para el dueño, los tres siguientes `r-x` indican los permisos para el grupo y los tres siguientes `r-x` indican los permisos para el resto de usuarios. Como el orden de estos permisos es siempre el mismo (primero lectura `r`, después escritura `w` y después ejecución `x`), resulta que el archivo `x.tar.gz` no es ejecutable, que puede ser leido por el dueño y el grupo pero no por los demás usuarios, además puede ser escrito sólo por `pepe`. Del enlace podemos destacar que se llama `a`, que enlaza al archivo `/usr/bin/awk` y que su tamaño y permisos reales los heredará de `/usr/bin/awk`.

Los permisos de un archivo pueden ser modificados por el dueño o por el administrador del sistema con el comando **chmod** que espera dos parámetros: cambio por realizar al permiso y nombre del archivo por cambiar. Los permisos se pueden especificar en octal o con una o más letras para identificar al usuario (u para el usuario, g para el grupo, o para los demás usuarios y a para todos), un +, un - o un = y después letras para identificar los permisos (r, w o x). Por ejemplo

chmod og+x sube.sh

Da a los demás usuarios y al grupo permiso de ejecución del archivo `sube.sh` —que debe estar en el directorio desde el cual se da el comando.

chmod a-w deu.txt

Quita el permiso de escritura en el archivo `deu.txt` tanto al dueño, como al grupo, como a los demás usuarios. Este mismo resultado puede obtenerse con el comando **chmod -w deu.txt**. Cuando no se especifican usuarios **chmod** toma por defecto todos los usuarios.

3. Además de estos tipos de permiso, en Linux, hay otros que pueden aplicarse a programas y directorios para brindar algo de seguridad o afectar el desempeño. Por ejemplo el permiso `s` (set-user-ID) en un programa hace que la identificación del usuario que ejecuta el programa cambie por la identificación del dueño del programa durante su ejecución.

chmod u=rxw,g=rx,o= textos

Cambia permisos del archivo (o directorio) textos, el usuario puede leer, ejecutar y escribir, el grupo puede leer y ejecutar mientras que los demás usuarios no tienen permisos.

El dueño de un archivo pueden ser modificados sólo por el administrador del sistema con el programa **chown**. Un usuario que pertenezca a varios grupos puede cambiar el grupo de uno de sus archivos a alguno de los grupos a los que perteneza con el program **chgrp**, por ejemplo

```
chgrp estudiantes tarea1.txt
```

Cambiará el grupo del archivo `tarea1.txt` a `estudiantes`. Los grupos a los cuales un usuario pertenece son mostrados por el programa `groups`.

2.1.2. Lecturas recomendadas: Archivos y permisos

- En estas guías puede consultar más sobre permisos y administración, ver Administración de usuarios.
- Cada uno de los comandos para administrar archivos es muy flexible y configurable por medio de opciones, recomendamos de forma especial consultar las páginas **man** de cada uno de ellos.
- La jerarquía de directorios, los archivos y los permisos se organizan en las particiones de un disco duro o en un disquette empleando un sistema de archivo. Puede conocer algo más sobre el tema en la guía Administración de archivos o con **man fs**.
- Como complemento a esta lectura, puede consultar varias secciones de la documentación de fileutils con **info fileutils** o en línea en http://www.gnu.org/manual/fileutils/html_mono/fileutils.html.

2.1.3. Ejercicios: Archivos y permisos

1. ¿En su directorio personal hay archivos de configuración? En caso afirmativo cite algunos.

2. Estando en un intérprete de comandos vaya al directorio `/usr/share/pixmaps` y liste con **ls** y caracteres comodín todos los archivos con extensión `.xpm`. Desde bash, copie uno de esos archivos a su directorio con el comando **cp**. (Ayuda: busque ayuda sobre el comando **cp** y emplee ~ para referirse a su directorio).

3. Para mejorar la seguridad de sus archivos y evitar que otras personas puedan consultarlos, quite a su directorio el permiso de ejecución para el grupo y para otros usuarios.

4. Revise los directorios de los usuarios del sistema en `/home`. Identifique y liste directorios con permisos de seguridad deficientes. Intente cambiar los permisos de archivos o directorios de otro usuario.

2.2. Información personal y comunicación con otros usuarios

Indicadores de Logros:

- Mantiene su información pública y puede consultar la de otros usuarios.
- Puede comunicarse con otros usuarios empleando **talk**.
- Puede enviar y recibir correos con **mail**.

2.2.1. Lectura: Información personal y comunicación con otros usuarios

Uno de los propósitos de una red de computadores es facilitar la comunicación entre usuarios. En la red de su colegio usted podrá comunicarse con otros usuarios, ver información de ellos y publicar información sobre usted. Aún si emplea un computador con Linux que no esté conectado a una red, podrá experimentar empleando como nombre de máquina localhost[4]

2.2.1.1. Información pública y usuarios

Cada usuario de una red Linux puede publicar cierta información personal con el programa **chfn**, puede poner su plan de actividades en el archivo `~/.plan` y los proyectos que esté realizando en el archivo `~/.project`. Es importante que mantenga actualizada su información para que sea de utilidad para el colegio y otros usuarios de la red.

El programa **finger** permite ver la información que los usuarios han publicado y también permite ver qué usuarios están usando un sistema [5]. Los siguientes ejemplos, muestran como puede usarse **finger** para examinar:

1. La información de un usuario conectado a la misma máquina que usted
2. Los usuarios conectados a otra máquina
3. La información de un usuario de otra máquina

```
finger pepe
finger @servidor.colegio.edu.co
finger pepe@servidor.colegio.edu.co
```

El tercer ejemplo muestra la *dirección electrónica* de un usuario en una red, usted como usuario tendrá una dirección como esa que podrá ser usada por otros usuarios para comunicarse con usted con **talk** o con **mail**. Para formar su dirección electrónica añada a su login el símbolo `@` y el nombre del dominio de su red (e.g `agarcia@click.micolegio.edu.co`). El nombre del dominio debe preguntarlo al administrador de red.

4. El nombre localhost es normalmente asociado con la dirección IP 127.0.0.1 que corresponde a su propio computador (ver Sección 6.2.1.3).
5. Además de **finger** para examinar usuarios conectados a la misma máquina que usted puede emplear los comandos **who** y `w`.

2.2.1.2. Comunicación con usuarios que están conectados simultáneamente.

El programa **talk** permite comunicarse con un usuario que esté conectado al mismo computador que usted o en otro (mientras estén conectados por red). El programa **mesg** permite a un usuario activar o desactivar peticiones de **talk** (**mesg y** las activa y **mesg n** las desactiva). A continuación se presentan ejemplos, de como solicitar una conexión del **talk** a un usuario en el mismo computador o en otro:

```
talk pepe
talk pepe@purpura.micolegio.edu.co
```

Cuando haga la solicitud de conexión, si el otro usuario está aceptando mensajes será informado de su solicitud junto con instrucciones para establecer la comunicación (tendrá que responder con **talk** seguido de la dirección de quien hizo la solicitud) [6]. Para terminar una conexión de **talk** puede emplear la secuencia de teclas **Ctrl-D** —secuencia que significa fin de archivo—, o bien interrumpir el programa con **Ctrl-C**.

2.2.1.3. Correo electrónico

El programa **mail** permite enviar y recibir correos a otros usuarios. A diferencia de **talk**, el receptor del mensaje no necesita estar conectado mientras se envia el correo, pues este será almacenado en una casilla postal que el receptor podrá revisar cuando lo desee.

Para enviar correo a un usuario de su red puede usar:

`mail` *`login@dominio`*

Remplazando *`login`* con el login del destinatario y *`dominio`* por el dominio de su red (sin hay varios destinatarios puede separarlos con espacios). Tras esto el programa **mail** le pedirá el título [7] de su correo y le permitira escribir el mensaje. Teclee el texto y cuando termine escriba una línea con el caracter punto '.', o bien presione **Control-D**.

Para leer los correos que han llegado a su casilla postal, use:

`mail`

Al hacerlo el programa mail buscará correos en su casilla y en caso de haber le permitirá leerlos. Para ver los encabezados de los correos que haya pendientes teclee **h**, para leer el siguiente correo presione **Return**, para borrar un mensaje que ya leyó presione **d**. Puede consultar otros comandos del programa mail tecleando **?**.

Si desea redirigir los correos que llegan a la casilla de una cuenta a otra puede establecer la dirección a la cual reenviar en el archivo `~/.forward`.

6. Para enviar mensajes a usuarios conectados simultaneamente con usted en su mismo computador puede usar los comandos **write** y **wall**, con los cuales puede enviar un mensaje a uno o todos los usuarios –debe terminar el mensaje presionando **Ctrl-D**.
7. Título: en inglés *Subject*

2.2.1.4. `mutt` y `procmail`

Dada la popularidad del correo electrónico hay varios programas en modo texto y gráficos que le permiten leer y escribir correo electrónico además de **mail**. Eventualmente en su computador contará con el cliente de correo mutt, que es altamente configurable y que junto con procmail y algo de paciencia para configurarlos pueden ayudarle bastante a organizar su correo electrónico.

Al iniciar **mutt** presenta los correos que estén en su casilla, le permite desplazarse de uno a otro con flechas o con las letras **j** y **k** (cómo en vi), podrá consultar uno con **Enter**, salir con **q**, ver ayuda sobre comandos con **?**, eliminar uno con **d**, iniciar un correo (o continuar editando algún correo pospuesto) con **m**. Una vez este viendo un correo puede responder con **r**, reenviarlo con **f** o pasarlo como entrada a un comando con **|**. Cuando componga un mensaje puede posponer, puede poner adjuntos (con **a**) o emplear programas como PGP o GPG para firmar o encriptar sus mensajes (con **p**). Para organizar los correos que reciba, puede emplear carpetas, para pasar de uno a otro emplee **c**, para organizar una carpeta (por fecha, emisor, destinatario, tamaño, tema o hilos) emplee **o**. Si ordena una carpeta por hilos podrá emplear otros comandos como **Ctrl-D** para borrar un hilo completo, **Ctrl-P** para pasar al anterior, **Ctrl-N** para pasar al siguiente.

mutt puede configurarse en el archivo `~/.muttrc`, un ejemplo se presenta a continuación (las líneas iniciadas con # son comentarios):

```
# Para establecer una dirección en la que se prefieren respuestas
my_hdr Reply-To: Gabriel Torres
set reply_to=yes
unset reply_self

# Para emplear vim como editor de mensajes y editar el encabezado también
set edit_headers=yes
set editor=vim

# Para establecer algunas carpetas
mailboxes =/var/mail/grator
mailboxes =/home/grator/Mail/tareas
mailboxes =/home/grator/Mail/familia
mailboxes =/home/grator/Mail/paz
```

Si desea que su correo sea automáticamente organizado en carpetas cuando llegue puede emplear procmail y sincronizarlo con su configuración de mutt. procmail puede manejar automáticamente todo correo que reciba en una cuenta y realizar con este las acciones que usted describa. Para que separe automáticamente los correos en las carpetas del ejemplo anterior debe configurar y crear varios archivos. Cree la carpeta `~/.procmail` en ella el archivo `rc.maillists` con la configuración de los patrones por buscar en los correos y la carpeta en la que deben quedar:

```
:0:
* ^FROM.*fip@ideaspaz.org.*
paz

:0:
* ^TO.*colombia-paz@colnodo.apc.org.*
paz
```

```
:0:
* ^TO.*colombia@derechos.net.*
paz

:0:
* ^TOstructio.*
tareas
```

La última línea indica que deben dejarse correos enviados a direcciones que incluyan la palabra `structio` en la carpeta `tareas`.

El archivo `~/.procmailrc` debe contener algo como:

```
LINEBUF=4096
VERBOSE=off
MAILDIR=$HOME/Mail/
FORMAIL=/usr/local/bin/formail
SENDMAIL=/usr/sbin/sendmail
PMDIR=$HOME/.procmail
LOGFILE=$PMDIR/log
INCLUDERC=$PMDIR/rc.maillists
```

La línea **LOGFILE=$PMDIR/log** indica que debe registrarse toda acción de procmail en `~/.procmail/log`, después de configurar este servicio puede ser mejor quitarla. Si en su sistema procmail fue configurado para ser llamado por el MTA no tendrá que hacer más (e.g en la configuración por defecto de exim así ocurre). Si su MTA no fue configurado para ejecutar procmail puede crear el archivo `~/.forward` con:

```
"|IFS=' ' && exec /usr/bin/procmail -f- || exit 75 "
```

2.2.2. Lecturas recomendadas: Información personal y comunicación con otros usuarios

- Páginas del manual de los programas **chfn**, **finger**, **mail**, **talk**, **write** y **mesg**. Puede consultar la página **man** de cada uno o en Internet buscar, por ejemplo en: http://www.openbsd.org/cgi-bin/man.cgi
- Para conocer más sbre mutt puede consultar la página del manual o la documentación disponible en Internet: http://www.mutt.org/doc/manual/
- Para concer más sobre procmail puede consultar la página del manual o la documentación disponible en Internet: http://www.procmail.org/
- Puede ver ejemplos de configuración de mutt y procmail en: http://linux.ucla.edu/guides/mailguide.php3.

2.2.3. Ejercicios Información personal y comunicación con otros usuarios

1. Modifique su información personal con el programa **chfn**. En vez de "*Room Number*" ponga el grado en el que está, en vez de "*Work Phone*" ponga la dirección de su casa y en "*Home Phone*" el teléfono de su casa.

2. Escriba sus archivos `~/.plan` y `~/.project`.

3. Con el comando **finger**, revise información sobre usted mismo.

4. Revise la información del usuario root y de otros usuarios que empleen su misma máquina.

5. Busque otros usuarios conectados a su máquina.

6. *Opcional*: si tiene la oportunidad de conectarse a Internet revise la información accequible con finger en quake@geophys.washington.edu.

7. Abra dos terminales, en una de ellas comuníquese por **talk** con usted mismo y responda en la otra terminal.

8. Compruebe que el comando **mesg** funciona. (Ayuda: **mesg n** e intente el ejercicio anterior, recuerde volver a habilitar mensajes con **mesg y**).

9. Busque otros usuarios en su sistema o en otra máquina conectada por red y comuníquese con ellos.

10. Envíese un correo a usted mismo y después lealo.

2.3. Servicios de la Intranet

Indicadores de Logros:

- Emplea programas en otros computadores con telnet, ssh y X-Window.
- Transmite archivos de un computador a otro empleando **ftp**.
- Consulta el web de su institución y publica su página personal.

2.3.1. Lectura: Servicios de la Intranet

Además de permitirle comunicarse con otros usuarios, puede emplear la red de su colegio (i.e la intranet) para conectarse a otros computadores y emplear servicios que ofrecen. En esta sección se presentan brevemente 4 de tales servicios: **telnet** y **ssh** para emplear un interprete de comandos de otro computador, **ftp** para transmitir archivos y el web que permite publicar y consultar información.

2.3.1.1. telnet y ssh

Para iniciar una sesión con un interprete de comandos de otro computador, puede emplear el comando **telnet** seguido del nombre de la máquina en la que desea trabajar, por ejemplo si desea conectarse a la máquina *`purpura.micolegio.edu.co`*:

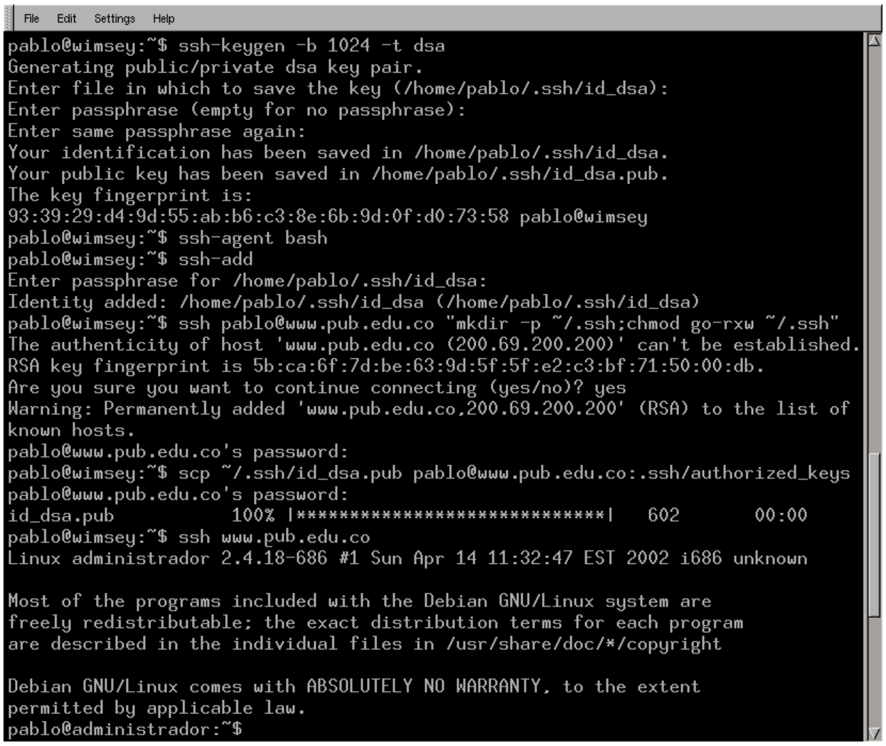
```
telnet purpura.micolegio.edu.co
```

Una vez conectado podrá ingresar su login y clave en esa máquina, y entonces se iniciará una sesión en modo texto como las que ve en las consolas virtuales (ver Lectura Sistema de usuarios y manejo de clave). La información que transmita (incluyendo su clave) no será protegida o encriptada y podría ser vista en otros computadores por los que transite la información (esto sólo puede hacerse con herramientas apropiadas i.e. *sniffers*, y desde la cuenta del administrador del sistema).

Una alternativa más segura para **telnet**, pero que requiere más recursos del computador es **ssh** que encripta la información antes de transmitirla, que autentica la máquina a la cual se conecta y que puede emplear mecanismos de autenticación de usuarios más seguros. Para conectarse como usuario `paz` a la máquina `purpura.micolegio.edu.co`:

```
ssh -l paz  purpura.micolegio.edu.co
```

ssh autenticará la máquina a la cual se está conectado, comparando la llave pública de esa máquina con alguna de las que conozca, si la llave pública de la máquina a cambiado o si es la primera vez que se conecta a ese sitio, **ssh** le pedirá confirmar la nueva llave pública. Después le pedirá su clave en ese servidor (o si ha configurado autenticación con llaves simétricas pedirá su clave RSA) para finalmente dejarlo en una sesión en modo texto, tal como **telnet**, pero que encripta toda la información que transmita.



Si en el computador al que se conecta hay programas para X-Window (con interfaces gráficas) puede iniciarlos y verlos en su computador —esto incluso puede hacerse con otros sistemas operativos si configura un servidor de X-Window. Para lograrlo se deben seguir cierto pasos que se ejemplifican a continuación, suponiendo que usted está conectado al computador `rojo.micolegio.edu.co` y abrió una sesión con **telnet** en el computador `purpura.micolegio.edu.co`:

1. En el computador *rojo.micolegio.edu.co* debe estar corriendo un servidor de X-Window (si usted está trabajando en el escritorio Gnome ya está funcionando su servidor X-Window). El computador *purpura.micolegio.edu.co* actuará como cliente de X-Window.
2. En *rojo.micolegio.edu.co* debe habilitar conexiones de X-Window desde la máquina en la cual ejecutará la aplicación. Si usa **telnet** o si **ssh** no se configuró para retransmitir información de X-Window, esto lo puede hacer con el comando **xhost**:

   ```
   xhost +purpura.micolegio.edu.co
   ```

3. En el computador en el cual ejecutará el programa (el cliente) debe indicar en la variable de ambiente `DISPLAY` [8], el nombre del servidor de X-Window:

   ```
   export DISPLAY=rojo.micolegio.edu.co:0
   ```

   Si emplea **ssh** normalmente no requerirá este paso.
4. Finalmente puede iniciar la aplicación gráfica en *purpura.micolegio.edu.co* por ejemplo:

   ```
   xeyes
   ```

2.3.1.2. Ftp

Puede emplear el programa **ftp** para enviar archivos de un computador a otro en una conexión no encriptada. Para iniciar una conexión, desde un interprete de comandos teclee **ftp** seguido del nombre del computador al cual se desea conectar:

```
ftp purpura.micolegio.edu.co
```

A continuación **ftp** pedirá nombre de usuario y clave de un usuario en el computador al cual se está conectado. Cuando las de quedará e un interprete de comandos especializado. Entre los comandos que puede emplear están:

quit

Que permite terminar la sesión ftp.

?

Para ver una lista de ordenes para ftp. Si a continuación se da el nombre de un comando se obtendrá ayuda especifica para ese comando.

ls

Para ver el listado de archivos disponibles en el computador remoto.

cd

Para cambiarse de directorio en el computador remoto.

8. Una variable de ambiente es información mantenida por el interprete de comandos y asociada con un nombre (ver Scripts básicos para bash).

cdup

Permite pasar al directorio padre, por ejemplo si en el computador al cual se conectó está en `/home/pepe` el comando **cdup** lo dejará en `/home`. Es análogo a la orden **cd ..** en un interprete de comandos.

mkdir

Para crear directorios en el computador remoto (si tiene permiso de escritura).

pwd

Para examinar el directorio en el que está en el computador remoto.

delete

Para borrar un archivo del computador remoto.

lcd

Para cambiar el directorio de trabajo en el computador local.

!*`comando`*

Ejecuta el comando especificado en el computador local, por ejemplo para examinar los archivos del computador local **!ls** para examinar el directorio de trabajo **!pwd**.

get

Para transmitir un archivo del computador remoto al local. Por ejemplo **get j.txt**.

put

Para transmitir un archivo del computador local al computador remoto.

mget

Para trasnmitir varios archivos del comptuador remoto al local. Por ejemplo `mget *.jpg`.

reget

Permite continuar la transmisión de un archivo, después de una interrupción. Por ejemplo `reget inmenso.gz`.

Algunos computadores (por ejemplo en el servidor o en Internet en *`rtfm.mit.edu`*) pueden haber repositorios de archivos accequibles por ftp anónimo. Esto significa que puede conectarse cualquier usuario empleando como login `anonymous` (o `ftp`) y como clave su dirección de correo (normalmente cualquier secuencia de caracteres que incluya '@' servirá). Es posible que por seguridad la red de su colegio esté configurada para recibir sólo conexiones ftp anónimas. En tal caso puede emplear **scp** como se explica más adelante en esta sección.

Como alternativa a ftp para descargar un sitio web entero, puede emplearse wget. Este programa recibe un URL y puede descargarlo así como todos los documentos que este enlace (y los que los documentos enlazados enlacen de forma recursiva). Por ejemplo para descargar recursivamente hasta 5 niveles de profundidad del sitio `structio.sourceforge.net`:

```
wget -r -l 5 http://structio.sourceforge.net
```

> **Aviso**
>
> Si el sitio que está descargando cuenta con mucha información, o si especifica un nivel de recursión muy alto, con wget consumirá bastante ancho de banda y llenará rapidamente el espacio disponible.

Si la transmisión se interrumpe y el servidor con el que se conecta lo soporta, podrá continuar después con la opción `-c` (análogo a `reget` en ftp).

Una alternativa segura a **ftp**, para transmitir archivos de su cuenta en un máquina a otra cuenta en otra máquina es **scp** (herramienta disponible sólo junto con **ssh**). Se usa de forma análoga a **cp** (ver Archivos y permisos), sólo que el archivo fuente (o los archivos fuente) y la vía destino pueden incluir el nombre del usuario y la máquina. Por ejemplo para copiar de la máquina desde la cual da el comando un archivo `carta.txt` a la cuenta del usuario *`paz`* en la máquina *`purpura.micolegio.edu.co`*:

```
scp carta.txt  paz@purpura.micolegio.edu.co:/home/paz
```

De forma análoga a **ssh**, este comando autenticará la máquina a la que se conecté, le permitirá autenticarse como usuario con su clave (que transmitirá encriptada a diferencia de ftp) y encriptará la información que se transmita.

2.3.1.3. Web

Un hipertexto es un documento con enlaces a secciones de otros documentos o del mismo. El nombre *web* se da a un conjunto de hipertextos disponibles en uno o más servidores interconectados. Cada hipertexto puede localizarse con un URL [9] único en la red que lo identifica y que un usuario puede emplear para consultarlo con un cliente web (también llamado *navegador* [10]).

El URL de los hipertextos comienza con `http://` a continuación el nombre de la máquina y a la ruta del hipertexto (e.g. `http://servidor.micolegio.edu.co/` o http://structio.sourceforge.net/guias/index.html (http://structio.sourceforge.net/guias/index.html)). Los hipertextos de usuarios pueden localizarse con una virgulilla y el login del usuario después del nombre de la máquina (e.g. `http://click.micolegio.edu.co/~agarcia`).

Como usuario usted puede publicar su página personal y otros archivos HTML para que sus compañeros la vean, creando en su directorio personal un subdirectorio `public_html` y ubicando allí sus hipertextos. Ese directorio debe tener permisos de lectura y ejecución para todos los usuarios, su directorio personal debe otorgar permiso de ejecución para todos los usuarios. En ese directorio (i.e `public_html`) puede tener varios hipertextos interconectados, el principal debe llamarse `index.html` y será el presentado cuando se use desde un navegador el URL: `http://servidor.micolegio.edu.co/~agarcia`

9. URL es sigla de *Universal Resource Locator*, que puede traducirse como Localizador universal de recursos. Como el nombre lo índica un URL identifica de forma única un recurso en una red; por ejemplo los hipertextos HTML comienzan con URLs de la forma `http://`, los archivos disponibles por ftp con `ftp://`. Por ejemplo el URL del archivo `index.html` disponible en la raíz del repositorio de ftp anónimo de rtfm.mit.edu es: `ftp://rtfm.mit.edu/index.html`, mientras que el archivo `doc.txt` disponible en el directorio `escritos` de la cuenta del usuario `paz` con clave `miclave` en el computador servidor.micolegio.edu.co es: `ftp://paz:miclave@servidor.micolegio.edu.co/escritos/doc.txt`
10. Navegador: del inglés *browser*, hay varios que funcionan en modo texto: links, lynx y w3m que funcionan en modo texto. Hay también navegadores gráficos que requieren más o menos recursos (memoria y procesador), entre los que más requieren por ofrecer más caraterísticas: mozilla, galeon, netscape 6, konqueror; entre los que menos requieren: amaya, netscape 4 o versiones previas.

(remplazando *agarcia* por su login). Los demás hipertextos HTML (extensión .html), podrán consultarse con URLs como `http://servidor.micolegio.edu.co/~agarcia/dibujos.html`.

Sus hipertextos deben estar escritos en un lenguaje de hipertextos llamado HTML, que no se explicará en detalle en estas guías, aunque el ejercicio 2 presenta algunas nociones (la extensión de los archivos escritos en tal lenguaje es `.html`).

2.3.2. Lecturas recomendadas: Servicios de la Intranet

- Las páginas del manual de **telnet** y **ftp**. Alternativamente a **telnet** podría usar **rsh**.
- Si tiene acceso a Internet puede aprender sobre HTML en http://www.w3.org/MarkUp/Guide/.

2.3.3. Ejercicios: Servicios de la Intranet

1. Conéctese a su propia cuenta en otro computador de la intranet de su colegio empleando tanto **telnet** como **ssh** —si en la red ya está configurado NFS y NIS como se describe en estas guías, no notará diferencia en su directorio personal al conectarse a otro computador de la intranet, emplee el comando **hostname** o examine por ejemplo archivos del directorio `/etc` (como `/etc/hostname`) para determinar en que computador está.

2. Inicie una sesión de ftp anónimo con el servidor de su colegio.

3. Opcional: Si tiene la oportunidad de conectarse a Internet puede emplear muchos repositorios de ftp anónimos. Recomiende algunos a sus compañeros.

4. Consulte el web de su colegio (pregunte el nombre de la máquina a su administrador)

5. Si tiene acceso a Internet hay muchos sitios web que puede consultar. Recomiende algunos a sus compañeros.

6. Cree su página personal para que pueda ser visitada por sus compañeros. Para comenzar puede partir de la siguiente porción de código HTML como archivo `~/public_html/index.html` –recuerde ajustar los permisos de su directorio y del directorio `public_html`:

```
<html>
<head><title>
Página personal de prueba
</title></head>

<body>
<h1>Software de libre redistribución</h1>
Viva la colaboración!
<i>Italicas</i> <b>Negrilla</b> <font color="red">Otro color</font>
<font size="+4">Otro tamaño</font>
</body>
</html>
```

7. Visite la página personal de otro miembro de su comunidad educativa.

2.4. Edición de texto

Indicadores de Logros:

- Emplea vi para editar textos.
- Emplea **emacs** para editar textos y conoce otras de sus capacidades.
- Corrige ortografía desde su editor de texto y emplea diccionarios.

2.4.1. Lecturas: Editores de texto

Aunque hay varios editores gráficos para X-Window y Gnome, dos de más populares son vi y emacs, ambos funcionan en modo texto y ambos fueron diseñados para lograr eficencia en la edición de textos.

Cada uno de estos editores tiene ventajas y desventajas frente al otro. vi requiere muy poca memoria, es veloz y busca minimizar el desplazmiento de los dedos, por su parte emacs ofrece gran diversidad de servicios adicionales a la edición de texto (calendario, correo electrónico, calculadora, comparador de archivos, juegos y muchas otras).

En esta sección se presenta una introducción extra breve a cada uno de estos editores y a algunas herramientas como correctores ortográficos y diccionarios. Los dos editores tienen muchisima más funcionalidad que la presentada aquí y para disfruta más la edición con estos editores es necesario consultar la documentación completa (se presentan referencias en las lecturas de profundización).

2.4.1.1. Vi

Para editar un texto (digamos `ma.txt`) con el editor vi, teclee desde un intérprete de comandos:

```
vi ma.txt
```

vi es un editor con dos modos: edición y comandos. En el modo de edición el texto que ingrese será agregado al texto, en modo de comandos las teclas que oprima pueden representar algun comando de vi. Cuando comienze a editar un texto estará en modo para dar comandos el comando para salir es **:** seguido de **q** y **ENTER** –con ese comando saldrá si no ha hecho cambios al archivo o los cambios ya están salvados, para salir ignorando cambios **:q!** seguido de **ENTER**.

Puede insertar texto (pasar a modo edición) con varias teclas:

i

Inserta texto antes del caracter sobre el que está el cursor.

a

Inserta texto después del caracter sobre el que está el cursor.

I

Inserta texto al comienzo de la línea en la que está el cursor.

A

Inserta texto al final de la línea en la que está el cursor.

o

Abre espacio para una nueva línea después de la línea en la que está el cursor y permite insertar texto en la nueva línea.

O

Análogo al anterior, pero abre espacio en la línea anterior.

Para pasar de modo edición a modo de comandos se emplea la tecla **ESC**, para desplazarse sobre el archivo puede emplear las flechas, PgUp, PgDn[11]. Para ir a una línea específica puede emplear **:** seguido del número de línea y **ENTER**, para ir al final de la línea en la que está el cursor **$**,para ir al comienzo **0**. Para buscar un texto **/** seguido del texto que desea buscar y **ENTER**. Después de hacer cambios puede salvarlos con **:w** o para salvar y salir puede emplear **ZZ**. Para ejecutar un comando del interprete de comandos puede emplear **:!** seguido del comando y **ENTER** (e.g `:!ls`).

En su sistema puede haber diversas versiones de vi, recomendamos vim que ofrece extensa ayuda y cuenta con varias extensiones, por ejemplo:

Digrafos

Útil para producir caracteres del español, si el teclado o la configuración del mismo no lo permiten (hay otras formas de lograrlo ver Teclado, ver Localización y configuración de algunos programas). Presione **Control-K** y depues una secuencia de dos teclas de acuerdo al caracter, e.g 'a produce á, n? produce ñ, u: produce ü, !I produce ¡, ?I produce ¿. Pueden verse otros caracteres con el comando `:digraphs`.

Resaltado de sintaxis

Permite ver con colores que resaltan la sintaxis de algunos tipos de archivos (por ejemplo fuentes de DocBook, C, LaTeX). Se activa con `:syntax enable`

Indentación automática

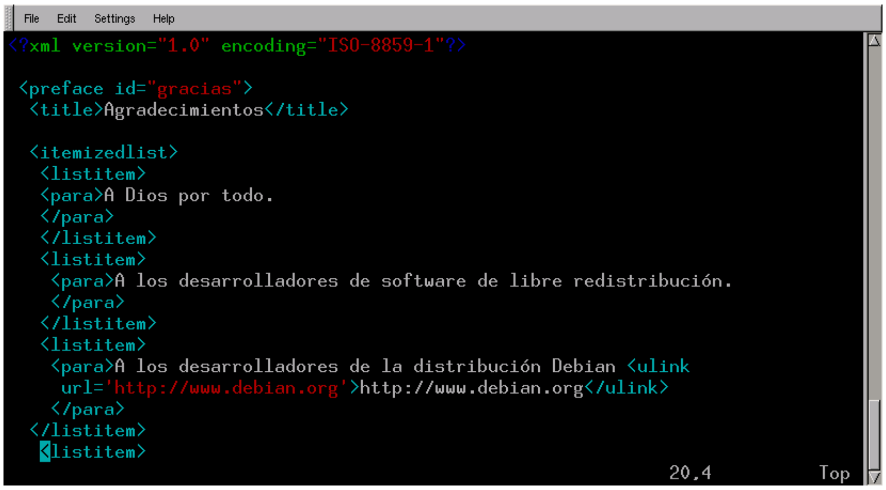
Permite indentar automáticamente diversos tipos de archivos por como XML, C, LaTeX, XSLT. Se activa con `:filetype indent on` y `:filetype indent on`.

Archivo de configuración

Cada usuario puede tener sus propias configuraciones en el archivo `~/.vimrc`, que consta de líneas con comandos o comentarios. Los comandos no deben iniciar con ':' y los comentarios se colocan en líneas completas o al final de una línea iniciando con comillas. Por ejemplo:

```
" Este es un ejemplo de archivo de configuración para vim
syntax enable
filetype on
filetype plugin on
filetype indent on
```

11. La función de las flechas también la cumplen las teclas **h**, **j**, **k** y **l**



2.4.1.2. Emacs

Para editar un texto (nuevamente `ma.txt`) con emacs teclee:

```
emacs ma.txt
```

que en ambiente X-Window abrirará un ventan nueva o para abrir el archivo en una terminal texto:

```
emacs -nw ma.txt
```

Una vez en emacs podrá desplazarse sobre el texto con las flechas y escribiendo insertará texto (no hay modos). Los comandos en emacs se indican con secuencias de teclas que comienzan con la tecla **ESC** (llamada Meta y abreviada como M en la documentación de Emacs) o teclas que se oprimen simultaneamente con **Control** (abreviada con C en la documentación de Emacs). Algunas secuencias de teclas útiles son:

C-x C-c

Para salir de Emacs. (Note que son dos secuencias de teclas, primero `Control` simultáneamente con **x** y después **Control** simultáneamente con **c**).

C-x C-s

Para salvar el archivo que se está editando.

C-e

Para ir al final de una línea.

C-a

Para ir al comienzo de la línea.

M-<

Para ir al comienzo del archivo.

M->

Para ir al final del archivo.

C-s

Para buscar incrementalmente un texto.

C-f

Avanzar a la derecha o a la siguiente línea si se está al final de una. Equivalente a flecha a la derecha.

C-b

Avanzar a la izquierda o al final de la línea anterior si está al comienzo de una. Equivale a flecha a la izquierda.

C-n

Avanzar a la línea siguiente, equivalente a flecha hacia abajo.

C-p

Pasar a la línea anterior, equivale a flecha hacia arriba.

M-x

Para dar un comando especial a Emacs, entre los comandos especiales están: **info** para consultar dentro de emacs páginas del manual info, **man** para consultar páginas del manual, **shell** para abrir una terminal dentro de emacs.

En parte inferior de este editor verá una línea de modo (tiene varios guiones) y debajo de esta el minibuffer o área de eco en la que Emacs recibe y envia información del usuario. La línea de modo puede aparecer con dos asteríscos al comienzo para indicar que el texto editado se ha modificado y no se ha salvado.

Hay algunas extensiones para emacs que dan nueva funcionalidad, por ejemplo gnus para leer noticias y correo (si está disponible lo inicia con **M-x**-gnus), w3 para navegar en el web (lo inicia con **M-x w3**) y emacspeak que lee la información presentada en Emacs con una tarjeta de sonido –muy útil para personas invidentes.

2.4.1.3. Corrección ortográfica

Para emplear esta funcionalidad en su sistema deben estar instaladas la lista de palabras de español y los archivos para español de ispell (ver Localización y configuración de algunos programas) Desde la línea de comandos puede emplear el programa look para verificar que una palabra esté en la lista de palabras de español:

```
look "amoroso" /usr/share/dict/spanish
```

También desde la línea de comandos puede emplear ispell para revisar y corregir interactivamtne un texto en español:

```
ispell -d spanish carta.txt
```

O un documento en LaTeX agregando la opción `-t` o en HTML o DocBook con la opción `-h` (ver Sistemas para preparar documentos).

Si emplea el editor Emacs, y en su sistema ya está instalado el diccionario en español para ispell de forma apropiada para Emacs (ver Localización y configuración de algunos programas) podrá elegir castellano del menú Edit+Spell, para revisar el documento completo desde el mismo menú puede elegir Check Buffer o para revisar la palabra sobre la que está el curso use **M-$**.

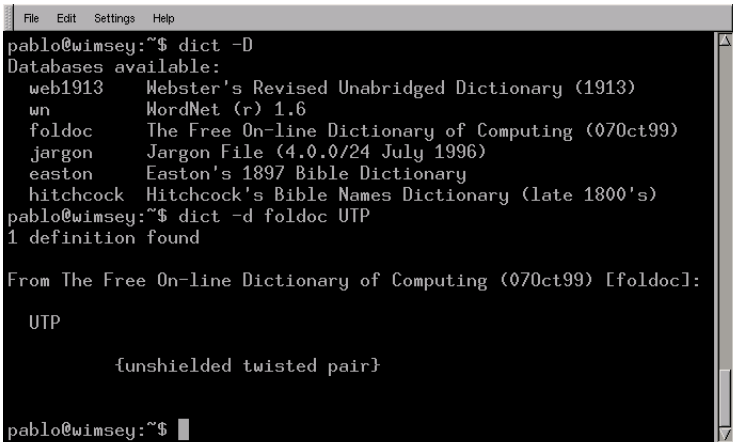
Si emplea el editor vim, puede agregar a su archivo de configuración `~/.vimrc`, un script que implemente la corrección de ortografía. Por ejemplo hay un script bien documentado en: http://www.stripey.com/vim/spelling.html

Modifique el idioma (let IspellLang = 'spanish') y la ubicación de la lista de palabras en español: **set dictionary+=/usr/share/dict/spanish**. Una vez lo agregue, e inicie una sesión de vim, podrá revisar el documento completo con **F9**, las palabras con errores quedarán resaltadas (con **F10** borra el resaltado).

2.4.1.4. Diccionarios

Hay algunos diccionarios en inglés, tanto generales como especializados. Puede revisar la lista de los que están instalados en su sistema con:

```
dict -D
```



Puede consultar información sobre el diccionario easton con **dict -i easton**, para buscar la palabra "life" en todos los diccionarios **dict life** o sólo en easton con **dict -d easton life**

Puede ver más sobre dict y los diccionarios en la sección sobre configuración de programas (Localización y configuración de algunos programas).

2.4.2. Lecturas recomendadas: Editores de texto

- Para aprender más sobre vi puede consultar la página del manual **man vi**, o los tutoriales disponibes en su sistema Debian como `/usr/doc/nvi/vi.beginner.gz` y `/usr/doc/nvi/vi.advanced.gz` (estos tutoriales fueron diseñados para ser estudiados con vi, copielos a su directorio descomprimalos y editelos con vi).

- Para aprender más sobre emacs puede consultar el tutorial (desde emacs, presione **C-h** y después **t**) o el manual completo que está disponible como documento info —Desde emacs puede verlo con **M-x info** o con **C-h i**.
- Si lo desea, puede consultar una traducción a español del tutorial de Emacs, realizada por estudiantes del colegio colombiano Gimnasio Fidel Cano en: http://www.tamarapatino.com/gfc/proyectos/traduccion/doc/TUTORIAL.es.txt En Internet puede consultar el manual de Emacs en: http://www.gnu.org/manual/manual.html

2.4.3. Ejercicios: Editores de texto

1. Copie a su directorio el tutorial para principiantes de vi, localizado en `/usr/doc/nvi/vi.beginner.gz`, descomprímalo y estudíelo desde vi.

2. Entre a emacs y estudie el tutorial.

3. Emplee ispell para corregir ortografía desde la línea de comandos, desde emacs y desde vim.

Capítulo 3. Linux desde la perspectiva de un usuario con más experiencia

Público y logros

Estudiantes, profesores, secretarias de un colegio y personas que tienen experiencia con Linux y desean aprovechar las herramientas de este sistema —suponemos que la plataforma de referencia ya está instalada y bien configurada (ver Plataforma de referencia).

- Emplea algunos sistemas para preparar documentación.
- Conoce algunas herramientas típicas de sistemas Unix y puede componerlas empleando tuberías.
- Maneja procesos y tareas.
- Emplea las facilidades que `bash` ofrece y crea scripts sencillos.
- Personaliza mejor su espacio de trabajo.

3.1. Sistemas para preparar documentos

Indicadores de Logros

- Puede examinar e imprimir textos planos y documentos PDF, PostScript y DVI.
- Puede crear documentos sencillos en LaTeX, Texinfo y DocBook.
- Genera documentos PDF, PostScript, DVI, info y HTML a partir de fuentes en DocBook, LaTeX y Texinfo.

3.1.1. Lectura: Sistemas para preparar documentos

En esta guía se presentarán brevemente tres sistemas para preparar documentos: LaTeX, Texinfo y DocBook así como la forma de imprimir textos y los documentos que prepare. Estos sistemas se diferencian de procesadores de texto gráfico en varios aspectos:

- En estos sistemas se separa el contenido y la estructura del texto de la presentación final.
- La estructura y el contenido los prepara el autor con un editor de texto usual (e.g emacs o vi), incluyendo "comandos" como parte del texto. La presentación final la realiza el sistema para preparar documentos, aunque el autor puede modificar el comportamiento por defecto de tal sistema y ajustar detalles de la presentación —sin embargo, la presentación generada por estos sistemas es de calidad profesional.
- El sistema de documentación emplea la información de la estructura, los textos, los comandos editados por el usuario y algunas reglas sobre tipografía, ortografía y organización del documento para generar un documento apropiado para cierto medio de publicación (e.g. PostScript o PDF para imprimir, o en HTML para consultar en línea).

- El programa que genera la presentación final, usualmente genera también la tabla de contenido, índices, glosarios y mantiene actualizadas las referencias del documento en sí mismo, así como las citas bibliográficas.

3.1.1.1. Impresión y formatos para impresión

Para imprimir se emplea el sistema de impresión **lpd** que permite a cada usuario enviar sus impresiones a una cola de impresiones pendientes que es atendida automáticamente y permite usar tanto impresoras locales (físicamente conectadas a su computador) como remotas (conectadas por intermedio de una red). Para **lpd** cada impresora tiene un nombre que la identifica, la impresora por defecto comunmente se llama **lp** [1], aunque si en su caso no funciona pregunte al administrador del sistema.

Aun cuando algunos programas ofrecen menús con una opción para imprimir, prácticamente todos usan en últimas el programa **lpr** que se encarga de poner la información que debe enviarse a la impresora en una cola de trabajos pendientes [2], que es atendida automáticamente (i.e de ella se envian trabajos pendientes cuando la impresora está disponible) por el programa **lpd** (ver Impresora).

Posiblemente su sistema esté configurado para permitir a **lpr** la impresión de textos planos y documentos PostScript o PDF. Para programar la impresión de un texto plano con nombre `tarea.txt` en la impresora por defecto, puede emplear:

```
lpr tarea.txt
```

o eventualmente para especificar una impresora diferente a **lp**, digamos **imp2** use la opción `-P`:

```
lpr -Pimp2 tarea.txt
```

Puede examinar la cola de sus impresiones pendientes con **lpq** que junto con los nombres de las impresiones pendientes presentará un número que la identifica. Tal número le permitirá cancelar una de sus tareas de impresión pendiente, usándolo como parámetro de **lprm**.

Para imprimir gráficas y documentos con diversos tipos de letra o colores, primero debe convertirse la información a secuencias de control particulares de su impresora. Cómo de una impresora a otra varian los secuencias de control, en Linux se emplea PostScript (que es un lenguaje apropiado para documentos por imprimir) como formato común y se usan los filtros del programa Ghostscript para traducir de PostScript al formato particular de su impresora [3]. Tal conversión en el caso de PostScript y PDF es realizada normalmente por **lpr**.

Ghostscript es un programa que puede leer documentos PostScript y PDF [4] y presentarlos en una ventana de X-Window o transformarlos al lenguaje particular de algunas impresoras. Puede experimentar con este intérprete tecleando **gs** y por ejemplo ingresando la siguiente secuencia de instrucciones en lenguaje PostScript:

```
100 100
moveto
200 200
```

1. Puede configurar otro nombre para la impresora por defecto en la variable de ambiente PRINTER (ver Ambiente y variables de ambiente).
2. La cola de sus trabajos de la impresora **lp** está en el directorio `/var/spool/lpd/lp`.
3. Hay algunas impresoras que pueden imprimir PostScript directamente, pero en general para hacer la traducción de PostScript al lenguaje de una impresora se requiere un filtro que el administrador del sistema debe configurar (see Impresora).
4. PDF (*Portable Document Format* es otro lenguaje para impresión, de documentos con gráficas y diversos tipos de letras, basado en PostScript (de la misma compañía —Adobe).

```
lineto
stroke
```

Para visualizar un documento PostScript o PDF puede emplear el programa **gv** —el cual se apoya en Ghostscript— por ejemplo:

```
gv micarta.ps
```

mostrará el documento PostScript `micarta.ps` en una ventana de X-Window y con menús le permitirá consultarlo e imprimirlo.

Para imprimir y hacer transformaciones a un PostScript (por ejemplo 2 páginas en una sola), o para convertir de otros formatos a PostScript puede emplear el programa **a2ps**:

```
a2ps --columns=2 micarta.ps
```

Tanto PostScript como PDF requieren bastante espacio para describir un documento, usualmente los documentos PDF requieren menos porque mantienen la información comprimida. Para convertir entre PostScript y PDF se emplean **ps2pdf** y **pdf2ps** [5]. Para visualizar e imprimir un PDF, además de **gv**, puede emplear **xpdf** o el programa Acrobat Reader (**acroread**).

3.1.1.2. LaTeX

Tanto LaTeX como Texinfo son extensiones a un sistema llamado TeX, desarrollado para escribir documentos de matemáticas. A continuación se presenta un ejemplo de un documento LaTeX y el resultado que se obtiene tras procesarlo.

```
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage[spanish]{babel}
\begin{document}
\author{Rupertino Gonzales}
\title{Algunas posibilidades de LaTeX}
\maketitle

\section{Elementos}
Puede estructurar el documento en capítulos, secciones, etc.
Este texto es el contenido de la primera sección de este ejemplo,
puede escribir cada párrafo en líneas consecutivas.

\subsection{Ayudas}
Puede lograr efectos como \emph{Itálicas}, \textbf{negrillas} o
cambios en el \textsf{tipo o {\small tamaño} de letra} (note
como se anidaron ambientes en este ejemplo).

Puede crear listas:
\begin{itemize}
\item Primer elemento de lista.
```

5. De acuerdo a *Printig-HOWTO* estas herramientas ofrecen la funcionalidad de las herramientas "*distiller*" de Adobe.

```
\item Segundo elemento de lista.
\end{itemize}
o tablas
\\
\begin{tabular}{|l|r|} \hline
Título 1 & Título 2 \\\hline
elemento 1 & elemento 2 \\\hline
\end{tabular}
\subsection{Ecuaciones}
LaTeX es un experto en esta materia:
\[ \int_{x=-\inf}^{\inf}e^{-|x|} \]
\end{document}
```

LaTeX ofrece plantillas para varios tipos de documentos: artículo, reporte, libro y ofrece el concepto de ambiente para indicar como presentar cierta información de acuerdo a la plantilla. En el ejemplo presentado, el tipo de documento es artículo (lo indica la línea `documentclass{article}`), y uno de los ambientes empleados es `tabular`, que genera una tabla.

Una vez edite un documento puede procesarlo con LaTeX para obtener un archivo DVI, por ejemplo para generar el archivo `documento.dvi` a partir de `documento.tex`:

```
latex documento.tex
```

El archivo DVI es apropiado para imprimir, puede imprimirlo con un comando como **dvilj**, **dvidj** o un nombre análogo que corresponda a su impresora [6]. Para visualizar un archivo DVI puede emplear el comando xdvi:

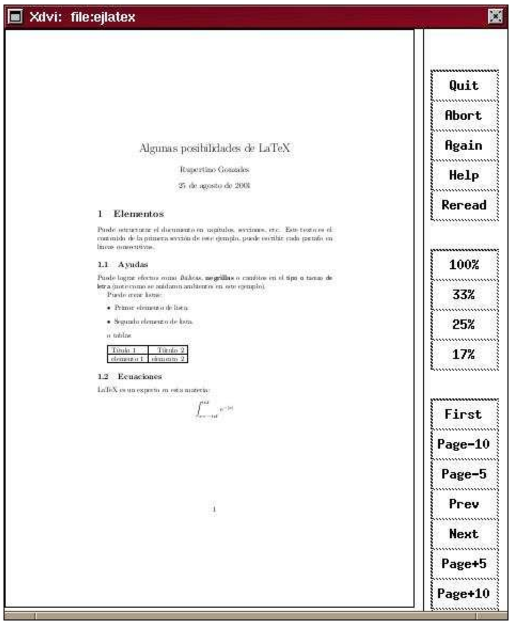
```
xdvi documento.dvi
```

y para convertirlo a PostScript puede emplear **dvi2ps**:

```
dvi2ps -c documento.ps documento.dvi
```

A continuación se presenta como se ve el ejemplo de esta sección con el programa **xdvi**.

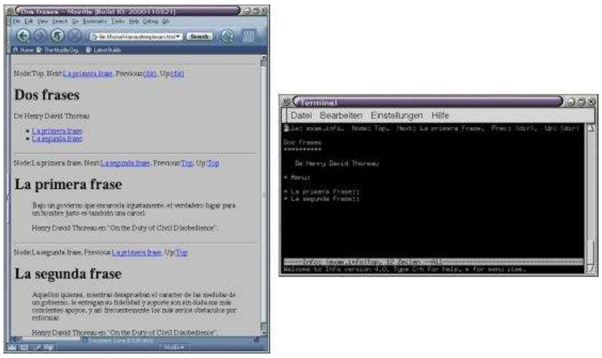
6. Puede ver una lista de posibles programas que le permitan imprimir, tecleando `dvi` desde un intérprete de comandos y presionando **Tab** dos veces (ver Lectura Scripts básicos para bash)



Existen además otros programas para convertir de LaTeX a HTML como latex2html y HeVeA. Puede encontrar más información de latex2html en http://ctan.tug.org/ctan/tex-archive/support/latex2html/ y de HeVeA en http://pauillac.inria.fr/hevea/.

3.1.1.3. Texinfo

Texinfo es otra extensión a TeX, orientada a documentos técnicos. Además de poder generar un DVI para imprimir, puede generar documentación en formato **info** (ver Búsqueda y consulta de documentación) y HTML para consultar en su computador o en Internet. A continuación se presenta un documento en Texinfo y el resultado que se ve desde un navegador, desde los programas **info** y xdvi.




```
@settitle Dos frases
@titlepage
@title Dos frases
@author De Henry David Thoreau
@end titlepage
@setchapternewpage off
```

```
@ifnottex
@node Top, La primera frase, (dir), (dir)
@top Dos frases
De Henry David Thoreau
@menu
* La primera frase::
* La segunda frase::
@end menu
@end ifnottex

@node La primera frase, La segunda frase, Top, Top
@chapter La primera frase

@quotation
Bajo un gobierno que encarcela  injustamente, el verdadero lugar para un
hombre justo es también una cárcel.
  @flushright
  Henry David Thoreau en "On the Duty of Civil Disobedience".
  @end flushright
@end quotation

@node La segunda frase,  , La primera frase, Top
@chapter La segunda frase

@quotation
Aquellos quienes, mientras desaprueban el  carácter de las medidas de un
gobierno,  le entregan  su  fidelidad y  soporte  son sin  duda sus  más
conscientes apoyos,  y así frecuentemente  los más serios  obstáculos
para reformar.
  @flushright
  Henry David Thoreau en "On the Duty of Civil Disobedience".
  @end flushright
@end quotation

@bye
```

Para generar el archivo **info** a partir de un archivo Texinfo con nombre `doc.texi`:

```
makeinfo doc.texi
```

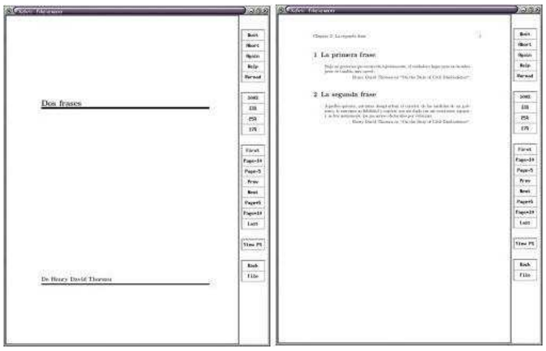
para generar el HTML:

```
      makeinfo -html doc.texi
```

y para generar el archivo DVI a partir de `doc.texi`:

```
texi2dvi doc.texi
```

A continuación se presenta el mismo documento de ejemplo de esta sección visto desde xdvi.



La versión de Texinfo disponible en Debian 2.2 no maneja del todo bien caracteres del español (vocales tildadas, eñe y diéresis), en particular **texi2dvi**. Puede sobrellevar el problema cambiando caracteres de español antes de usar **texi2dvi** por: @'a, @'e, @'{i}, @'o, @'u, @\"u, @~n (y sus versiones en mayúsculas). Esto puede hacerse automáticamente con un script y generar nuevos archivos (digamos con extensión .texes) que pueden ser pasados a texi2dvi (si una de las fuentes usa @include debe también renombrarse la extensión del archivo incluido a .texes).

3.1.1.4. DocBook

DocBook es un formato que tiene algún parecido con HTML pues ambos provienen de un ancestro común (el estándar para documentos SGML). A continuación se presenta un ejemplo:

```
<!DOCTYPE article PUBLIC "-//OASIS//DTD DocBook V3.1//EN">
<article lang="es" id="ejemplo">
 <artheader>
  <title>Un ejemplo de DocBook</title>
  <abstract>
   <para>Nociones básicas de DocBook.</para>
  </abstract>
  <author>
   <firstname>Tomás</firstname><surname>Ramírez</surname>
  </author>
  <date>2 de Septiembre de 2001</date>
  <address>tomas.ramirez@dominio.org</address>
 </artheader>

 <sect1 id="idsec1">
  <title>Título de la sección</title>
  <sect2 id="idsec1.1">
   <title>Título de subsección</title>
   <para>Los textos se encierran entre marquillas que indican como
    debe presentarse el texto, las marquillas se abren con &lt;nombre_marq>;
    y se cierran con &lt;/nombre_marq>, por ejemplo la marquilla &lt;para>
    se emplea antes de iniciar un párrafo, y el párrafo debe cerrarse con
    &lt/para>. Otro ejemplo de tag es &lt;programlisting>, cuyo efecto
    se presenta a continuación:</para>
<programlisting>
int main()
{
```

```
  int n,k;
}
</programlisting>
  </sect2>
  <sect2 id="idsec1.2">
   <title>Título de la segunda subsección</title>
   <para>Pueden emplearse  varios tipos de documentos, este es
    un ejemplo de un artículo. Cada documento puede estructurarse
    en capítulos, secciones, subsecciones.</para>
  </sect2>
 </sect1>
</article>
```

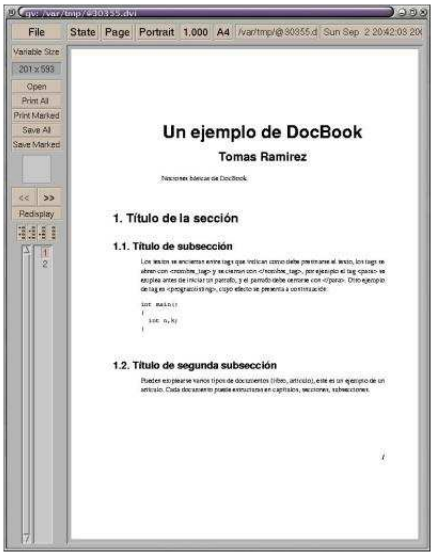
Los textos se encierran entre marquillas que indican como debe presentarse el texto, las marquillas se abren con `<nombre_marq>` y se cierran con `</nombre_marq>`, por ejemplo la marquilla `<para>` se emplea antes de iniciar un párrafo, y el párrafo debe cerrarse con `</para>`. Otro ejemplo de tag es `<programlisting>`, que permite mostrar una porción de texto en un tipo de letra diferente (apropiada para preentar programas).

```
int
main() {
int n,k;
}
```

Pueden emplearse varios tipos de documentos, este es un ejemplo de un artículo. Cada documento puede estructurarse en capítulos, secciones, subsecciones.

Suponiendo que edita el ejemplo en un archivo `ejdb.sgml`, y que en su sistema ya está instalado el paquete sgmltools-2, podrá generar un PostScript con:

```
sgmltools -b ps ejdb.sgml
```



Otra posibilidad para generar HTML y PostScript a partir de documentos en DocBook (SGML o XML) se presentan brevemente en otra sección (ver Localización y configuración de algunos programas). Para aprender DocBook recomendamos de forma especial el Tutorial de DocBook:Un enfoque integrado y a través de ejemplos (http://structio.sourceforge.net/guias/dbktut)

3.1.1.5. Uso de CVS

Eventualmente deseará crear documentos de forma colaborativa con otras personas que tengan acceso a la red de su colegio. Precisamente para esa labor, existe CVS que controla las versiones de los archivos tipo texto de un directorio (que en terminología de CVS es un módulo), ofrece un mecanismo para que varias personas trabajen simultaneamente sobre la misma información y para detectar conflictos y permitir al usuario que lo genera, resolverlo (conflicto se refiere a dos ediciones diferentes de una misma porción de texto).

Para desarrollar en grupo con CVS basta tener en cuenta:

1. Hay un repositorio central donde está la copia "oficial" más actualizada. (puede ver como crear el repositorio e importar un módulo en la sección sobre Servicio CVS). Usted ha configurado las variables de ambiente (ver Ambiente y variables de ambiente) **CVSROOT** y **CVS_RSH** con la localización del repositorio y el programa para ingresar a la máquina donde está. Las posibilidades son:

   - Si el repositorio está en la misma máquina en la que usted está trabajando en, digamos en el directorio `/var/cvs`:

     ```
     export CVS_RSH=""
     export CVSROOT=/var/cvs
     ```

   - Si el repositorio está en un directorio en otra máquina, y la otra máquina tiene un servidor ssh o rsh (ver telnet y ssh) y usted tiene cuenta en esa máquina (o su clave pública ssh está autorizada en alguna cuenta de esa máquina, ver Servicio ssh). Suponiendo que la otra máquina es *`purpura.micolegio.edu.co`*, que cuenta con ssh , que el repositorio está en el directorio `/home/pepe/cvs`, y que en esa máquina usted puede usar con ssh la cuenta *`pepe`*:

     ```
     export CVS_RSH=ssh
     export CVSROOT=pepe@purpura.micolegio.edu.co:/home/pepe/cvs
     ```

   - Si el repositorio está en un directorio de otra máquina, el protocolo pserver de CVS funciona en esa máquina y usted tiene una cuenta en esa máquina o está autorizado para usar pserver en el repositorio:

     ```
     export CVS_RSH=""
     export CVSROOT=:pserver:pepe@purpura.micolegio.edu.co:/home/pepe/cvs
     cvs login
     ```

2. Cada desarrollador debe obtener una copia de ese repositorio y dejarla en su cuenta para trabajarla. Esto es hacer un *checkout*, por ejemplo para obtener una copia local del módulo `quimica`:

   ```
   cvs checkout quimica
   ```

   En lugar de `checkout` puede emplear `co`. Si lo requiere puede pasar opciones generales a CVS antes de `checkout` (como `-z3` que indica transmitir información comprimida):

```
cvs -z3 co quimica
```

3. Cada desarrollador con alguna frecuencia debe actualizar su copia local con respecto a la del repositorio central, para mantener al dia su copia con los cambios más recientes introducidos por otras personas. Esto es hacer un *update* (es importante que lo haga antes de comenzar a hacer cambios a su copia local). Por ejemplo puede usar:

   ```
   cvs -z3 update -Pd
   ```

   Note que en este comando la opción `-z3` es una opción general de CVS, mientras que `-Pd` es particular a `update` (indica que no deben agregarse a la copia local directorios que estén vacios en el repositorio).

4. Cada desarrollador trabaja en su copia personal, y cuando completa una parte del trabajo (mejor asegurandose de no introducir errores), publica la actualización junto con un comentario (en el siguiente ejemplo es "Ortografía corregida"):

   ```
   cvs commit -m "Ortografía corregida"
   ```

   Eventualmente al intentar esta operación, si otro usuario realizó una actualización primero que tiene alguna diferencia con respecto a los cambios que se quieren agregar, CVS detecta el conflicto, modifica el archivo para resaltar los conflictos y permite solucionar el conflicto antes de repetir esta operación. Por ejemplo un conflicto puede verse como:

   ```
   «««1.3
   la presion interna de un gas es directamente proporcional a su temperatura.
    -----
   La presión interna de un gas es inversamente proporcional a la temperatura
   »»»
   ```

   Para solucionar el conflicto elimine las líneas que CVS agregó y deje la versión más precisa (o mezclelas), para este caso sólo quedaría:

   ```
   La presión interna de un gas es directamente proporcional a la temperatura.
   ```

3.1.2. Lecturas recomendadas: Sistemas para preparar documentos

- Si está interesado en PostScript puede consultar en Internet esta guía de libre redistribución: A first guide to PostScript (http://www.cs.indiana.edu/docproject/programming/postscript/postscript.html).
- LaTeX
  - En su sistema Debian puede consultar un índice de documentación relacionada con LaTeX en el directorio `/usr/doc/texmf` en particular puede consultar el archivo `index.html`. De los documentos disponibles sugerimos de forma especial "*Essential LaTeX*" e "*Introduction to LaTeX2e*".
  - En Internet puede consultar más sobre LaTeX y TeX en el completo FAQ del grupo de usuarios del Reino Unido: http://www.tex.ac.uk/cgi-bin/texfaq2html y en español en el sitio del proyecto

  CervanTeX: http://lucas.hispalinux.es/CervanTeX/CervanTeX.html. De igual forma en sus sistema puede consultar la ayuda de LaTeX (Ver **info latex**))

  - El libro clásico sobre LaTeX es de Leslie Lamport (autor de LaTeX): "*LaTeX - A document preparation System - User's guide and reference manual*", Addison-Wesley, 1985. El libro clásico sobre TeX es de Donald Knuth (autor de TeX): "*The TeX Book*", Addison-Wesley, 1986.

- Texinfo
  - En su computador puede consultar el manual info de Texinfo (Ver (**info texinfo**)).
  - En Internet puede consultar un resumen de Texinfo en: http://www2.linuxjournal.com/lj-issues/issue6/2840.html y el manual junto con más información en: http://www.texinfo.org

- DocBook
  - Recomendamos de forma especial el tutorial escrito por Jaime Irving Dávila y liberado al dominio público: "*Tutorial de DocBook: Un enfoque integrado y a través de ejemplos*". Está disponible en Internet en: http://lucas.hispalinux.es/Tutoriales/DOCBOOK/doctut/multiple-html/ y el proyecto Structio mantiene una copia en http://structio.sourceforge.net/guias/dbktut
  - La referencia por excelencia es el libro "*DocBook: The Definitive Guide*" escrito por Norman Walsh y Leonard Muellnery, y editado por O'Reilly & Associates. Este libro está disponible en Internet en: http://docbook.org/tdg/en/html/docbook.html

3.1.3. Ejercicios: Sistemas para preparar documentos

1. En el directorio `/usr/local/share/doc/AA_Linux_colegio` de su sistema encontrará una versión en PostScript de estas guías. Copiela a su directorio, examínela con el programa apropiado y después conviértala a PDF.

2. Transcriba uno o varios de los ejemplos presentados en este capítulo y genere a partir del mismo un documento para imprimir y/o consultar en Internet.

3. Opcional: El editor `emacs` cuenta con modos especiales para editar LaTeX (Ver (**info emacs**)), Texinfo (Ver (**info texinfo**)) y DocBook (ver "*Tutorial de DocBook: Un enfoque integrado y a través de ejemplos*"). Experimente con alguno de estos modos para escribir un documento.

4. Opcional: estas guías están escritas en DocBook, puede usar las fuentes tanto para aprender DocBook como para mejorar este documento. Encuentra las fuentes en el sitio de distribución: http://structio.sourceforge.net/guias/AA_Linux_colegio/.

5. Opcional: el tutorial de DocBook citado en las lecturas recomendadas está escrito en DocBook, puede emplear las fuentes para aprender más sobre DocBook. Las encuentra en el sitio de distribución: http://lucas.hispalinux.es/Tutoriales/DOCBOOK/

3.2. bash y el juego de herramientas

Indicadores de Logros

- Emplea facilidades que `bash` ofrece para operación interactiva.
- Emplea algunas herramientas para manipular textos (**cat**, **tac**, **nl**, **tail**, **head**, **sort**, **split**, **csplit** y **wc**).
- Emplea redireccionamiento y compone herramientas empleando tuberías [7].

3.2.1. Lectura: bash y el juego de herramientas

El intérprete de comandos puede ser un gran aliado de usuarios, administradores y programadores, porque permite automatizar labores rutinarias y componer diversos programas para que interactúen y realicen labores complejas.

La idea para componer diversos programas es formar "tuberías" en las que cada tubo es un programa cuya salida se conecta como entrada a otro. Esta idea se comenzará a desarrollar en esta guía, introduciendo primero algunas facilidades del intérprete de comandos bash [8] para editar comandos, después se presentarán algunos programas y comandos de bash y finalmente se presentará como se compone y se controla el flujo de datos.

3.2.1.1. Ayudas interactivas de bash (*readline* y *history*)

Para recibir información del usuario bash y otros programas emplean las librerías *readline* y *history*. *readline* brinda facilidades para escribir comandos con un teclado y *history* almacena los últimos comandos que el usuario haya dado para facilitar su reutilización. Algunas de las posibilidades que estas librerías ofrecen son:

Flechas o **Control-p**, **Control-n**

Para "navegar"sobre la historia de comandos

Completación con **Tab**

Puede emplear la tecla **Tab** para completar nombres de programas, de máquinas (mencionadas en el archivo `/etc/hosts`) o de usuarios, si hay varias posibles completaciones un segundo **Tab** presentará la lista de posibilidades. Por ejemplo: si estando en bash teclea **chm** y después presiona **Tab**, se completará el nombre del comando a **chmod**; si teclea **ls /usr/inc** y presiona **Tab** se completará la vía `/usr/include`; si teclea **finger @** y después **Tab** dos veces verá los nombres de algunas máquinas conocidas por su sistema; si teclea **ls ~** y presiona **Tab** dos veces verá los nombres de usuarios (**ls ~carlos** mostrará el contenido del directorio personal del usuario con login `carlos`).

Edición con teclas de emacs o vi

`readline` puede configurarse para efectuar algunas operaciones de edición bien con las teclas de emacs o bien con las de vi. Por ejemplo cuando las teclas de emacs están configuradas (que es la opción por defecto) puede emplear las flechas hacía atrás y adelante (o **Control-f** y **Control-b**) para

7. tubería: del inglés *pipe*.
8. Aunque hay varios intérpretes de comandos para Unix y Linux (ksh, ash, zsh, tcsh y csh), algunos muy buenos, bash es el intérprete de comandos escogido por el proyecto de estandarización de Linux: *LSB - Linux Standard Base*. http://www.lsb.org

devolverse o avanzar sobre el comando; **Control-a** para ir al comienzo; **Control-e** para ir al final; **Control-k** para cortar el resto del comando; **Control-y** para insertar el texto antes cortado. Cuando las teclas de **vi** están configuradas puede presionar **ESC** para pasar a modo comando y usar las teclas **h**, **j**, **k** y **l** (o las flechas) para moverse o **f** para buscar una letra en la línea que está editando.

Buscar comandos en la historia de comandos

Puede buscar un comando entre los que ya ha tecleado para reutilizarlo. Con el modo emacs emplee **Control-r** y comience a teclear el comando que busca –es una búsqueda incremental, en el modo vi emplee **/** y después teclee parte del comando que busca.

3.2.1.2. Algunas herramientas

Un sistema Linux cuenta con gran cantidad de herramientas, cada una de las cuales realiza una labor muy específica pero rápidamente y bien; en esta sección presentamos unas cuantas típicamente usadas para procesar textos planos. Junto con cada herramienta presentamos algunas de sus opciones y ejemplos de su uso (la documentación completa de cada herramienta puede consultarse en la página del manual o en la documentación **info**).

cat, **tac**, **nl**, **more**, **less**, **head**, **tail**

Permiten ver uno o más archivo o parte de un archivo. **cat** y **tac** presentan los archivos que reciben uno a continuación de otro, aunque **tac** los presenta invertidos. La orden:

```
cat /etc/profile /etc/hostname
```

presenta el archivo `/etc/profile` y a continuación `/etc/bashrc`. **cat** entre sus opciones acepta `-n` con la cual numera las líneas, aunque esa labor puede hacerse mejor con **nl**. **more** y **less** permiten al usuario "desplazarse" sobre uno o más archivos empleando teclas (e.g barra espaciadora para avanzar una página), **less** permite también buscar y acepta algunos comandos como los de vi. **head** y **tail** muestran parte del comienzo o del final de un archivo, por defecto 10 líneas aunque puede especificarse la cantidad de líneas con la opción `-n` o puede especificarse la cantidad de bytes con la opción `-c`.

split, **csplit**

Dividen un archivo en varias partes. **split** divide empleando como criterio un tamaño, mientras que **csplit** divide de acuerdo al contexto —divide cada vez que en el texto aparezca una cadena [9] o patrón. **split** acepta como opciones para especificar el tamaño `-b` y `-l`, el primero seguido del tamaño en bytes y el segundo seguido del tamaño en líneas. Por ejemplo, para dividir un archivo `grande` en partes de 100000 bytes:

```
split -b 100000 grande
```

las partes quedarán en archivos con nombres como `xaa`, `xab`, etc. Para dividir un archivo `dir.txt` siempre que la palabra `nombre` aparezca:

```
csplit dir.txt /nombre/ {*}
```

9. En estas guías y otros documentos relacionados con informática (búsquedas, procesamiento de texto, programación) una cadena es una secuencia de caracteres —un texto.

En el ejemplo anterior la cadena por buscar [10] se especificó a continuación del nombre del archivo entre los caracteres '/', la cadena final "`{*}`" indica que debe dividirse cada vez que se encuentre una nueva ocurrencia de `nombre`, si no se usa **csplit** sólo buscará la primera ocurrencia y dividirá el archivo en dos partes, si en vez de `*` se emplea un número **csplit** buscará a lo sumo esa cantidad de veces la palabra y dividirá el archivo por cada ocurrencia que encuentre.

sort

Por defecto ordena las líneas de un archivo, con la opción `-m` mezcla dos archivos ordenados y con la opción `-c` verifica que un archivo esté ordenado. Al ordenar puede considerar las líneas completas, considerarlas números enteros (opción `-n`) o flotantes (opción `-g`), considerarlas meses (opción `-M`) o dividirlas en campos y emplear algunos campos como llaves de ordenamiento (opción `-k` para especificar llaves y `-t` para cambiar separador de campos). En las comparaciones puede ignorar diferencias entre mayúsculas y minúsculas con la opción `-f`, puede ordenar de mayor a menor con la opción `-r` —por defecto ordena de menor a mayor— y puede eliminar líneas repetidas con la opción `-u` [11].Por ejemplo para ordenar por líneas el archivo `nombres.txt`:

```
sort nombres.txt
```

si cada línea del archivo `nombres.txt` tiene el apellido y después el nombre de una persona separados por espacios, puede ordenarse por nombre con:

```
sort -k 2 nombres.txt
```

wc

Cuenta cantidad de palabras, líneas y caracteres en uno o más archivos. Por defecto presenta los tres datos por cada archivo que reciba y después presenta las sumas. Con la opción `-c` presenta la cuenta de palabras, con `-l` la cuenta de líneas, con `-w` la cuenta de palabras y con `-L` la longitud de la línea más larga.

3.2.1.3. Redirección y tuberías

En Unix, todo programa tiene 3 canales estándar para intercambiar información con el usuario, con el hardware o con otros programas: salida estándar, entrada estándar y error estándar, que verá abreviados como `stdout`, `stdin` y `stderr` respectivamente. Estos canales pueden conectarse a archivos, a dispositivos o a canales de otros programas (tuberías). Por defecto la entrada estándar es el teclado, mientras que la salida y el error estándar son la pantalla.

Por ejemplo los programas **cat** y **sort** si no reciben un archivo como parámetro esperan información de la entrada estándar y envián el resultado de su operación a la salida estándar; para comprobarlo ejecute el programa **cat** o el programa **sort** sin argumentos.

Para redireccionar la salida estándar a un archivo se agrega al final del comando una cadena de la forma "`>` *`nombre_archivo`*". Por ejemplo para ordenar un archivo `palabras.txt` y dejar el resultado en `ord.txt`:

```
sort palabras.txt > ord.txt
```

10. Más que una cadena se trata de una expresión regular.
11. También puede emplear el programa `uniq` para eliminar líneas repetidas en un archivo ordenado.

Para redireccionar el error estándar a un archivo se agrega tras el nombre del comando `2>` *`nombre_archivo`* [12]. Por ejemplo:

```
cat /ceq11 2> err.txt
```

escribirá en el archivo `err.txt` un mensaje como "`cat: /ceq11: No existe el fichero o el directorio`" —claro está mientras efectivamente no exista el archivo `/ceq11`. Pueden redireccionarse tanto salida estándar como error estándar simultáneamente, por ejemplo:

```
cat /etc/fstab > /dev/null 2> /dev/null
```

redireccionará tanto salida como error estándar al dispositivo `/dev/null` —El dispositivo `/dev/null` es como un agujero negro, todo lo que envié ahí desaparece (ver Ubicación de archivos y directorios).

Por defecto al redireccionar con > a un archivo, se crea un nuevo archivo, en caso de que ya existiera su contenido previo se pierde y es remplazado por la información redireccionada. Por ejemplo después de ejecutar la secuencia:

```
sort 1.txt > ord.txt
sort 2.txt > ord.txt
```

quedará en `ord.txt` únicamente el contenido de `2.txt` ordenado. Para agregar la información redireccionada al final de un archivo ya existente puede emplear » en lugar de >, retomando el ejemplo anterior:

```
sort 1.txt > ord.txt
sort 2.txt » ord.txt
```

dejará en `ord.txt` el contenido de `1.txt` ordenado y a continuación el de `2.txt` ordenado.

Para lograr que la entrada estándar sea leída de un archivo, se agrega al final del comando < *`nombre_archivo`*. Por ejemplo

```
head  < arc.txt
```

presentará las 10 primeras líneas del archivo `arc.txt` (es equivalente a **head arc.txt** o a **head - < arc.txt** [13]). Poder redireccionar la entrada estándar de un programa puede ahorrarle tiempo, porque puede escribir la secuencia de líneas que desea enviar al programa en un archivo y después repetirla tantas veces como desee. Por ejemplo para enviar el mismo correo electrónico a varias personas puede escribirlo en un archivo `correo.txt` y después ejecutar algo como:

```
mail usuario1@micolegio.edu.co usuario2@micolegio.edu.co < correo.txt.
```

Empleando redireccionamiento y los comandos y programas que ya se han presentado se pueden hacer rápidamente muchas labores. Un ejemplo es dividir un archivo grande para transportarlo en varios disquetes: con `split -b 1000000 grande.gz` divide el archivo `grande.gz` en varias partes, cada una de las cuales cabe en un disquete (digamos que queda dividido en tres partes `xaa`, `xab` y `xac`). Puede copiar cada una de las partes en un disquete, llevar los disquetes a otro computador y emplear **cat** y redireccionamiento para unirlos:

12. Otra forma de redireccionar salida estándar es agregando `1>` *`nombre_archivo`*. En general **n**> *`nombre_archivo`* permite redireccionar el descriptor de archivo **n**. Por convención 1 es el descriptor de archivo de salida estándar, 2 el de error estándar y 0 el de entrada estándar. Puede emplear otros números para referenciar nuevos descriptores de archivos.
13. Varios programas emplean como convención tratar al nombre de archivo - como entrada estándar

```
cat xaa xab xac > grande.gz
```

Otra facilidad para redireccionar entrada estándar es «, que después de un comando permite escribir un texto que se constituirá en la entrada estándar. A continuación de « debe ponerse una palabra, escogida por usted, que indicará fin de entrada. La entrada estándar constará de las líneas que se digiten a continuación hasta la primera que contenga sólo la palabra que indicaba fin de entrada. Por ejemplo:

```
sort «FIN
pábulo
opíparo
FIN
```

ordenará las palabras dadas (excepto `FIN` que indica el fin de la entrada). Así, « es equivalente a editar un archivo y después redireccionarlo a la entrada estándar de un programa.

El operador de composición '|' permite redireccionar la salida de un comando en la entrada estándar de otro (es como pegar dos tubos). Por ejemplo:

```
ls -l / | wc -l
```

mostrará la cantidad de archivos y directorios en el directorio `/`. Al ejecutar **ls -l /** se envía a salida estándar la lista de archivos del directorio raíz, cada uno en una línea, con ' | ' esta salida se convierte en entrada de **wc -l**; el programa **wc** con la opción `-l` presenta sólo la cantidad de líneas de la entrada estándar.

El ejemplo anterior pudo haberse realizado en dos pasos empleando un archivo temporal, digamos `temp.txt`:

```
ls -l / > temp.txt
wc -l < temp.txt
```

pero claro está, | es más práctico y aún más cuando deben componerse más de dos programas.

3.2.2. Lecturas recomendadas: bash y el juego de herramientas

- En esta sección se han presentado unas cuantas de las facilidades de `readline` y `history`. Puede consultar más sobre el uso de estas librerías en la página del manual de bash en la sección "READLINE y HISTORY". Hay más información de `readline` (por ejemplo sobre la configuración) en el manual info de `rluserman` (Ver (**info rluserman**)q).
- Puede consultar más información sobre las herramientas para procesar textos introducidas en esta sección y sobre otras que no se mencionaron en la documentación info de textutils (Ver (**info textutils**)).
- Puede consultar otras explicaciones sobre tuberías y el uso del kit de herramientas Unix en "Linux: Instalación y Primeros Pasos. Sección 3.8 Fontanería UNIX". http://lucas.hispalinux.es/Manuales-LuCAS/LIPP/lipp-1.1-html-1.2/lipp3.htm#3.8 y en la sección "*Opening the software toolbox*" de la documentación info de textutils (Ver (**info textutils**)).

3.2.3. Ejercicios: bash y el juego de herramientas

1. Ingrese a una sesión con bash y practique las facilidades interactivas presentadas en esta guía. En particular practique **Control-k** y **Control-y** para cortar y pegar líneas, después de usar **Control-y** emplee **Alt-y** varias veces (si tiene inconvenientes con **Alt-y** intentente **ESC-y**). ¿Para qué se usa **Alt-y** ?

2. ¿Qué hace el comando **cat datos2.txt > datos3.txt** ?

3. ¿Qué hace el comando **ls -Rl ~ | wc -l** ?

4. ¿Qué hace el comando **nl /usr/doc/debian/debian-manifesto | less** ?

5. Dado un archivo `resumen.txt`, ¿cómo puede determinar si el texto tiene menos de 5000 palabras?

6. Dado un archivo `correo.txt`, ¿cómo puede verificar que todas las líneas tengan menos de 75 caracteres?

7. Sin emplear **split** ni editores de texto, ¿cómo puede dividir un texto `pord.txt` de 100 líneas, en dos archivos `uno.txt` y `dos.txt` el primero con las primeras 40 líneas y el segundo con las últimas 60?

8. Dado un archivo con nombres de personas `pers.txt` —un nombre en cada línea—, escriba un comando para `bash` que ordene los nombres y presente sólo los 10 primeros.

9. Dada una lista de palabras en un archivo `palabras.txt` —una en cada línea—, escriba un comando para bash que presente la cantidad de palabras no repetidas.

10. Escriba un comando para **bash** que muestre la cantidad de archivos de configuración en su directorio personal (ver Lectura Archivos y permisos para repasar que son archivos de configuración).

11. Escriba un comando para bash que le envíe a usted un correo electrónico con el listado de los archivos de su directorio personal ordenado alfabéticamente.

3.3. Procesos y tareas

Indicadores de Logros

- Puede iniciar, detener y ver el estado de procesos (`&` en bash, programas **ps**, **pstree**, **top**, **gtop** y **kill**).
- Maneja control de tareas del intérprete de comandos bash (i.e. **%**, **fg**, **bg**).
- Programa tareas en el futuro y maneja comandos relacionados con el tiempo (**at**, **date**, **sleep**, **time**, **crontab**).

3.3.1. Lectura: Procesos y tareas

Un sistema Linux típico puede prestar muchos servicios simultáneamente, puede ser servidor de web, al tiempo que es servidor de correo electrónico, puede atender varios usuarios y cada usuario puede estar realizando simultáneamente diversas acciones. Por esto Linux es llamado un sistema multitarea.

A cada acción en un sistema Linux se le llama proceso. Un proceso abstrae una acción que el sistema debe realizar, independiente del momento en que debe ejecutarse. En esta sección se explica como puede controlar procesos y como puede aprovechar al máximo las capacidades multitarea de Linux, por ejemplo realizando diversas labores simultáneamente, o haciendo que la ejecución de un programa continúe después de que usted cierra su sesión (por ejemplo si se trata de un programa que debe correr durante varias horas o días), o incluso programando el inicio de procesos en momentos en los que usted no tiene una sesión abierta (por ejemplo durante la noche –claro está mientras el computador esté encendido a la hora que programe la tarea).

3.3.1.1. Procesos

Cada proceso tiene asociado un número que lo identifica, un estado que indica como está operando, un grupo que lo asocia con otros procesos, una prioridad que determina su "importancia" con respecto a otros procesos y un dueño que puede controlarlo (normalmente el dueño es el usuario que inicia el proceso). Todos los procesos comparten el procesador —su computador normalmente tendrá un solo procesador—, para lograrlo, cada proceso emplea el procesador durante un intervalo corto de tiempo y después duerme [14] o se bloquea para dar posibilidad a otro proceso de emplearlo (el orden en el que se ejecutan depende de la prioridad de cada proceso). Normalmente junto con cada programa iniciado por el usuario se inicia un proceso [15], que a su vez puede iniciar otros procesos formando así un árbol; puede examinar tal árbol con el programa **pstree**. Existen también procesos que no son iniciados explícitamente por un usuario, por ejemplo procesos iniciados durante el arranque del sistema o por X-Window, tales procesos generalmente pueden ser controlados sólo por el administrador del sistema —quien también podría controlar los procesos de los usuarios.

Cada programa o tubería que inicie desde el intérprete de comando se ejecutará en un nuevo proceso que por defecto estará en primer plano [16] , es decir que bash suspenderá su ejecución y la reanudará cuando el programa que inició termine. Si desea iniciar un programa (o una secuencia de programas unidos por tuberías) en segundo plano [17] , agregué al final del comando un espacio y el carácter '`&`'. Esto es útil cuando debe ejecutar un programa no interactivo que toma bastante tiempo en completarse, porque mientras la ejecución del programa se completa puede continuar trabajando en el intérprete de comandos —el programa que inicie se ejecutará en segundo plano mientras `bash` continua ejecutándose en primer plano. Por ejemplo la conversión de DVI a PostScript (ver Sistemas para preparar documentos) de un documento grande puede tomar bastante tiempo, para realizar la labor en el fondo puede emplear:

```
dvi2ps -o salida.ps entrada.dvi &
```

o aún mejor redireccionando salida estándar a archivos para que no se mezclen con su sesión con bash (error estándar sigue redireccionado a consola así que verá en su sesión con bash los errores que puedan producirse):

```
dvi2ps -o salida.ps entrada.dvi >log  &
```

Cuando inicia un programa (o una tubería) en segundo plano, bash reanuda su ejecución inmediatamente, presenta el número de tarea que asignó al comando y a continuación el número del proceso.

Además de **pstree**, un usuario puede ver sus procesos con el programa **ps** (con la opción `-e`, **ps** muestra todos los procesos del sistema). Junto con cada proceso **ps** presenta: identificación del proceso; la terminal en la que presenta información, en caso de que funcione de forma interactiva (por ejemplo una consola virtual como `tty1` o una terminal de X-Window como `pts/0`); el estado del proceso y el tiempo

14. Dormido en este contexto se refiere a un estado al cual entra un proceso durante cierto intervalo de tiempo para dar oportunidad a otros procesos de emplear el procesador (en **ps** se identifica con la letra `S`). Un proceso está en estado bloqueado (letra `D` en **ps**) si está esperando un recurso que otro proceso está ocupando —por ejemplo un dispositivo. El estado suspendido (letra `T` en **ps**) indica que el usuario solicitó suspender el proceso para reanudarlo después.
15. Puede hacerse que un programa corra en el mismo proceso del intérprete de comandos ejecutándolo con el comando **exec**. Con **exec** también puede redireccionarse descriptores de archivos del intérprete de comandos.
16. Primer plano: del inglés *foreground*
17. Fondo: del inglés *background*

que ha usado el procesador —el resto del tiempo que el proceso haya existido ha estado durmiendo o esperando algún evento o recurso. Para examinar interactivamente los procesos de un sistema pueden emplearse los programas **top** o **gtop**, los cuales además de presentar los procesos y refrescar continuamente sus estadísticas, permiten enviar señales a cada proceso (entre otras diferencias **top** funciona en modo texto mientras que `gtop` es una aplicación Gnome).

3.3.1.2. Señales

En ocasiones usted deseará terminar algún proceso, por ejemplo porque deja de responder o tarda demasiado en completarse; para hacerlo puede emplear el programa **kill** para enviarle una señal de terminación. Una señal es como un "llamado de atención" que se hace a un proceso en situaciones excepcionales (por ejemplo errores), pueden ser producidas por otros procesos, por el usuario o por el sistema operativo y en la mayoría de los casos conducen a la terminación del proceso que recibe la señal. Hay diversos tipos de señales, cada una tiene un número, un nombre que la identifica y una acción predefinida (que generalmente puede ser cambiada por el proceso). Un usuario puede enviar una señal a un proceso con el programa **kill** seguido de la señal que enviará y del proceso que la recibirá:

```
kill -SIGTERM 945
```

Este ejemplo envía la señal `SIGTERM` al proceso con identificación 945 (en vez de `SIGTERM` pudo haberse usado 15 que es el número que corresponde a esa señal). Puede consultar un listado de todas las señales y sus números con **kill -l**.

A continuación se presenta una breve descripción de algunas señales comúnmente empleadas por usuarios:

`15 SIGTERM`

Esta señal solicita la terminación del proceso que la recibe.

`9 SIGKILL`

Esta señal termina el proceso que la recibe de forma inmediata. Empleela sólo para detener procesos que no terminan con la señal `SIGTERM`.

`2 SIGINT`

Es la misma señal que se produce cuando un usuario en un programa interactivo presiona, **Control-C** para solicitar su terminación.

`3 SIGQUIT`

La misma señal producida por **Control-**, su efecto es análogo al de `SIGINT` pero además actúa como si el programa hubiera provocado algún error interno (volcando el contenido de memoria a un archivo `core`).

`20 SIGTSTP`

La misma señal producida por **Control-z**, su efecto es suspender la ejecución de un proceso —para reanudarla después.

`18 SIGCONT`

Reanuda un proceso suspendido previamente por la señal `SIGTSTP`.

`1 SIGHUP`

Esta señal es enviada por bash a todas las tareas que se ejecutan en segundo plano, cuando el usuario cierra la sesión (por ejemplo al cerrar una terminal en X-Window o cuando sale de su sesión desde una consola virtual). Normalmente un proceso terminará cuando reciba esta señal, pero puede lograrse que el proceso continué (es decir que ignore la señal `SIGHUP`) si el comando se inició con `nohup` —que evita que el programa reciba la señal `SIGHUP`) o si durante su ejecución se indicó a `bash` no enviarle esta señal cuando se cierre la sesión, empleando el comando `disown`. Esto es muy útil cuando debe dejar corriendo un proceso muy demorado (horas o días) mientras usted no tiene una sesión abierta, por ejemplo para ejecutar el programa **make** [18] en segundo plano, redireccionado salida estándar al archivo `sm`, error estándar a `esm` y lograr que continue después de que se cierre la sesión:

```
nohup make > sm 2> esm &
```

3.3.1.3. Control de tareas

Además de las facilidades para controlar procesos que se han presentado en esta sección, bash ofrece "control de tareas". Una tubería o un programa que se ejecute desde bash tiene asociado un número de tarea, diferente al número del proceso. El número de la tarea aparece entre paréntesis cuadrados después de ejecutar un programa en segundo plano. La lista de las tareas de una sesión de bash, puede verse con el comando `jobs`, por ejemplo allí vera los programas que inició en segundo plano o que haya suspendido con la tecla **Control-z**. Con el comando `fg` puede poner en primer plano una tarea que esté en segundo plano, es decir reanudar la aplicación permitiendo que controle la consola mientras que bash se suspende. Por ejemplo si está editando un correo electrónico con **mail**, suspende la edición con **Control-z** y después desde bash emplea **jobs** verá algo como:

```
[1]+ Stopped           mail amigo@micolegio.edu.co
```

que indica que el programa **mail** esta suspendido y su número de tarea es 1. Para bash los números precedidos del caracter '`%`' indican tareas, así que para reanudar su ejecución en primer plano puede usar:

```
fg %1
```

claro que **fg** le permite emplear el nombre del programa o sus primeras letras en lugar de `%1`, omitir el símbolo `%`, o incluso si emplea el símbolo `%` puede omitir `fg`. Así que los siguientes comandos son equivalentes al ejemplo anterior:

```
fg ma
fg 1
%1
```

18. **make** se emplea para compilar rogramas o documentos (como estás guías). En ocasiones esta labor puede tomar mucho tiempo.

Otro comando que le permite controlar tareas desde bash es **bg** el cual le permite ejecutar en segundo plano un programa que está suspendido. Por ejemplo si inicia una impresión de un documento postscript que imprime 2 páginas del original en una con:

```
a2ps --columns=2 documento.ps
```

puede suspenderla con **Control-z**, y continuarla en segundo plano con **bg -** (- es una convención que indica la tarea más reciente) o suponiendo que el número de tarea es 1, con **bg %1** o con `%1 &`.

Note que cuando suspende un programa con **Control-z**, la ejecución se *suspende*, si desea continuarla en segundo plano debe reanudarla en segundo plano con **bg**.

3.3.1.4. Tiempo

Usted también puede programar cuando ejecutar un proceso con el programa **at** o eventualmente puede programar eventos periódicos con **cron**. Antes de introducirlos, describimos algunos programas relacionados con tiempo:

date

Programa para ver o poner la fecha y hora del sistema (aunque sólo puede ser cambiada por el administrador). Por defecto presenta la hora local de acuerdo a la zona geográfica donde esté el computador que está usando, con la opción `-u` presenta la hora en el meridiano 0 (i.e tiempo universal coordinado). El administrador puede establecer la fecha y la hora con la opción `-s` seguida de la fecha y/o hora entre comillas. La información que el comando **date** presenta puede ser desplegada con un formato diferente con las opciones `-I` y `-R`.

time

Es un comando interno de bash, que permite medir el tiempo que emplea la ejecución de un programa. Por ejemplo:

```
time cat /etc/hosts
```

ejecuta el programa **cat** con argumento `/etc/hosts` y después presenta el tiempo real, de usuario y del sistema que requirió la operación. La diferencia en estos tiempos se debe a las diversas tareas que Linux realiza. Tiempo real se refiere a el tiempo que transcurre desde que se inicia el programa hasta que este termina (sumando tiempos de otros procesos), el tiempo que emplea sólo el proceso es la suma del tiempo de usuario y tiempo del sistema (el primero indica tiempo realizando operaciones fuera del kernel y el segundo tiempo dentro del kernel).

sleep

Este programa duerme el proceso en el que se ejecuta durante un tiempo, por defecto especificado en segundos. Pueden emplearse los posfijos `s`, `m`, `h` y `d` para indicar segundos, minutos, horas o días. Por ejemplo para dormir un proceso durante 10 segundos **sleep 10**.

bash puede ejecutar diversos programas y comandos uno después de otro cuando se separan con punto y coma ";", puede aprovechar esto y el programa **sleep**, para ejecutar tareas después de cierto intervalo de tiempo. Por ejemplo para iniciar la conversión de un documento de PostScript a PDF (ver Sistemas para preparar documentos) 30 minutos después de dar el comando:

```
sleep 30m ; pdf2ps carta.pdf carta.ps
```

Sin embargo una mejor forma de iniciar tareas en el futuro es con el comando **at**. Este comando recibe la hora a la que debe ejecutarse el o los programas que se le den por entrada estándar (puede especificar un archivo con comandos con la opción `-f`). Por ejemplo:

```
at 8:40PM today « EOF
pdf2ps carta.pdf carta.ps
EOF
```

convertirá el documento `carta.pdf` a formato postscript a las 8:40PM del mismo día. La fecha puede especificarse de muchas maneras, por ejemplo `20:40 20.01.2005` o `now+2 hours`. Con el comando **atq** puede ver las tareas programadas y con **atrm** puede eliminar una tarea especificando el número (el número de cada tarea programada es mostrado por **atq**). Cuando programa una tarea con **at**, esta se ejecutará aún cuando usted no tenga una sesión abierta y mientras el sistema esté operando a la hora programada.

Para que un usuario pueda programar eventos periódicos con **cron**, el administrador del sistema debe otorgarle permiso. Si usted tiene el permiso podrá emplear el programa **crontab** para agregar acciones que se ejecutaran periodicamente. Si teclea sólo **crontab -e** entrará al editor que tenga configurado (variable EDITOR ver Ambiente y variables de ambiente) para modificar su archivo de acciones periódicas (`/var/spool/cron/crontabs/`*`usuario`*), un ejemplo de tal archivo es:

```
# Mi archivo de acciones periodicas para cron
PATH=/usr/bin:/usr/local/bin
0 0 24 12 * $HOME/cron/recuerda1.sh   # Cumpleaños
0 * * * * $HOME/cron/hora.sh
```

que establece la variable de ambiente PATH que se usará al realizar las acciones y especifica dos comandos por ejecutar periodicamente. El script `$HOME/cron/recuerda1.sh` se ejecutará el 24 de diciembre de cada año a las 00:00 (medianoche), mientras que `$HOME/cron/hora.sh` se ejecutara cada hora (a las 0:00, 1:00, 2:00, ... 23:00), todos los días, todos los meses, todos los años. Un archivo para **crontab** puede tener líneas con comentarios (iniciadas con el caracter '#'), líneas que definen variables de ambiente y líneas que especifican acciones periodicas compuestas por 6 campos y eventualmente seguidas por un comentario.

Los campos de estas acciones se separan unos de otros con uno o más espacios y son en orden: minuto, hora, dia del mes, mes, dia de la semana y comando por ejecutar. **cron** es un proceso que cada minuto examina los archivos **crontab** de los usuarios y ejecuta los comandos cuyo tiempo concuerde con la hora del sistema. El tiempo de un comando concuerda con la hora del sistema si la hora, los minutos, el mes y bien el día del mes o bien el día de la semana [19] concuerdan. La hora del sistema concuerda con la hora de una acción si ambas son iguales o sni la hora de la acción es el caracter '*', lo analogo ocurre con los minutos, meses y días.

Puede examinar su archivo de acciones periódicas bien editandolo o con **crontab -l**, para borrarlo puede usar **crontab -r**, puede remplazar sus acciones periodicas con las de un archivo (digamos `microntab`) con:

```
crontab microntab
```

19. En un archivo `crontab` los días de la semana se especifican con números de 0 a 6, 0 es domingo, 1 lunes y así sucesivamente.

3.3.2. Lecturas recomendadas: Procesos y tareas

- Los programas **pstree**, **ps**, **top** y **gtop** tiene muchas opciones que permiten configurar qué información presentar y la forma de presentarla, puede consultarlas en el manual —la documentación de **gtop** puede consultarse con el sistema de ayuda de Gnome (ver Lectura Búsqueda y consulta de documentación).
- Puede consultar más sobre señales en la página del manual sobre **kill**, también puede consultar la explicación detallada sobre cada señal que se presenta en el manual info de `libc` (Ver (**info libc**)).
- Podrá encontrar información completa sobre control de tareas en la página del manual correspondiente a bash (sección Task Control).
- Otra explicación en español, sobre procesos y control de tareas está disponible en la sección 3.11 del libro de Matt Welsh "*Linux: Instalación y Primeros Pasos*", traducido a español por los miembros del proyecto Lucas: http://lucas.hispalinux.es/Manuales-LuCAS/LIPP/lipp-1.1-html-2/lipp3.htm#3.11
- La descripción del comando **time** que se presentó en esta sección corresponde a un comando interno de bash, puede consultar más sobre la misma en la página del manual de bash. Su sistema tiene también un programa `/usr/bin/time` que se emplea de forma análoga pero que puede presentar más información de los recursos empleados durante la ejecución de un programa. See (time), para más información.
- Puede consultar más sobre **at** en la página del manual. La sintaxis precisa del formato de fechas y horas que **at** acepta, puede consultarla en `/usr/doc/at/timespec`.
- El programa **batch** permite ejecutar un proceso cuando la carga en el sistema es baja. El programa **nice** inicia un proceso con una prioridad especificada por el usuario, la prioridad de un proceso que se está ejecutando puede cambiarse con el programa **renice**. La información completa sobre estos programas puede verla en las páginas del manual.
- El comando **trap** permite asociar una secuencia de comandos a una señal, es decir que ciertos programas se ejecuten cuando bash recibe cierta señal. Por ejemplo **trap "rm *.err" SIGTERM** hará que el comando **rm *.tmp** se ejecute cuando bash reciba la señal `SIGTERM`. Puede consultar más sobre trap en la página del manual de bash.

3.3.3. Ejercicios: Procesos y tareas

1. Revise los procesos que están corriendo en su sistema con los programas **ps**, **pstree**, **top** y **gtop**.

2. ¿Qué hace el siguiente comando? **dvi2ps -o salida.ps entrada.dvi >log 2>err &**

3. Inicie un proceso que no termine en el fondo (por ejemplo **cat < /dev/zero > /dev/null** o `yes`), revise su estado empleando **ps**, después suspenda el proceso enviando la señal apropiada con **kill**, revise nuevamente el estado y compruebe que es `T`; reanude la ejecución enviando la señal `SIGCONT`;

compruebe que el estado sea nuevamente en ejecución y finalmente termine el proceso enviando la señal `SIGTERM`.

4. Inicie un programa interactivo (que requiera interacción con el usuario, por ejemplo vi), desde bash y después de iniciado suspéndalo. Revise entonces la lista de procesos y compruebe que el nuevo proceso esté y que su estado sea suspendido. Después reinicielo y finalmente eliminelo.

5. Mida el tiempo que el siguiente programa tarda en ejecutarse: **ls -R /usr/doc**

6. Usando el comando **at** haga recordatorios de la fecha de cumpleaños de un familiar, de forma tal que el día anterior le envíe un correo recordando.

7. Algunos programas requieren mucho tiempo para ejecutarse (horas o días). Por ejemplo el siguiente programa para bc(42) que presenta los factores de un número, puede demorarse mucho tiempo para factorizar un número grande.

```
define factores(x) {
     f=2;
     maxf=x/f;
     while (f<=maxf) {
       if (x%f==0)
         { x=x/f; maxf=x/f; print f,"\n"; } else { f=f+1; }
       if (f%100000==0) { print "*"; }
    }
    return (x);
  }
```

Puede entrar a bc y teclear el programa presentado, después probarlo por ejemplo con `factores(10)` –que indica que se debe factorizar el número 10– el cual presentará como respuesta 2 y 5. El objetivo de este ejercicio es factorizar [20] el número 129098564527119574834 empleando bc, el programa presentado y lo que ha aprendido sobre procesos en esta guía, opcionalmente también se quiere saber el tiempo que toma la factorización. Ayudas: Emplee redireccionamiento y nohup. Para salir de bc emplee `quit`

3.4. Scripts básicos para bash

Indicadores de Logros

- Crea scripts sencillos para bash.
- Emplea variables de ambiente y reconoce algunas empleadas por bash.
- Aprovecha las expansiones que bash realiza.
- Emplea comandos de bash y programas comúnmente usados al hacer scripts.

20. El problema de factorizar un número es muy importante porque de su dificultad dependen muchos sistemas de criptografía, incluso hay premios en dinero para quienes logren factorizar ciertos números http://www.rsasecurity.com/rsalabs/challenges/factoring/numbers.html

3.4.1. Lectura: Scripts para bash

Un script para bash es un archivo tipo texto, cuyas líneas tienen comandos que son ejecutados (interpretados) por bash. Para lograr que el intérprete de comandos intérprete las líneas de un archivo puede:

- Ejecutar `/bin/bash` seguido del nombre del archivo (o redireccionar la entrada estándar para que provenga del archivo).
- Emplear el comando **source** seguido del nombre del archivo.
- Emplear el caracter '.' seguido de un espacio y el nombre del archivo.
- Agregar en la primera línea del archivo la cadena `#!/bin/bash`, dar permiso de ejecución al archivo y teclear el nombre del archivo desde el intérprete de comandos —como si fuera un nuevo comando.

En este capítulo presentamos algunas facilidades que bash brinda y que resultan muy útiles para escribir scripts. Antes se presentará como ejecuta bash una orden (lo cual en particulars explica porque la primera línea de un script para bash debería ser `#!/bin/bash`).

Dado que las facilidades que presentaremos no son exclusivas de scripts sino de bash, puede hacer experimentos desde el intérprete de comandos mientras lee.

3.4.1.1. Ejecución de un comando en bash

Para determinar cual es la orden bash realiza varias acciones:

1. Primero realiza ciertas expansiones a la línea de comando, e identifica la orden, los parámetros y eventualmente las variables de ambiente que se den junto con la orden (en esta sección se estudiarán las expansiones y variables de ambiente).
2. Si la orden va precedida de una ruta —se trata del nombre completo de un archivo— y el archivo existe y es ejecutable, bash lo trata como un programa y lo carga a memoria para ejecutarlo. Si la ruta no conduce a un archivo ejecutable bash presenta un mensaje de error.
3. Si la orden no va precedida de una ruta, busca un alias que pueda corresponder con la orden y si lo encuentra lo remplaza por su valor (más adelante en esta sección explicaremos como manejar alias).
4. Si la orden no esta precedida de una ruta y no es un alias, determina si se trata de un comando interno de bash (como **fg** o `bg`) y en caso de serlo realiza la acción correspondiente.
5. Si la orden no está precedida de una ruta, no es un alias y no es un comando interno busca en varios directorios un archivo ejecutable con el nombre dado y si lo encuentra en alguno lo carga a memoria para ejecutarlo (el orden y los directorios donde busca se especifican en la variable de ambiente PATH que se explicará en esta sección).
6. En otro caso bash presenta un mensaje de error.

En caso de que la orden corresponda a un archivo ejecutable (bien porque se dio la ruta completa o bien porque existe un archivo en un directorio de PATH), bash determinará como ejecutarlo:

1. Si el archivo es tipo texto y comienza con la cadena "`#!`" seguida del nombre de un programa, bash emplea tal programa como intérprete del archivo. Como parámetros para el intérprete emplea los

que estén en la primera línea del archivo, seguidos del nombre del archivo y a continuación otros parámetros que el usuario hubiera dado desde la línea de comandos.

2. Si el archivo es tipo texto pero no comienza con `#!`, lo interpreta con el programa bash [21]. Es decir lo trata como un script para el intérprete de comandos.
3. Si el archivo es binario lo envía al kernel para su ejecución. El kernel podrá ejecutarlo si está en un formato reconocido (e.g el ejecutable producido por un compilador en un formato reconocido por Linux —ELF, a.out).

3.4.1.2. Ambiente y variables de ambiente

Una variable es un nombre al cual se le puede asociar un valor, tal valor puede cambiar durante la ejecución de un programa —es 'variable'. Cada programa (incluyendo al intérprete de comandos) se inicia en un ambiente el cual consta de variables —variables de ambiente— que hereda del programa que lo inició y que pueden tener un significado especial para el programa. Las variables de ambiente que un programa usa se especifican en la página del manual del programa.

Para asignar un valor a una variable de ambiente y crearla si no existe, teclee el nombre de la variable, en seguida el carácter '=' y después el valor. El valor puede constar de letras, símbolos o números, pero tenga en cuenta que hay algunos caracteres con significado especial y que es mejor evitar en sus primeros experimentos: $, {, }, ', ' (podrá escribirlos precedidos de \ o encerrando el valor a asignar entre apostrofes). Por ejemplo para asignar el valor `/home` a la variable DIR:

```
DIR=/home
```

Después de asignar una variable, puede emplear el valor asociado a la misma en comandos que de al intérprete de comandos, para hacerlo emplee el nombre de la variable precedido del carácter '$'. Por ejemplo **ls -l $DIR** listará los archivos de la ruta asociada a la variable `DIR`.

En bash puede emplear el comando **echo** para enviar a salida estándar una cadena (por ejemplo **echo tarea > porhacer.txt** dejar en el archivo `porhacer.txt` una línea con la palabra `tarea`), esto puede usarse para examinar el valor de una variable de ambiente, e.g. **echo $DIR** presentará el valor de la variable DIR.

Puede examinar las variables de ambiente de bash con el comando **set**. Algunas de las variables que verá son: USER y USERNAME cuyo valor es el login del usuario; UID con el número que identifica al usuario; TERM mantiene el nombre de la terminal que está usando (ver Lectura Configuración de una sesión); SHELL la ruta y nombre del intérprete de comandos; PWD el nombre del directorio de trabajo; $HOME el nombre del directorio personal del usuario; PS1 y PS2 indican a bash como presentar prompts (see Lectura Configuración de una sesión); PATH es la ruta de directorios donde bash busca archivos ejecutables, se separan unas rutas de otras con el caracter ':'; OSTYPE el tipo de sistema operativo; MAILCHECK la frecuencia en segundos con la que bash debe revisar si ha llegado un nuevo correo a la cola de correos especificada en la variable MAIL (por defecto la del usuario); LS_COLORS colores que emplea el programa ls; LINES y COLUMNS indican la cantidad de filas y columnas de la terminal que está usando; LANG, LANGUAGE y otras variables que comienzan con el prefijo LC_ especifican el idioma en el que los programas deben interactuar con el usuario (ver Lectura Configuración de una sesión); HOSTNAME es el nombre del sistema; HISTFILE mantiene el nombre

21. En rigor lo interpreta con `/bin/sh` pero en Debian `/bin/sh` es un enlace a `/bin/bash`

del archivo con la historia de comandos, su tamaño lo limitan HISTFILESIZE y HISTSIZE; DISPLAY mantiene la dirección del servidor X-Window (see Lectura Servicios de la Intranet).

Cuando se inicia un programa desde bash, el ambiente que tendrá constará de las variables que estén marcadas como exportables y de otras variables que se especifiquen al comienzo del comando (pueden separarse unas de otras con espacios y si el valor de alguna variable debe tener espacios puede encerrar el valor completo entre comillas), por ejemplo para iniciar el programa **man** en un ambiente con las variable LANGUAGE y LANG en el valor DE_de:

```
LANG=de_DE LANGUAGE=de_DE man man
```

Para exportar una variable y lograr así que forme parte del ambiente de procesos creados por su sesión, puede emplear bien **declare -x** *`VAR`* o `export` *`VAR`*, empleando el nombre de la variable que desea exportar en lugar de *VAR*. Empleando sólo **export** o sólo **declare -x** puede ver los nombres y valores de variables exportables.

En un script puede emplear ciertas variables especiales ($1, $2, ...) para referenciar los parámetros que el usuario empleó al iniciarlo. $1 tendrá el valor del primer paramétro, $2 del segundo y así sucesivamente. El siguiente script ejemplifica su uso:

```
#!/bin/bash

echo "Creando $2.tar.gz de $1"

mkdir $2
cp -rf $1/* $2
tar cvf $2.tar
gzip $2.tar
```

Este script recibe dos parámetros, el primero es una ruta y el segundo el nombre de un archivo. Si el nombre del script es `comp` y tiene permiso de ejecución podría usarse para crear un paquete comprimido `d.tar.gz` con el contenido del directorio `~/mand` con:

```
./comp ~/mand d
```

Note que en el ejemplo indicamos la ruta completa del archivo `comp`, suponiendo que es ejecutado desde el mismo directorio donde se encuentra. Si la ruta donde está el archivo está en la variable PATH, no es necesario especificar la ruta.

3.4.1.3. Expansiones

bash trata algunos caracteres de forma especial: `' " { } $`. Al asignar una variable o iniciar un comando bash "expande" estos caracteres y su contexto de varias formas :

`$var` - **Expansión de variables o parámetros**

El caracter '`$`' es empleado para distinguir variables o bien parámetros de un script. Las variables son remplazadas por su valor, por ejemplo **echo $PATH** presentará el contenido de la variable PATH. En un script los parámetros se referencian con números, `$1` es el primero, `$2` el segundo y así sucesivamente. Otros nombre especiales en un script son:

`$#`

Es remplazado por la cantidad de parámetros que el script recibe.

`$*`

Que se expande a todos los parámetros que el script haya recibido, un parámetro se separa de otro con el valor de la variable IFS que normalmente es un espacio.

`$?`

Todo programa al terminar debe retornar un número al sistema operativo, por convención 0 significa operacióne exitosa y números diferente representan errores. `$?` se expande al número retornado por el último programa ejecutado en primer plano. Un script puede retornar un 3 en lugar de 0 con `exit 3`

`$-`

Opciones que se pasaron al script durante su ejecución.

`$$`

Identificación del proceso del intérprete de comandos.

`$!`

Identificación del proceso del último comando que se ejecutó en segundo plano.

`$0`

Nombre del script o del shell.

`"`*`texto`*`"` - **citas**

Cuando un conjunto de caracteres (incluyendo espacios), se encierra entre comillas, bash los trata como una sola cadena. Esto es útil por ejemplo cuando el nombre de un directorio o archivo tiene espacios, e.g. **cd "los amigos"**. Otra forma de representar el caracter espacio es con el caracter '`\`' seguido de un espacio, así el efecto del ejemplo anterior también podría lograrse con **cd los\ amigos**. Hay otros caractéres que pueden representarse con ayuda de '`\`', por ejemplo:

`\n`

Representa el caracter fin de línea.

`\b`

Caracter para borrar a la izquierda.

`\\ \{ \} \$`

Representan los caracteres '\' '{' '}' y '$' respectivamente. Esto es útil para producir estos caracteres sin que bash trate de interpretarlos –estos son caracteres para hacer expansiones.

`\a`

Caracter para emitir un sonido.

`\t`

Caracter tabulador (como la tecla **Tab**).

`{letras}` - **expansión de corchetes**

Una cadena que contenga `{ letra1, letra2, ... }` será expandida a varias cadenas similares a la inicial pero la posición de `{letras1, letra2, ...}` será remplazada por cada una de las letras. Por ejemplo `ls /home/pepe/sal{a,e}n` se expandirá a `ls /home/pepe/salan /home/pepe/salen`. Esta expansión es la primera que se realiza cuando hay varias en un mismo comando, y sólo surge efecto si está fuera de comillas o apóstrofes.

`` `comando` `` o `$(comando)` – **sustitución de comandos**

Un comando encerrado entre apóstrofes invertidos (i.e ‘*comando*‘) o entre las cadenas "`$(`" y "`)`", será expandido al resultado que tal comando envíe a salida estándar cuando es ejecutado. Por ejemplo

```
TEXTOS=`ls *.txt`
```

asignará a la variable TEXTOS los nombres de los documentos tipo texto (i.e el resultado de **ls *.txt**).

```
ls $(cat rutas.txt)
```

presentará los archivos de los directorios que estén en el archivo `rutas.txt`.

`'texto'` - **citas**

Un texto que se encierra entre apóstrofes no es expandido. Esto es útil cuando se requiere una cadena que tiene algunos caracteres reservados para expansiones. Por ejemplo

```
N=10
echo '$N' es $N
```

enviará a salida estándar `$N es 10`

`$((expresión))` - **expansión aritmética**

Una expresión aritmética [22] será evaluada cuando se encierre entre `$((` y `))`, por ejemplo:

```
echo "1+2 es $((1+2))"
```

22. En la expresión pueden emplearse los operadores - (unario y binario), + (unario y binario), ** para elevar a una potencia, * para multiplicar, / para divir, % para obtener residuo de una división. Hay operadores que operan a nivel de bits (representación binaria de los números que operan): ~ para negar bits, & para hacer "Y" entre bits de operandos y | para efectuar "O". También pueden emplearse valores booleanos — 0 se interpreta como falso y 1 como verdadero—, pueden compararse números con los operadores >, <, <= (menor o igual), >= (mayor o igual), == (igualdad), != (diferentes) y pueden operarse booleanos con ! para negar, && como conjunción (Y) y || como disyunción. Los números pueden escribirse en decimal, o en otras bases empleando como prefijo del número la base seguida del caracter '`#`' (también puede escribirse números en octal iniciándolos con 0 o en hexadecimal iniciándolos con 0x).

`*, ?, ~ [letras] [rango(s)]`- **Expansión de rutas**

Algunos caracteres y secuencias son expandidos a nombres de archivos. A continuación se presentan con ejemplos:

- **echo ***, presentará todos los nombres de archivos del directorio de trabajo —porque * expande a todos estos y `echo` los presenta—. En caso de que no haya archivos el caracter `*` no será expandido y será mostrado por `echo`.
- **echo datos?.gnumeric** se expandirá a todos los nombres de archivos que comiencen por `datos` seguidos de un caracter arbitrario a su vez seguido de la cadena `.gnumeric`.
- `echo ~/.*errors` presentará todos los nombres de archivos de configuración que estén en el directorio del usuario y que terminen con errors.
- **rm *[cho]** eliminará todos los archivos que terminen con una de las letras `o`, `h` o `c`.
- **echo [0-9][a-z]*** presentará todos los archivos que comiencen con un dígito seguido de una letra.

3.4.1.4. Comandos y programas útiles al hacer scripts

En un script puede emplear cualquier programa o comando, junto con redireccionamiento, procesos y control de tareas. bash ignora comandos que comienzan con el caracter '#'. Esto es útil para agregar comentarios explicativos a los scripts.

A continuación introducimos algunos comandos de bash y programas útiles al hacer scripts.

read

Lee una línea de entrada estándar y asigna las palabras a las variables que sigan al comando **read**. Puede especificarse un mensaje que se presentará como prompt antes de empezar a leer con la opción `-p` *`mensaje`*. El siguiente ejemplo lee dos palabras en las variables NOMBRE y APELLIDO:

```
read -p "Teclee nombre y apellido: " NOMBRE APELLIDO
```

Si la línea leída tiene más palabras que la cantidad de variables, el resto serán ignoradas. Si tiene menos palabras, las primeras variables (de izquierda a derecha) serán empleadas y el resto quedarán en blanco (i.e con la cadena vacía ""). En caso de que no se den nombres de variables, la línea leída quedará en la variable REPLY.

alias

Con este comando puede definir alias para comandos, después de definir un alias bash lo remplazará por el comando completo. Por ejemplo un alias para abreviar **ls -l** puede ser **ll**, que se definiría como:

```
alias ll="ls -l"
```

Puede listar los alias definidos con la opción `-p` del comando **alias**.

unalias

Permite eliminar alias creados con el comando **alias**, por ejemplo para eliminar el alias **ll** se emplea:

```
unalias ll
```

Si se emplea la opción `-a`, el comando **unalias** elimina todos los alias definidos.

dirname

Recibe como primer parámetro el nombre completo de una archivo, incluyendo su ruta y envía a salida estándar sólo la ruta. Por ejemplo:

```
dirname /usr/doc/xterm/README.Debian
```

presenta en salida estándar `/usr/doc/xterm`.

basename

Análogo a **dirname**, pero envía a salida estándar el nombre del archivo.

hostname

Envía a salida estándar el nombre del computador en la red, esto también puede verse en la variable de ambiente HOSTNAME. Con la opción `-f`, el programa **hostname** presenta el nombre completo, con la opción `-i` la dirección IP, con la opción `-d` el dominio DNS, con la opción `-a` presenta nombres alternos de su máquina y con la opción `-y` el dominio NIS —este programa extrae parte de la información del archivo `/etc/hosts` y de `/etc/hostname`.

whoami

Retorna el nombre del usuario que lo ejecuta. También puede verse en la variable de ambiente USER.

id

Retorna nombre y número del usuario que lo ejecuta, así como nombre y número del grupo o grupos a los que pertenezca. Con la opción -g sólo retorna el número del grupo principal, con -u retorna sólo el número del grupo principal, con `-G` retorna los números de todos los grupos a los que pertenece. Estas opciones pueden seguirse de `n` para presentar nombres en lugar de números.

declare

Las variables de ambiente tienen atributos que pueden examinarse o cambiarse con este comando interno de bash. Los atributos pueden ser sólo lectura (opción `-r`), exportable (opción `-x`) y variable entera (opción `-i`). Una variable de sólo lectura no puede ser modificada, y una variable entera sólo puede asociarse con números. [23]. Para examinar las variables que tengan un cierto atributo se emplea **declare** seguido del código de la opción, por ejemplo para examinar las variables con atributo de sólo lectura **declare -r**. Para cambiar un atributo a una variable se emplea la opción (precedida de - o + para activar y desactivar) seguida del nombre de la variable, por ejemplo **declare -i N** pone el atributo de variable entera a N.

23. Otros atributos pueden ser "función" `-f` y arreglo `-a`

3.4.2. Lecturas recomendadas: Scripts básicos para bash

- Recomendamos de forma especial consultar la página del manual de bash. Hay algunas partes que complementan lo que se ha presentado en esta guía:
  - bash puede hacer más expansiones de parámetros, con la sintaxis `${...}` —por ejemplo sustituciones en el valor de una variable—. Puede consultar la información completa sobre este tema en la sección "expansión de parámetros".
  - Pueden hacerse scripts para bash con otras características que no se han presentado en esta guía: **if**, **case**, **while**, **for**, **function**.

- El intérprete de comandos es una parte importante de la filosofía Unix, el libro clásico sobre este tema es de Brian. W. Kernighan y Rob Pike "El Entorno de Programación UNIX", Prentice-Hall Hispanoamericana. Este libro explica y muestra el uso de csh, un predecesor de bash —la mayoría de características de csh están disponibles en bash.
- En Internet puede consultar "*Advanced Bash-Scripting Guide: A complete guide to shell scripting, using Bash*" de Mendel Cooper, disponible en: http://linuxdoc.org/LDP/abs/html/index.html o Programación en BASH - COMO de introducción http://lucas.hispalinux.es/COMO-INSFLUG/COMOs/Bash-Prog-Intro-COMO/Bash-Prog-Intro-COMO.html

3.4.3. Ejercicios: Scripts básicos para bash

1. Cree un script que sea interpretado por bash, que al ejecutarse busque entre sus directorios, todos los archivos, ordene los nombres alfabéticamente y le envíe un correo con esa información. *Ayuda:* Puede emplear la opción `-R` de **ls** para listar subdirectorios, y la opción `-u` de **sort**.

2. La variable de ambiente HOME contiene el nombre de su directorio personal. Comprúebelo y después empleela para cambiarse a su directorio. Después cambie esta variable y explique que ocurre con el comodín '~'.

3. El programa **man** presenta páginas del manual, empleando el programa especificado en la variable de ambiente PAGER o en su defecto con el programa **less** (ver Lectura bash y el juego de herramientas). Modifique y exporte la variable PAGER para que **man** emplee el programa **more** para presentar información y pruebe el cambio.

4. ¿Qué hace el comando **echo ‘ls‘** ? (note que se usan apóstrofes invertidos).

5. ¿Qué hace el comando **`N=6 echo "'1+$N' = $((1+$N))"`** ? Después de dar su respuesta, compruébela empleando un intérprete de comandos.

6. Haga un script que cada 90 minutos presente el mensaje "Hacer tareas". *Ayuda:* **source**.

7. Cree un script que al ser ejecutado, cree un archivo `script2`. El archivo `script2` creado debe poner la variable PATH en el valor que tenga cuando el primer script sea ejecutado, añadir a tal variable la ruta `/opt/bin` y debe crear un alías que cuando se ejecute muestre el contenido de la variable PATH (no olvide cambiar el modo del archivo `script2` generado para que sea ejecutable).

8. Haga un script que después de ejecutarse pida al usuario 2 números y después presente la suma, la resta, el producto y la división.

3.5. Configuración de una sesión

Indicadores de Logros

- Personaliza sesiones con bash.
- Personaliza sesiones con X-Window.
- Personaliza sesiones con Gnome.

3.5.1. Lectura: Configuración de una sesión

En Linux puede iniciar una sesión desde una consola virtual o desde X-Window. Lo que ocurre en cada caso es diferente, y para lograr configurar la sesión, es muy útil saber algunos de los eventos que ocurren, así como los archivos de configuración que se emplean y su formato –para editarlos de ser necesario.

3.5.1.1. Una sesión con bash

Si pasa a una consola virtual (ver Lectura Sistema de usuarios y manejo de clave) o inicia una sesión de **telnet**, verá un mensaje que presenta el sistema mientras espera que un usuario ingrese, algo como:

```
Debian Linux 2.2 oxigeno tty1
oxigeno login:
```

Este mensaje es generado por el programa `getty`. Después de que usted ingresa su login, se inicia el programa **login** el cual pide su clave y después si la clave es correcta inicializa algunas variables de ambiente (e.g. HOME y SHELL), muestra algunos mensajes (en particular muestra el mensaje del día —archivo `/etc/motd`) e inicia una sesión de login interactiva con bash, o con el intérprete de comandos que haya configurado el usuario (puede configurarse con el programa **chsh**). Usted puede desactivar los mensajes presentado por **login** creando un archivo vacío `~/.hushlogin`.

Al iniciar bash, ejecutará el script `/etc/profile` y después intentará ejecutar uno de los archivos `~/.bash_profile`, `~/.bash_login` o `~/.profile` (en ese orden y sólo se ejecutará el primero que encuentre). El script `~/.bash_profile` también puede contener una llamada al script `~/.bashrc` el cual es el script que bash lee cuando se usa en modo interactivo pero no como intérprete de login [24], por ejemplo con:

```
if [ -f ~/.bashrc ]; then
    source ~/.bashrc
fi
```

Sugerimos que configure su archivo `~/.bash_profile` para ejecutar `~/.bashrc` y que configure variables de ambiente y alias en `~/.bashrc`. Por otra parte, cuando cierra una sesión interactiva con bash se ejecuta el script `~/.bash_logout`.

Dado que bash emplea la librería `readline` (ver Ayudas interactivas de bash), puede configurarse la interacción. `readline` lee la información de configuración del archivo `/etc/inputrc` y después de `~/.inputrc` o del archivo indicado por la variable de ambiente INPUTRC. En este archivo de

24. Cuando abre una terminal desde X-Window con un intérprete de comandos se dice que el intérprete es interactivo pero no de login. Dado que un intérprete de login se ejecuta máximo una vez cuando usted inicia una sesión mientras que uno interactivo puede ejecutarse muchas (una por cada terminal que abra), los archivos de configuración de cada uno son diferentes.

configuración pueden asociarse acciones con secuencias de teclas o pueden configurarse varios detalles de `readline`, por ejemplo para emplear teclas de vi y completar con la tecla **TAB** debe incluir las siguientes líneas:

```
set editing-mode vi
TAB: complete
```

o para asegurar que bash permita ingresar y presente caracteres propios de español (tildes, eñe, diéresis), puede agregar a este archivo las siguientes líneas (y eventualmente configurar la variable LANG en `~/.bash_profile` como se explica más adelante):

```
set input-meta on set output-meta on set convert-meta off
```

Por defecto la historia de comandos se almacena en el archivo especificado por la variable HISTFILE (normalmente `~/.bash_history`) y la máxima cantidad de comandos por almacenar se especifica en `HISTSIZE`.

Ciertos aspectos que afectan a bash y otros programas pueden ser configurados por medio de variables de ambiente. Aunque algunas ya se presentaron brevemente (ver Ambiente y variables de ambiente), a continuación se presentan detalles de otras:

LANG LANGUAGE

Estas variables junto con LANGUAGE, LC_CTYPE, LC_NUMERIC, LC_TIME, LC_COLLATE, LC_MONETARY, LC_MESSAGES y LC_ALL indican el idioma por usar a bash y a otros programas que soportan diversos idiomas [25] —la configuración de estas variables puede ser examinada con el programa **locale**. El soporte para diversos idiomas incluye el idioma en el que los programas presentan mensajes al usuario, el tipo de letra (porque varios lenguajes tienen su propio juego de caracteres), el orden en el que se presentan las fechas, el formato de números decimales, el formato de cantidades monetarias y otros detalles. Para activar un lenguaje particular basta poner las variables LANG y LANGUAGE en el código del lenguaje o *locale* que se desea usar. En el caso de español el código es `es` o si desea localización aún más particular de Colombia puede usar `es_CO` —el código para otros lenguajes sigue el estándar ISO 639 y generalmente corresponde a las dos primeras letras del idioma cuando se escribe en el mismo idioma. Las otras variables (LC_*) permiten afinar detalles de la localización con códigos diferentes a los de LANG y LANGUAGE: LC_CTYPE se refiere al tipo y clasificación de caracteres, LC_COLLATE a secuencias de varios caracteres que deben interpretarse como uno sólo (por ejemplo en español "ch" y "ll"), LC_TIME al formato en el que se presentan fecha y hora, LC_NUMERIC al formato de números, LC_MONETARY al formato de cantidades monetarias, LC_MESSAGE al idioma de los mensajes.

Si después de configurar estas variables en el valor `es_CO` no puede ver caracteres con tildes, por ejemplo desde vi o si los programas no presentan mensajes en español (pruebe por ejemplo entrando a dselect), intente con el valor `es_AR` o recuerdele al administrador configurar el locale para Colombia (ver Localización y configuración de algunos programas).

25. Un término para denotar soporte de diversos idiomas es NLS, que es sigla de *Native language support*. No todos los programas de su sistema emplearán esta característica y algunos aunque la emplean aún no tienen soporte para español —esa es otra oportunidad para ayudar. Si está interesado en colaborar en agregar soporte para español a Debian o traducir mensajes, páginas del manual o documentos, puede unirse al proyecto de traducción http://www.debian.org/international/spanish/proyectos.es.html

`TERM`

En esta variable se mantiene el tipo de terminal [26] que el usuario está empleando, las variables LINES y COLUMS mantienen el número de líneas y columnas de la terminal. El valor especificado en TERM es empleado por el intérprete de comandos para enviar secuencias de control [27] apropiadas para la pantalla y para interpretar la información proveniente del teclado. Por ejemplo si usted se conecta a un sistema Unix desde una terminal Wyse 60 debe poner en TERM el valor `wyse60`. Al conectarse desde otro PC con Linux o Windows puede emplear el valor `ansi` o en su defecto `vt100`, al abrir bash desde una ventana de X-Window se debe emplear el valor `xterm`. Puede consultar la lista de nombres que puede poner en la variable TERM en el directorio `/usr/share/terminfo`, esa ruta es la ubicación de la base de datos `terminfo`, la cual mantiene las secuencias de control para todas las terminales soportadas por su sistema —más de 800. Desde bash puede emplear el programa **clear** para borrar pantalla y puede consultar la base terminfo con el programa **tput**, por ejemplo para cambiar el tipo de letra a negrillas use

```
echo `tput bold`
```

Para configurar algunos parámetros de comunicación con su terminal emplee **stty** y para consultar el dispositivo con el cual el kernel identifica su terminal emplee **tty**.

PS1 PS2

Estas variables contienen prompts que bash presenta al usuario cuando está esperando información. PS1 es el prompt principal, que bash presenta cuando espera comandos, PS2 es presentado cuando bash espera más información para completar un comando —por ejemplo si usa **cat «EOF** (ver bash y el juego de herramientas).

Estas variables pueden tener mensajes, secuencias de control para su terminal (por ejemplo para borrar pantalla o ubicar el cursor) y algunas secuencias de caracteres que se interpretan de forma especial. El siguiente prompt borrará pantalla antes de presentarse [28], mostrará el mensaje `hola` seguido del nombre del usuario, la palabra `en` y el nombre de la máquina:

```
PS1="`tput clear`hola \u en \h: "
```

Las secuencias que se interpretan de forma especial son:

`\a`

Produce un sonido

`\d`

Fecha

`\e`

Caracter que representa inicio de una secuencia de control para su terminal (ASCII 33).

26. En este contexto terminal se refiere al tipo de teclado y pantalla que está empleando.
27. Secuencias de control se refiere a "instrucciones" para la terminal que indican operaciones como borrar pantalla, ubicar el cursor en una posición específica, cambiar colores, etc.
28. El ejemplo usa sustitución de comandos (ver Scripts básicos para bash) para sustituir el comando `tput clear` por el valor que retorna, es decir por la secuencia que su terminal emplea para borrar pantalla.

`\h`
: Nombre de la máquina hasta el primer '.'

`\H`
: Nombre completo de la máquina.

`\n \r`
: Nueva línea.

`\s \v \V`
: Nombre y versión del intérprete de comandos.

`\t \T \@`
: Hora en diversos formatos.

`\u`
: Nombre del usuario.

`\w \W`
: Directorio de trabajo.

`\$`
: El caracter '#' si el usuario es el administrador y '$' en otro caso.

3.5.1.2. Una sesión con X-Window

Una sesión con un administrador de escritorio como Gnome requiere una compleja infraestructura para funcionar. Gnome es un administrador de escritorio que maneja los menús y algunos programas como el manejador de archivos (otros administradores de escritorio son KDE y XFCE). Estos administradores de escritorio requieren un administrador de ventanas, el administrador de ventanas ubica las ventanas y maneja las barras de título y controles de cada ventana (*Tab Window Manager*, XFWM, Sawfish y Enlightment son ejemplos de administradores de ventanas). Los diversos elementos gráficos (como textos, botones, menús, etc) son presentados empleando una librería de *widgets* (en el caso de Gnome se usa GTK, otros ejemplo son Athena y Qt). Estas librerías emplean X-Window, que se encarga finalmente de comunicarse con el hardware para mostrar imágenes —X-Window también envía datos por la red en caso de que inicien aplicaciones gráficas en máquinas remotas (ver telnet y ssh).

Aunque es posible iniciar el servidor X con **startx**, normalmente el administrador del sistema configurará un administrador de vistas [29] que se ejecutará automáticamente duratne el arranque. El administrador de vistas iniciará X-Window y le permitirá ingresar su login y su clave con un programa gráfico. En el caso de GNOME, el administrador de vistas es **gdm** (que usa el progama **gdmlogin** para pedir al usuario su login y clave).

29. Administrador de vistas: del inglés *display manager*.

gdmlogin le permite configurar el administrador de escritorio que desea usar y el idioma en el que desea iniciar su sesión (es decir fijará la variable de ambiente LANG). Los eventos que ocurran después de que teclee su login y clave dependen del administrador de escritorio que escoja.

Si escoge una sesión sin administrador de escritorio (XSession), su archivo `~/.xsession` será ejecutado o si no existe, se ejecutará un script por defecto que iniciara un manejador de ventanas sencillo como twm y una terminal X.

A continuación se presenta un archivo `~/.xsession` típico:

```
#!/bin/sh

if (test -f $HOME/.Xresources;) then {
  xrdb -merge $HOME/.Xresources;
} fi
if (test -f $HOME/.Xmodmap;) then {
  xmodmap $HOME/.Xmodmap;
} fi
xmessage -file /etc/motd &
xterm -title zzz -g 70x35+1+1 &
xclock -g +500+0 &
xcalc -g -0+298 &

twm
```

Note que este archivo es un script para el intérprete de comandos —debe darle permiso de ejecución para que funcione correctamente—, el cual inicializa variables de ambiente, ejecuta diversos programas en el fondo (una ventana con un mensaje, una terminal X, un reloj y una calculadora) y que el último programa ejecutado (twm) no se ejecuta en el fondo. De hecho el último programa es un manejador de ventanas (Tab Window Manager), y no debe ejecutarse en el fondo porque la sesión X se cerrará cuando se cierre este. Los programas **xrdb** y **xmodmap**, que se explicarán a continuación, no son interactivos y terminan su ejecución rapidamente, por eso no requieren ser ejecutados en el fondo.

Algunas propiedades de aplicaciones X-Window como color, tipo de letra [30] y geometría (i.e ubicación y tamaño de la ventana) se denominan recursos, usted puede ajustarlos y puede hacer sus cambios durables creando un archivo de recursos. Para manejar recursos se emplea el programa **xrdb** que en particular permite mezclar los recursos del servidor X que está corriendo con los de un archivo. El archivo usual en el que se especifican recursos para aplicaciones X es `~/.Xresources`.

Para iniciar una aplicación X con recursos diferentes a los que tiene por defecto o a los configurados en `~/.Xresources` puede emplear opciones en la línea de comando, por ejemplo para iniciar **xterm** con un tipo de letra diferente se emplea la opción `-fn` seguida del nombre del tipo de letra:

```
xterm -fn "-adobe-avant garde gothic-book-r-normal-14-120-75-*-p-80-iso8859-1"
```

El nombre del tipo de letra puede escogerlo con ayuda del programa fontsel. Para cambiar la geometría de una ventana (tamaño y ubicación) puede emplear la opción `-geometry` o `-g` como se uso con los programas del archivo `~/.xsession` de ejemplo (puede examinarse la geometría de una ventana ya abierta con el programa **xwininfo**). Para cambiar los colores puede emplear las opciones `-fn` y `-bg` para

30. Tipo de letra: del inglés *font*

especificar colores de la letra y del fondo respectivamente; a continuación de estas debe especificarse el nombre del color, el cual puede ser escogido con ayuda del programa `xcolors`; por ejemplo:

```
xterm -geometry 100x30 -fg yellow -bg red
```

En un archivo de recursos como `~/.Xresources` pueden especificarse las propiedades de los ejemplos con las siguientes líneas:

```
xterm*Foreground: yellow
xterm*Background: red
xterm*VT100.geometry: 100x30
xterm*font: -adobe-avant garde gothic-book-r-normal-14-120-75-*-p-80-iso8859-1
```

Puede aprender más sobre recursos mientras explora los de las aplicaciones abiertas con el programa **editres**.

La configuración del teclado puede alterarse, por ejemplo para intercambiar dos teclas o para activar teclas muertas en un teclado sin tildes [31], empleando el programa **xmodmap**. Por ejemplo para configurar el apóstrofe como tecla muerta para la tilde en un teclado `us`:

```
xmodmap -e "keycode 48=dead_acute"
```

La configuración del teclado puede mantenerse en el archivo `~/.Xmodmap` que normalmente es leído cada vez que inicia una sesión gráfica. Por ejemplo para configurar teclas muertas para la virgulilla como **~**, la tilde como **'**, la dieresis como **"** y la tecla múltiple [32] como **Alt Izquierdo** en un teclado `us`:

```
keycode 64=Alt_L Multi_key
keycode 48=dead_acute dead_diaeresis
keycode 49=dead_grave dead_tilde
```

Los códigos de cada tecla pueden examinarse con el programa **xev**. El mapa del teclado que está configurado en X-Window puede consultarse con la opción `-pk` de **xmodmap**, las teclas modificadoras (con significado especial) pueden consultarse con la opción `-pm`, los nombres de las teclas en el archivo `/usr/include/X11/keysymdef.h`.

X-Window empleará tipos de letra por defecto que puede listar con los frogramas **xfontsel** y **xlsfonts**, si requiere otras fuentes que no están disponibles en su servidor X, debe darla a conocer al servidor X antes de usarla:

```
cp nuevafuente.pcf ~/misfuentes
mkfontdir ~/misfuentes
xset fp rehash
```

31. Teclas muertas son teclas que permiten ingresar caracteres propios del español en teclados que no tienen teclas para estos caracteres, por ejemplo cuando el apóstrofe se configura como tecla muerta para el acento, al presionar ' y después a se produce á; para producir el apóstrofe su presionaría el apóstrofe y después la barra espaciadora

32. La tecla múltiple o *compose* cambia el comportamiento por defecto de algunas teclas para generar caracteres de diversos alfabetos.

3.5.1.3. Una sesión con Gnome

La mayoría de características de Gnome pueden ser configuradas desde el centro de control Gnome (ver Lectura Escritorio y herramientas Gnome), sin embargo como Gnome emplea X-Window algunas características deben configurarse de forma diferente, por ejemplo la configuración del teclado y los recursos de algunas aplicaciones X-Window los puede configurar como se explicó en la sección anterior (archivos `~/.Xmodmap` y `~/.Xresources`). La configuración del administrador de ventanas normalmente la podrá hacer desde el centro de control Gnome.

Gnome permite resumir una sesión en el estado que quedó la última vez que la cerró (característica llamada administración de sesiones). Puede mantener varias sesiones de este tipo e iniciarlas escogiendo la que desee desde gdmlogin. Para guardar una sesión con un nombre nuevo emplee el centro de control Gnome (menú Sesiones) o edite el archivo `~/.gnome/sessions`.

La administración de sesiones funciona bien con aplicaciones diseñadas para Gnome, pero no siempre para otras. Si debe ejecutar alguna aplicación o comandos al inicio de cada sesión Gnome y no logra configurarlas con la administración de sesiones, agréguelos al archivo `~/.gnomerc` o emplee el centro de control Gnome (menú Sesiones) para especificar la aplicación por iniciar.

La configuración de varias aplicaciones Gnome la encuentra en el directorio `~/.gnome`. Los menús que Gnome presenta pueden ser configurados sólo por el administrador, excepto el menú "Favoritos" que puede ser editado por cada usuario. Para editarlo emplee el botón derecho sobre el pánel, elija el menú pánel y después edición de menús.

3.5.2. Lecturas recomendadas: Configuración de una sesión

- Puede ver otra explicación sobre la configuración de bash en *G.L.U.P. - Guía de Linux Para el Usuario*. Sección 9.1 Personalización del bash. http://lucas.hispalinux.es/Manuales-LuCAS/GLUP/glup_0.6-1.1-html-1.1/guide09.htm
- bash puede configurarse aún más con los comandos **set** y **shopt**. Puede consultar más sobre ambos en la página del manual de bash.
- La base de datos terminfo es común a todos los sistemas libres tipo Unix (e.g FreeBSD, OpenBSD, Linux). Dicha base e información sobre la misma pueden obtenerse en http://www.tuxedo.org/~esr/terminfo/ Puede consultar más sobre terminales y terminfo en las páginas del manual de **terminfo**, **tput** y **stty**.
- Puede aprender más sobre localización e internacionalización en el manual info de gettext (Verr (**info gettext**).) —`gettext` es la librería usada por Linux para manejar mensajes en varios idiomas. En Internet puede consultar el sitio del proyecto de internacionalización de Linux http://www.li18nux.org o el proyecto de localización a español de Debian http://www.debian.org/international/spanish/proyectos.es.html.
- Para aprender más sobre la configuración y operación de X-Window, puede consultar algunas páginas del manual (e.g **xmodmap**, **editres**), en especial recomendamos **man X**.
- Para aprender más sobre los recursos de X-Window (y sobre X-Window) puede revisar la sección "The X Resources" de "*X Window User HOWTO*". Disponible en su sistema en

`/usr/doc/HOWTO/en-txt/XWindow-User-HOWTO.txt.gz` o en Internet http://www.linuxdoc.org/HOWTO/XWindow-User-HOWTO-8.html.

- Para aprender más sobre el gestor de sesiones de Gnome puede consultar la sección "Gestor de Sesiones" de la Guía de usuario GNOME, disponible en su sistema en `file:/usr/share/gnome/help/users-guide/C/gccsession.html` o en Internet en http://www.croftj.net/~barreiro/spanish/gnome-es/users-guide/session.html

3.5.3. Ejercicios: Configuración de una sesión

1. Edite su archivo `~/.bash_profile` y agregue líneas para que ejecute el archivo `~/.bashrc` en caso de que exista. Después ejecute un intérprete de comandos de ingreso para comprobar que el cambio funciona.

2. En su archivo `~/.bashrc` agregue la vía `/opt/bin` a la variable de ambiente PATH y después agregue y exporte la variable LD_LIBRARY_PATH con el valor `/opt/lib` —ambos son cambios requeridos por nuestra plataforma de referencia (ver Plataforma de referencia) y le permitirán ejecutar algunos programas que estén en el servidor.

3. Edite el archivo de configuración de `readline` y agregue una línea con:

```
"\C-xp": "PATH=${PATH}\e\C-e\C-a\ef\C-f"
```

después reinicie su sesión de bash y presione **Control-x p**. ¿Qué acción asoció con esa secuencia de teclas? *Ayuda*: para entender la notación y las teclas vea la sección sobre emacs (ver Editores de texto)

4. Momentáneamente configure una sesión de bash para que presente mensajes en otros idiomas (por ejemplo en inglés cuyo código es `en`). Pruebe después su cambio examinando los mensajes de algún programa o una página del/manual.

5. Modifique el prompt de bash para que cada vez que se presente emita un sonido y muestre el nombre del usuario y la hora.

6. Configure GNOME para que al iniciar su sesión abra el calendario de Gnome (gnomecal) y el programa emacs (Ayuda: gnomecal es una aplicación Gnome mientras que emacs no).

7. Configure emacs para que comience con colores diferentes. *Ayuda:* debe modificar los recursos, puede usar **editres** para ver algunos o consultar la sección "X Resources" del manual de Emacs.

Capítulo 4. Desarrollo de habilidades para administrar computadores con Linux

Público y logros

Personas que administren un computador con Linux Debian no necesariamente conectado en red (ver Plataforma de referencia) [1]

- Entiende la responsabilidad como superusuario de una máquina y tiene una conducta ética para llevar a cabo su papel.
- Implementa una política para encendido y apagado de la red.
- Administra los usuarios del sistema.
- Administra los archivos y sistemas de archivos.
- Administra los programas instalados.

4.1. El rol del administrador de red

Indicadores de Logros:

- Conoce el rol de administrador de red.

4.1.1. Lectura: El rol de administrador

Un administrador de red sirve a los usuarios: crea espacios de comunicación, atiende sugerencias; mantiene las herramientas y el espacio requerido por cada usuario, a tiempo y de buena forma (piense si usted fuera usuario como le gustaría que fuera el administrador); mantiene en buen estado el hardware y el software de los computadores y la(s) red(es) a su cargo; mantiene documentación que describe la red, el hardware y el software que administra; respeta la privacidad de los usuarios y promueve el buen uso de los recursos. A cambio de tantas responsabilidades la recompensa es el buen funcionamiento de la red como un medio que vincula personas y de los computadores y programas como herramientas para agilizar algunas labores que dan tiempo y dar tiempo para realizar otras.

El administrador de red debe conocer las claves de la cuenta `root` de las máquinas que administra. Desde esa cuenta puede configurar servicios y establecer políticas que afectarán a todos los usuarios. Algunas de las labores que sólo pueden hacerse desde esta cuenta son:

- Apagar el equipo (aunque por defecto en Debian esta operación la puede realizar cualquier usuario).
- Configurar los programas que se inician junto con el sistema.
- Administrar cuentas de usuarios.
- Administrar los programas y la documentación instalada.

1. Para realizar la mayoría de ejercicios de estas guías necesita acceso a la cuenta `root`.

- Configurar los programas y los dispositivos.
- Configurar la zona geográfica, fecha y hora.
- Administrar espacio en discos y mantener copias de respaldo.
- Configurar servicios que funcionarán en red.
- Solucionar problemas con dispositivos o programas. Labor que resultaq en ocasiones la más dispendiosa, pero que se facilitará entre más aprenda del sistema y la red a su cargo.
- Revisar bitácoras [2], solucionar y prevenir inconvenientes de seguridad.

Por las posibilidades que se tienen desde esta cuenta, no debe emplearla para realizar operaciones como usuario (cree más bien una cuenta de usuario) y debe evitar que otras personas tengan la clave de esta cuenta (comience por emplear una clave segura ver Sistema de usuarios y manejo de clave).

Al igual que los demás usuarios, la cuenta `root` tiene un espacio independiente (`/root`), sin embargo la mayoría de configuraciones se hacen editando archivos del directorio `/etc`, otras se hacen ejecutando programas disponibles en los directorios `/usr/sbin` y `/sbin` y las bitácoras se examina con programas (e.g **last**), o en archivos del directorio `/var/log`.

4.1.2. Lectura recomendadas: El rol de administrador

- Secciones 4.6.6, 4.6.7, 4.6.8 y 4.6.9 de "*Linux Installation and Getting Started*" http://www.linuxdoc.org/LDP/gs/node6.html
- Revise los tres primeros grupos de guías. Usted como administrador podría ayudar a los usuarios a responder preguntas que los usuarios hagan sobre estas guías (o las que use para guiar a los usuarios en su institución).
- Opcional: Sección "Acerca de Raíces, Sombreros y la Sensación de Poder" de "*Linux Instalación y Primeros Pasos (LIPP)*" disponible en http://lucas.hispalinux.es/Manuales-LuCAS/LIPP/lipp-1.1-html-2/lipp4.htm.

4.1.3. Ejercicios: El rol de administrador

1. Piense y discuta en clase mecanismos para:

1. Crear espacios de comunicación con usuarios en un colegio
2. Mantener documentación útil para el administrador y para los usuarios.

2. Una de las mayores dificultades tras instalar Linux es la aceptación por parte de los usuarios (quienes generalmente están acostumbrados a otro sistema operativo). ¿Cómo puede medirse la aceptación y qué puede hacerse para mejorarla?

3. ¿Cómo puede facilitar a los usuarios la utilización del sistema?

2. Bitácora: del inglés *log file* archivo donde se registran eventos importantes del sistema.

4.2. Apagado y encendido de una red

Indicadores de Logros:

- Implementa una política para el apagado de los computadores.
- Inicia una red Linux y puede solucionar eventuales inconvenientes desde el nivel de ejecución 1.
- Emplea algunas bitácoras relacionadas con el inicio del sistema y de sesiones.

4.2.1. Lectura: Apagado y encendido de una red

Los sistemas Unix fueron diseñados para brindar operación continua (no requieren ser apagados), por eso antes de apagar cada computador debe "cerrar" el sistema operativo con el programa **shutdown**. En esta sección se presentan indicaciones para apagar y encender una red Linux así como la forma de pasar parámetros al kernel cuando se inicia.

4.2.1.1. Apagado

Para cerrar y apagar puede emplear **halt**, **poweroff** o **shutdown -h now**. Y para cerrar y reiniciar puede emplear **reboot** o **shutdown -r now**.

Los programas **halt**, **poweroff** y **reboot** realizan las siguientes acciones:

- Cierran el sistema ejecutando **shutdown** [3] . Este programa envía un mensaje a todos los usuarios indicando que el sistema será apagado, envía a todos los procesos la señal `SIGTERM` (ver Procesos y tareas) y cambia el nivel de ejecución a 0 si se está apagando, 6 si está reiniciando o 1 si se ejecuta desde la línea de comandos sin las opciones `-h` ni `-r`. Al cambiar el nivel de ejecución se ejecutarán otras acciones especificadas en `/etc/inittab` como se explica en el siguiente capítulo (ver Inicialización del sistema).
- Salvan en disco información que quede en memoria. Esto es necesario porque para hacer eficiente el uso de discos, parte de la información de estos es mantenida por el sistema operativo en memoria (en *cache*) y sólo se escribe a disco cuando resulta indispensable o cuando el usuario lo solicita explicitamente con el programa **sync**. Los programas **halt**, **reboot** y **poweroff** pueden no efectuar esta operación con la opción `-n`.
- Registran la acción de apagar en la bitácora `/var/log/wtmp`. Esta acción no se realiza si se emplea la opción `-d` y se efectuará pero sin reiniciar o apagar con la opción `-w`.
- Detienen interfaces de redes y apagan o reinician el computador. En máquinas que empleen APM [4] los programa **halt** y **poweroff** apagaran el computador después de cerrar el sistema. En caso de que un

3. Cuando **shutdown** es ejecutado desde la línea de comandos recibe como parametro la hora a la que debe cerrar (**now** indica en el mismo instante), como segundo parámetro opcional el mensaje por presentar a los usuarios, y entre las opciones que acepta `-h` indica apagar después de cerrar, `-r` reiniciar, `-f` para evitar el chequeo de discos la siguiente vez que inicie el sistema, `-F` para obligar un chequeo de discos cuando el sistema reinicie, `-c` para cancelar una llamada previa a `shutdown`, `-t seg` para que realice una pausa de los segundos especificados entre el envío de la señal SIGTERM y el cambio de nivel de ejecución (puede servir para dar más tiempo para cerrarse a algunos procesos).
4. APM: *Advanced Power Management*, permite manejar la energía consumida por el computador, por ejemplo para disminuirla cuando no se usa durante cierto tiempo o para apagarlo por software.

computador emplee APM pero el comando **halt** no lo apague, agregue al archivo `/etc/lilo.conf` la línea

```
append="apm=on"
```

y después ejecute el programa **/sbin/lilo**. Esta línea indica al kernel durante el arranque que debe activar el soporte para APM (ver Inicialización del sistema).

Dada la importancia de cerrar el sistema antes de apagarlo, puede tomar medidas para que los usuarios no puedan hacerlo:

- Por defecto gdm permite que todo usuario apague el computador con una opción del menú. Esto puede cambiarse editando el archivo `/etc/gdm/gdm.conf` (línea `HaltCommand`) o empleando la herramienta de configuración de gdm (menú System/Configure).
- Por defecto al presionar **Ctrl-Alt-Del** se ejecutará el programa **shutdown**. Puede desactivarse por completo esta secuencia en el archivo `/etc/inittab` eliminando o poniendo como comentario (poniendo un caracter '#' al comienzo) la línea de la forma `ca:12345:ctrlaltdel:/sbin/shutdown` *`opciones`*. Otra opción es agregar a tal línea la opción `-a` que indica a **shutdown** sólo apagar el sistema en caso de que uno de los usuarios que tengan sesiones abiertas cuando se presiona **Ctrl-Alt-Del** aparezca en el archivo `/etc/shutdown.allow`. Note que esto solamente ocurre en una interfaz no gráfica

Otra alternativa es enseñar a los usuarios por qué y cómo cerrar el sistema.

4.2.1.2. Encendido

Cuando inicie una red, encienda primero el servidor. Eventualmente al arranque de algún computador podría realizarse una revisión de discos que toma más tiempo de lo normal (esta revisión se evita con la opción `-f` de **shutdown**). En caso de que el sistema no haya sido cerrado antes de apagar, la revisión tomará más tiempo, tratará de resolver inconsistencias en el sistema de archivos, eventualmente puede requerir la intervención del administrador para completar la revisión con el programa **fsck** y en un caso extremo se perderá información o no podrá iniciarse el sistema.

En caso de requerir intervención manual durante el arranque, entrará a nivel de ejecución 1 (también llamado modo monousuario o *single*), aunque en algunos casos ni siquiera podrá entrar a modo mono usuario y deberá recurrir a un disquette de rescate (cuando Linux es instalado tiene la oportunidad de crearlo).

Tanto en modo monousuario como cuando emplee un disquette de rescate, no podrá emplear X-Window, así que debe estar preparado para operar desde la consola modo texto, por ejemplo para:

- editar archivos de configuración con un editor que opere en modo texto como vi o emacs con la opción `-nw` (ver Editores de texto),
- manejar el intérprete de comandos y el juego de herramientas Unix (ver bash y el juego de herramientas) y
- usar diversos programas para configurar el sistema (que se estudian en este capítulo y el siguiente).

Desde un sistema que haya iniciado normalmente (nivel de ejecución entre 2 y 5), puede pasar a nivel de ejecución 1 con los comandos **shutdown now** o **init 1**. También puede iniciar este nivel desde el cargador de arranque, antes de entrar al sistema, pasando como parámetro al kernel la opción `single`.

El cargador de arranque es un programa que se ejecuta tras encender el computador, y cargar el kernel pasandole eventuales parámetros. En Debian 2.2 el cargador de arranque por defecto es LILO, aunque puede remplazarse con otros como GRUB o Loadlin. Estos cargadores permiten cargar más de un sistema operativo (en caso de que tenga en sus discos varios y que lo haya configurado apropiadamente).

Notará que en Debian para entrar a modo mono usuario es necesario dar la clave de `root`, si la olvida puede pasar como parámetro *`init=/bin/sh`* desde el cargador de arranque o puede iniciar con un disquette de rescate. Dado que la posibilidad de pasar parámetros podría en algunos casos ser una falla de seguridad, puede configurar el cargador de arranque para restringirla.

4.2.1.3. Parámetros desde el cargador de arranque

Puede determinar si su cargador de arranque es LILO, observando si durante el arranque, después de completar la inicialización del BIOS, aparece una línea que comienza con la palabra LILO. GRUB puede presentar un menú o presentar una línea de comandos.

4.2.1.3.1. Parámetros desde LILO

Para permitirle elegir el sistema por usar durante el arranque, LILO presenta un prompt, desde el cual puede escribir la etiqueta del sistema por cargar (las etiquetas disponibles se presentan presionando **TAB** o examinando `/etc/lilo.conf`). En caso de que haya configurado LILO para que no presente prompt, puede hacer que aparezca presionando la tecla **shift** [5] durante el arranque cuando aparezca la palabra LILO.

Una vez en el prompt puede pasar parámetros al kernel dándolos a continuación de la etiqueta del sistema y separando unos de otros con espacios. Así por ejemplo para pasar a modo mono usuario, suponiendo que `linux` es la etiqueta de Linux:

```
linux single
```

Las etiquetas pueden configurarse en `/etc/lilo.conf` (líneas de la forma `label=`*`etiqueta`*), así como el prompt (línea de la forma `message=`*`archivo`*).

> Tenga en cuenta que después de modificar `/etc/lilo.conf` debe ejecutar el programa **/sbin/lilo** para que sus cambios sean instalados en el MBR o en el sector donde LILO está instalado. (sección cargador de arranque ver Discos duros).

Por seguridad es posible restringir con una clave el paso de parámetros con las líneas:

```
password=clave
restricted
```

de las cuales la primera establece una clave para LILO y la segunda indica que sólo debe solicitarse si se pasan opciones a una etiqueta. Para evitar que la clave pueda ser vista por otros, establezca los permisos

5. LILO puede configurarse para presentar el prompt siempre (agregando una línea con la palabra `prompt` a `/etc/lilo.conf`) y para iniciar un sistema por defecto después de que transcurra un tiempo límite (especificado en una línea de la forma `delay=`*`tiempo en decimas`*).

de `/etc/lilo.conf` para que sólo pueda ser leido y escrito por el usuario root (ver Archivos y permisos).

Puede hacer el paso de un parámetro permanente (es decir que se realice automáticamente cada vez que inicia), agregandolo a la línea **append** (o creando esta línea) a continuación de la etiqueta de Linux. Por ejemplo para indicar al kernel que el computador tiene 128Mb de RAM (en caso de que no lo detecte ver Memoria RAM), la parte de LILO que corresponde a Linux puede ser algo como:

```
image=/vmlinuz
        label=linux
        read-only
        optional
        append="mem=128M"
```

4.2.1.3.2. Parámetros desde GRUB

GRUB tiene un modo interactivo desde el cual puede cargar el kernel que desee (si tiene varios en su computador). Desde ese modo para iniciar Linux suponiendo que está en la raíz, y que la raíz está en la primera partición del primer disco IDE se emplea:

```
kernel (hd0,0)/vmlinuz root=/dev/hda1 single
```

También puede configurarse un menu para GRUB que permitirá al usuario elegir un sistema y ejecutar los comandos necesarios para iniciarlo y las opciones para el kernel. Tal menu se configura en el archivo `/boot/grub/menu.lst`, que podría contar con entradas diferentes para iniciar en modos diferentes como se ejemplifica a continuación (suponiendo que el kernel y la raíz están en la primera partición del disco IDE primario):

```
# Por defecto, iniciar primera entrada
default 0
# Arrancar automaticamente en 30 segundos
timeout 30
# Entrada del menu para iniciar Linux
title  Debian
kernel /vmlinuz root=/dev/hda1
# Entrada del menu para iniciar Linux en modo mono usuario
title  Debian
kernel /vmlinuz root=/dev/hda1 single
```

Al emplear un menú un usuario también puede emplear el modo interactivo de GRUB, para evitarlo puede proteger con clave el modo interactivo agregando **password *su_clave*** al archivo de configuración [6] Otra opción es que cree dos menús, uno para usuarios normales (`/boot/grub/menu.lst`) y otro para administradores (digamos `/boot/grub/menu-admin.list`) y que proteja el segundo con clave empleando **password *su_clave* /boot/grub/menu-admin.list**.

6. Con GRUB puede poner en el archivo de configuración una clave transformada con el algoritmo MD5 empleando `password -md5 clave_transformada`.

4.2.1.3.3. Algunos parámetros para el kernel

La referencia completa es *BootPrompt-HOWTO*, a continuación presentamos algunos parámetros:

`root=`*`disp`*

Por ejemplo `root=/dev/hdd1` o en lugar de `/dev/hdd1` la partición que debe montarse como directorio raíz (útil por ejemplo si debió cambiar su disco duro con Linux a otra interfaz o si agrega o elimina particiones). Por defecto será el configurado en `/etc/lilo.conf` en la sección apropiada en una línea de la forma:

```
root=/dev/hda2
```

Los dispositivos que podrían especificarse son IDE (e.g `/dev/hda1`), SCSI (e.g `/dev/sda1`), NFS en caso de una terminal que cargue directorio raíz por red (`/dev/nfs`), imagen `initrd` como raíz (`/dev/ram` ver Configuración del kernel)

`ro`

Para montar la partición raiz en modo sólo lectura.

4.2.1.4. Bitácoras

Las bitácoras en Debian están configuradas por defecto en archivos del directorio `/var/log`, hay varias de acuerdo al programa que registra los mensajes, todas excepto `wtmp` y `btmp` son textos que puede examinar con un editor o un paginador (e.g `less`):

`auth.log`

Mantiene mensajes de autenticación, producidos por ejemplo por la librería PAM indicando que usuarios abrieron y cerraron sesiones. Manejado por syslog.

`kern.log`

Mensajes del kernel. Manejado por syslog.

`btmp, wtmp`

`wtmp` mantiene información de usuarios que han abierto o cerrado sesiones, se examina con el programa **last** (está relacionada con `/var/run/utmp` que mantiene información de los usuarios que están conectados —usado por el programa **who**). `btmp` mantiene información de sesiones que se intentaron abrir pero que no pudieron autenticarse, se examina con el progama **lastb**.

`lpr.log`

Mensajes sobre impresoras. Manejado por syslog.

`mail.log, mail.err, mail.info, mail.warn, exim`

Mensajes sobre correo. Todos excepto exim son manejados por syslog.

`user.log`

Mensajes de diversos programas, tipo **user**. Manejador por syslog.

`messages`

Mensajes informativos de diversos programas (e.g del kernel). Manejado por syslog.

`daemon.log`

Mensajes varios.

`debug`

Mensajes de depuración de algunos programas, empleados usualmente por los desarrolladores para encontrar fallas.

`uucp.log`

Empleado por el sistema *Unix to Unix Copy* (transferencia de archivos en algunas redes no tan modernas o sin muchos recursos).

`news/news.crit news/news.err news/news.notice`

Empleado por un servidor de Netnews.

Algunas bitácoras son manejadas por el programa **syslogd** (paquete `sysklogd`), que se ejecuta durante el arranque y espera mensajes. Cada mensaje que recibe de algún programa, lo registra junto con la fecha y la procedencia aun archivo o dispositivo (e.g. una consola). Cada mensaje tiene un tipo y una prioridad, que permiten determinar el archivo o dispositivo al cual se envia. La relación entre tipos y prioridades con archivos se configura en el archivo `/etc/syslogd.conf` cuyas líneas pueden ser comentarios, o similares a:

```
news.crit                    /var/log/news/news.crit
```

Donde *`news`* se remplaza por el tipo de mensaje, *`crit`* se remplaza por la prioridad y `/var/log/news/news.crit` por el archivo o dispositivo al que se deben enviar los mensajes con ese tipo y prioridad. Los tipos posibles son: `authpriv` o `security` empleados por programas que realizan autenticación de usuarios, `cron`, `daemon`, `kern`, `lpr`, `mail`, `mark`, `news`, `security`, `syslog`, `user`, `uucp`, `local0` a `local7`. Las prioridades posibles son: `debug`, `info`, `notice`, `warning`, `err`, `crit`, `alert`, `panic`. También puede emplearse * para especificar todos los tipos o todas las prioridades.

Los programas que emplean syslogd, enviarán mensajes con tipos acordes con la funcionalidad del programa y con prioridad acorde con la gravedad o importancia. Para agregar eventos a una bitácora desde el interprete de comandos (o desde un script) puede emplear el programa **logger** (paquete `bsdutils`). Para revisar los archivos empleados por syslogd, puede ejecutar **syslogd-listfiles -a**.

Dado que las bitácoras pueden crecer mucho (algunas en un día), en Debian son rotadas con cierta frecuencia (diaría, semanal o mensual) empleando cron (ver Tiempo), en particular `syslog` y `auth.log` son rotados a diario, `mail.log`, `daemon.log`, `messages`, `lpr.log`, `auth.log`, `debug`, `mail.err`, `mail.info`, `kern.log`, `mail.warn`, `uucp.log` y `user.log` son rotados semanalmente con el programa `savelog` (ver `/etc/cron.weekly/syslogd`). Por defecto se rota mensualmente el archivo `wtmp` con el programa logrotate y se genera un reporte de la actividad en `wtmp.report` (ver `/etc/cron/weekly/logrotate`).

4.2.2. Lectura recomendadas: Apagado y encendido de una red

- Páginas del manual de los programas **shutdown**, **halt**, **reboot** y **poweroff**.
- Para aprender más sobre bitácoras puede consultar las páginas del manual de **last**, **blast**, **lastlog**, **lastcomm**, **syslogd**, **syslog.conf**, **logrotate** y **savelog**.
- La bitácora `/var/account/pacct` registra información sobre los procesos iniciados por cada usuario y que permiten sacar estadísticas de uso por usuario y programas. Puede consultar sobre esta bitácora y los programas que la usan en las páginas del manual de **ac**, **accton**, **last**, **lastcomm** y **sa**, o puede ver el manual info **accounting**.
- El programa **fsck** permite chequear discos y eventualmente reparar algunos errores. Lo que hace es ejecutar un programa apropiado para el sistema de archivos del disco, el sistema de archivos propio de Linux es `ext2`, que es chequeado con el programa **e2fsck**. Otros programas apropiados para otros sistemas de archivos los encunetra en el directorio `/sbin` con nombres de la forma `fsck.`*`sisarch`*. Para conocer más sobre estos consulte las páginas del manual.
- Los manuales de LILO y GRUB son muy completos. Puede consultar el manual de LILO en `/usr/doc/lilo/Manual.txt.gz` y en caso de que instale GRUB en ver (**info grup**)el manual del usuario de GRUB. En Internet el manual de LILO está disponible en postscript en http://www.ibiblio.org/pub/Linux/system/boot/lilo/!INDEX.html y el de GRUB en http://www.gnu.org/manual/grub-0.90/html_mono/grub.html.

4.2.3. Ejercicios: Apagado y encendido de una red

1. Escoja una política para el apagado de los equipos y describa como la implementaría.

2. Emplee el programa **shutdown** desde la línea de comandos para pasar al nivel de ejecución 1 (también llamado modo *single*). ¿Qué procesos hay cuando inicia este modo? ¿Por qué se llama *single* a este modo?

3. Emplee el programa **last** para examinar la bitácora `/var/log/wtmp`, ¿qué información se consigna en esa bitácora?

4. Las bitácoras en Debian 2.2 son rotadas periódicamente por el program logrotate. Por defecto la bitácora `/var/log/wtmp` se borrará mensualmente (quedará una copia de la bitácora del mes anterior en `/var/log/wtmp.1` y un reporte de la actividad del mes anterior en `/var/log/wtmp.report`. El periodo de rotación o la cantidad de meses pasados que se deben mantener puede configurarse en el archivo `/etc/logrotate.conf`. Configure ese archivo para que almacene la información de los 2 últimos meses. ¿Qué otras características pueden configurarse con `logrotate` ?

5. Una bitácora que puede resultar importante para detectar fallas de seguridad es `/var/log/btmp` que mantiene intentos fallidos de iniciar sesiones en el sistema [7]

6. Los comandos dados por cada usuarios son registrados por `bash` (ver Lectura Configuración de una sesión) y los procesos iniciados son registrados por `acct`. Para examinar la bitácora de los procesos

7. Otra bitácora que mantiene información sobre logins de usuarios es `/var/lastlog` que se examina con el programa `lastlog`

iniciados puede emplear **lastcomm** y para ver estadísticas **ac** y **sa**. ¿Qué información presentan estos programas y para que pueden servir?

7. Restrinja el cargador de arranque para que requiera clave antes de pasar parámetros al kernel.

4.3. Administración de usuarios

Indicadores de Logros

- Implementa una política para administración de usuarios: grupos, usuarios, permisos.
- Configura el sistema buscando mantener privacidad y hacer más fácil la adaptación de usuarios nuevos.

4.3.1. Lecturas: Administración de usuarios

El objetivo de los usuarios, grupos y permisos en Unix es brindar privacidad y organización permitiendo también compartir información cuando así se desea.

Con respecto a usuarios el administrador puede crear, eliminar o modificar información de cuentas y grupos. También puede configurar detalles de los programas que permiten iniciar sesiones (ver Lectura Configuración de una sesión) y prevenir fallas de seguridad. Si aún no lo ha hecho, antes de consultar esta sección recomendamos estudiar la visión que un usuario tiene del sistema de usuarios y grupos (ver Sistema de usuarios y manejo de clave).

En una red con NIS la información de cuentas y grupos está centralizada en el servidor. Desde este debe hacerse la administración con los programas y archivos presentados en esta sección y después de cada cambio debe reconfigurarse NIS como se explicará posteriormente (ver Servicio NIS).

4.3.1.1. Usuarios

Los siguientes programas permiten realizar operaciones relacionadas con usuarios:

su [*`usuario`*]

Permite entrar a una sesión como un usuario diferente, si no se especifica un usuario como primer parametro, **su** cambia al usuario root. Desde una cuenta de usuario pedirá la clave del nuevo usuario, e iniciará el intérprete de comandos que el usuario haya configurado. Si se emplea la opción `-c` *`comando`* el comando se ejecutará. Por defecto el ambiente y el directorio de trabajo serán heredados de la sesión inicial, pero si se desea iniciar con un ambiente y directorio como el que tendría el usuario después de hacer login (pero sin ejecutar archivos de inicialización del interprete de comandos), emplee la opción **-** [8]. Si desea ejecutar un intérprete de comandos diferente al configurado por el usuario emplee la opción `-s` *`intérprete`*, donde *`intérprete`* debe ser la ruta completa del intérprete de comandos que además debe estar listada en el archivo `/etc/shells` [9]. En Debian 2.2 **su** al igual que **login** emplea los servicios de la librería PAM (*Pluggable*

8. Al emplear la opción - de **su** se inicializa el ambiente tal como el programa **login** lo haría, el valor de la variable PATH será tomado de `/etc/login.defs`, de ENV_PATH si se pasa a un usuario normal o de ENV_SUPATH si se trata del usuario `root`
9. En `/etc/shells` se listan los intérpretes de comandos disponibles. Algunos programas que opean en red sólo aceptan conexiones de usuarios cuyo intérprete de comandos esté listado en ese archivo.

authentication modules), así que puede configurar varios detalles relacionados con seguridad de **su** en el archivo `/etc/pam.d/su` —por ejemplo para restringir el uso de este comando.

passwd [*`usuario`*]

Permite cambiar la clave del usuario especificado (sólo root puede especificar un usuario). Si no se especifia un usuario permite cambiar la clave de la cuenta desde la cual se ejecuta. Por defecto un usuario podrá cambiar su clave cuando él/ella lo desee, aunque puede implementarse una política de expiración de claves. La política de un usuario puede examinarse con la opción `-S` que presenta: estado de la cuenta (L bloqueada, NP sin clave, P con clave utilizable), fecha del último cambio de clave, tiempos mínimo y máximo para cambiar clave, tiempo de avisos y tiempo para desactivación. Puede fijarse la política con las siguentes opciones: `-x` *`días`* para especificar el máximo de días antes de que un usuario deba cambiar su clave [10]; `-w` *`días`* para indicar con cuantos días de anterioridad a la expiración de la clave el sistema debe empezar a enviar correos recordando el cambio; `-i` *`días`* índica el mínimo número de días que el usuario debe esperar para cambiar nuevamente su clave; `-i` *`días`* para deshabilitar una cuenta cuya clave no haya sido cambiado en los días especificados. Una cuenta puede bloquearse para impedir que pueda usarse con el programa login. Pueden bloquear y desbloquear cuentas con las opciones `-u` (de *unlock*) y `-l` respectivamente.

chage *`usuario`*

Cambia la información sobre el envejecimiento de la clave de un usuario, cuando se están usando claves shadow. Para cambiar el mínimo de días entre cambios se usa la opción `-m` *`días`* (0 índica que puede cambiarse siempre), `-M` *`días`* para indicar máximo de días antes de la expiración de la clave, `-d` *`días`* para establecer fecha del último cambio de clave (contada en días a partir de 1/Ene/1975), `-E` *`fecha`* fecha en la cual la cuenta expirar (puede especificarse como una fecha mm/dd/aaaa o contada en días a partir de 1/Ene/1975), `-I` *`días`* establece cantidad de días de inactividad antes de bloquear una cuenta después de la expiración de la clave, `-W` *`días`* cantidad de días de preaviso antes de la expriación de la clave.

adduser *`login`*

Permite agregar un usuario con el login dado [11], opcionalmente indicando el directorio de trabajo (opción `--home`) que de no existir será creado, un número que identificará al usuario de forma única (opción `uid`), el grupo principal al que pertenecerá (opción `--ingroup` *`grupo`*), el intérprete de comandos por defecto (opción `--shell` *`nombre`*) y otros datos del usuario con la opción `--gecos "`*`datos`*`"` (los datos se separan con comas y por defecto son: nombre, número de cuarto, teléfono del trabajo y teléfono de la casa). A menos que se especifique `--disabled-login` o `--disabled-password` pedirá clave inmediatamente después de crear la cuenta (`--disabled-login` bloquea la cuenta hasta que se asigne una clave, `--disabled-password` no bloquea la cuenta pero asigna una clave no válida, el usuario podría entrar por ejemplo con RSA ssh).

Las operaciones que **adduser** realiza se configuran en el archivo `/etc/adduser.conf`. Por defecto creará cuentas de usuarios en el directorio `/home` y copiará en los nuevos directorios los

10. Cuando un usuario no ha cambiado su clave dentro del máximo de días especificado, en la siguiente oportunidad que entre a una sesión el sistema solicitará una nueva clave –que debe ser diferente a la anterior.

11. Hay varios programas similares a **adduser**, **addgroup**, **deluser** y **delgroup** se trata de **useradd**, **groupadd**, **userdel** y **groupdel**. Recomendamos emplear los primeros porque son más flexibles (se configuran en los archivos `/etc/adduser.conf` y `/etc/deluser.conf`) y porque permiten seguir las políticas de numeración de grupos y usuarios de Debian

archivos del directorio `/etc/skel` (e.g `.bash_profile`), pondrá como intérprete de comandos por defecto **/usr/bin/bash** y asignará un número de usuario [12] de acuerdo a la política de Debian y asignará un grupo nuevo a cada usuario. Normalmente después para agregar un usuario debe especificar la clave que tendrá, si prefiere crear una cuenta inicialmente sin clave emplee la opción `--disabled-password` (antes de poder emplear la cuenta debe establecer la clave con **passwd**).

chfn [*`login`*]

Para cambiar la información GECOS de un usuario (si no se especifica un login se cambiará la información del usuario que emplee el programa). Como opciones recibe `-f` *`nombre`*, `-r` *`cuarto`*, `-w` *`teléfono_trabajo`*, `-h` *`teléfono casa`* y `-o` *`otros_datos`*. Nuestra sugerencia es emplear en lugar de cuarto el grado y en lugar del teléfono del trabajo la dirección de la casa. Estos datos no deben contener los caracteres ',' ';' ni '='. Los datos que no se especifiquen en la línea de comandos serán solicitados interactivamente. **chfn** también será llamado por **passwd** si se emplea la opción `-f`.

chsh [*`login`*]

Para cambiar el intérprete de comandos de un usuario (si no se especifica un login se cambiará el intérprete del usuario que emplee el programa). Como opción recibe el nombre del intérprete, el cual debe estar listado en el archivo `/etc/shells`. Un usuario que emplee un intérprete de comandos restringido (`/bin/rsh`) no puede cambiar su shell. **chsh** también será llamado por **passwd** si se emplea la opción `-s`.

deluser *`login`*

Para eliminar una cuenta Por defecto no elimina el directorio personal ni la cola de correos, puede indicarse que se borren estos directorios con la opción `--remove-home` y puede indicarse que se busquen y eliminen todos los archivos del usuario (en los demás directorios) con la opción `--remove-all-files`. Con la opción `--backup` creará un archivo comprimido con los datos del usuario en el directorio de trabajo con nombre login.tar.gz. El comportamiento por defecto de este comando puede configurarse en el archivo `/etc/deluser.conf`.

La información sobre usuarios se mantiene en el archivo que todos los usuarios pueden leer: `/etc/passwd`. Cada línea de este archivo tiene información de un usuario separada con el caracter ':'. De cada usuario mantiene:

- `login`
- Clave o un caracter de control. En sistemas Unix las claves antes de ser almacenadas en este archivo (para su posterior comparación) son convertidas a una secuencia de letras y números con un algoritmo (bien DES o bien MD5 [13]), cada vez que un usuario desea ingresar al sistema la clave que teclea se transforma con ese algoritmo y se comparà con la almacenada para dar acceso sólo si son iguales. Por las características de DES y MD5 es muy dificil recuperar la clave original partiendo de la información almacenada en `/etc/passwd` así que una persona con acceso a este archivo no podrá

12. Si no se especifica un número de usuario o grupo a **adduser** o **addgroup**, estos asignarán uno disponible de acuerdo a la política de Debian: en el rango 1000 a 29999 si es un usuario normal o en el rango 100 a 999 si es una cuenta requerida por algún programa (que se indica con la opción `--system`, además al usar --system con **adduser** el intérprete de comandos será `/bin/false`). Las identificaciones entre 0 y 100 son comunes a cualquier sistema Debian y no deben modificarse.
13. Con DES el tamaño máximo de una clave es 8 caracteres, el algoritmo MD5 es más seguro y permite claves de mayor longitud.

conocer facilmente las claves de los usuarios. Sin embargo es mejor emplear el sistema de claves shadow, que mantiene las claves en un archivo aparte que sólo pueda ser leido por el administrador: `/etc/shadow`, las cuentas que empleen este mecanismo tendrán un caracter 'x' en lugar de clave. Tanto en `/etc/passwd` como en `/etc/shadow` en lugar de clave transformada puede aparecer el caracter '*' para indicar que la cuenta tiene clave deshabilitada (opción `--disabled-password` de `adduser`), el usuario podría ingresar con RSA ssh. En lugar de la clave también puede aparecer el caracter '!' para indicar que la cuenta está bloqueada (opción `--disabled-login` de **adduser**), en este caso el usuario no podrá entrar con **login** ni con gdm ni con RSA ssh, pero si con **su** y se ejecutarán procesos del usuarios iniciados por **cron** o **at**.

está bloqueada (el usuario no podrá entrar con login o gdm pero si con rlogin o **su** y se ejecutarán procesos del usuarios iniciados por **cron** o **at**).

- GID, es decir número que identifica al grupo principal del usuario.
- información GECOS [14], los datos se separan entre si con comas.
- directorio personal
- intérprete de comandos

En el archivo `/etc/shadow` [15] hay una línea por cada usuario con los siguientes datos separados por ':':

- Login
- Clave transformada
- Fecha del último cambio de la clave (contada en días a partir del Enero 1 de 1975).
- Días por esperar antes de que la clave pueda ser cambiada.
- Máximo de días antes de exigir un cambio de clave.
- Cantidad de días de preaviso antes de expiración de clave.
- Cantidad de días entre expiración de clave y bloqueo de cuenta.
- Fecha desde la cual la cuenta está deshabilitada (contada en días desde Enero 1 de 1975).
- Campo reservado.

Aunque puede editar manualmente `/etc/passwd` y `/etc/shadow` es recomendable que emplee los programas presentados en esta sección. Los GID y UID reservados en un sistema Debian están en los archivos `/usr/share/base-passwd/group.master` y `/usr/share/base-passwd/passwd.master`, si algún programa o adminsitrador erradamente asigna alguno de estos números puede emplear update-passwd para reasignar los originales.

4.3.1.2. Grupos

El objetivo de los grupos es dar o restringir permisos sobre algunos archivos a ciertos usuarios (ver Archivos y permisos). Por ejemplo un archivo `reporte.txt` que pertenezca al grupo `profesores`, que tenga permiso de lectura para el grupo y no para otros usuarios (si por ejemplo se estableció con `chmod`

14. De acuerdo a la página man de passwd, GECOS es sigla de General Electric Comprehensive Operating System.
15. Para convertir un archivo de claves shadow a passwd y viceversa se usan **pwunconv** y **pwconv** respectivamente. Puede desactivarse el uso de claves shadow con `shadowconfig off` y activarse con `shadowconfig on`

`ug=rw reporte.txt`), podrá ser leído únicamente por el dueño y por usuarios que pertenezcan al grupo `profesores`.

Cada usuario tiene un grupo principal (puede especificarse durante la creación con la opción `--gid` *`GID`* o `--ingroup` *`grupo`* de **adduser**), puede pertenecer a diversos grupos y si conoce la clave de algún grupo con clave puede volverse miembro durante una sesión. Los programas relacionados con grupos son:

adduser *`usuario grupo`*

Para agregar un usuario a un grupo. En Debian algunos dispositivos pertenecen a ciertos grupos, de forma que es indispensable agregarlos como grupos secundarios de los usuarios que los requieran:

`audio`

Permite acceder a dispositivos de sonido.

`lp`

Permite acceder a los puertos locales de impresión.

`floppy`

Para acceder a la(s) unidad(es) de disquette.

`tape`, `cdrom`

Ambos son requeridos para acceder al CDROM.

`dialout`

Para acceder a modems.

`disk`

Cuando se requiere acceder a discos a bajo nivel.

`kmem`

Para acceder de forma privilegiada la memoria.

`tty`

Para acceder de forma privilegiada a la consola.

groups

Un usuario puede ver los grupos a los que pertenece con este programa.

newgrp [*grupo*]

Para cambiarse a un grupo con clave. Si no se específica grupo alguno se cambiará al grupo principal del usuario. Con la opción `-` el ambiente será reinicializado como al inicio de la sesión [16].

passwd -g *grupo*

Para cambiar la clave de un grupo. Para quitar la clave de un grupo se emplea **passwd -g -r *grupo*** —los programas **newgrp** y **sg** no permiten cambiarse a un grupo sin clave.

gpasswd *grupo*

Para administrar grupos con clave, puede ser usado por el administrador del sistema y por el administrador de un grupo con clave. Con la opción `-A` *`login`* el administrador del sistema puede agregar un administrador de grupo a un grupo, con la opción `-M` *`login`* puede retirarse la administración de un grupo a un usuario, con la opción `-r` puede quitarse la clave a un grupo con clave y con la opción `-R` puede inhibir el acceso con **newgrp** a un grupo con clave. Un administrador de grupo puede agregar y eliminar usuarios del grupo con las opciones `-a` *`login`* y `-d` *`login`* respectivamente.

addgroup *nombre*

Permite agregar un grupo con el nombre dado [17] Con la opción `--gid` *`ID`* puede especificarse el número que identificará al grupo, número acorde con el archivo `/etc/adduser.conf` y a las convenciones de Debian: 0-100 estáticos reservados para Debian, 100-999 localizados dinámicamente para grupos del sistema, 1000-29999 para grupos normales (ver descripción del comando **adduser** en la sección anterior).

groupdel *group*

Permite eliminar un grupo. Sólo pueden eliminarse grupos que no sean el grupo principal de algún usuario.

groupmod *grupo*

Permite modificar información de un grupo. Las opciones posible son: `-g` *`GID`* para cambiar el número que identifica al grupo (ver convenciones en descripción de **addgroup**) y `-n` *`nombre`* para cambiar el nombre del grupo. El número que identifica al grupo debe ser único, excepto si se emplea la opción `-o` (para crear grupos alias, aunque el sistema de archivos no necesariamente presentará el alias como grupo dueño).

grpck

Para verificar la información de grupos en `/etc/group` y `/etc/gshadow`. Con la opción `-r` abre estos archivos en modo de sólo lectura.

La información de grupos se consigna en `/etc/groups`, cada línea tiene los siguientes datos de un grupo separados uno de otro con el caracter ':'

- Nombre del grupo

16. **sg** es un comando similar a **newgrp** que además permite especificar un comando por ejecutar con la opción `-c` *`comando`*
17. Un comando casi equivalente a **addgroup** es **groupadd**, sin embargo **addgroup** tiene en cuenta las políticas de Debian y la información de `/etc/adduser.conf`.

- Clave del grupo transformada con DES o MD5. Si el grupo tiene clave shadow en este archivo aparecerá el caracter 'x' y la clave transformada estará en otro archivo (por defecto `/etc/gshadow`.
- GID
- Lista de usuarios del grupo separados con comas.

No es recomendable editar directamente estos archivos, sino más bien emplear los programas presentados en esta sección.

4.3.1.3. Inicio de sesiones en consolas virtuales

Una sesión iniciada desde una consola virtual o desde una conexión remota (via **telnet**, **rsh** o **ssh**) es atendida inicialmente por el programa **getty** (ver Lectura Configuración de una sesión. El mensaje que presenta **getty** se configura en el archivo `/etc/issue` y puede contener algunas secuencias especiales como: `\d` que corresponde a la fecha, `\s` al nombre del sistema operativo, `\l` al número de la consola virtual (línea tty), `\m` al tipo de procesador (e.g i486), `\n` al nombre de la máquina, `\u` cantidad de usuarios conectados. Por ejemplo, si el archivo `/etc/issue` es:

```
Debian \s 2.2 en un \m. \n \u \l
```

getty mostrará un mensaje como:

```
Debian Linux 2.2 en un i586. oxigeno 3 tty1
```

Cuando un usuario teclea su login, **getty** pasa el control al programa **login**. El programa **login** por intermedio de la librería PAM espera la clave del usuario y la válida, cuando el usuario da la clave correcta verifica que el acceso para ese usuario a la hora del ingreso sea posible [18] y entonces inicializa algunas variables de ambiente, muestra algunos mensajes (por defecto la fecha de la última conexión y el contendio del archivo `/etc/motd` [19]) e inicia un intérprete de comandos (el que está configurado para el usuario en `/etc/passwd`). Las acciones que **login** realiza pueden configurarse en los archivos `/etc/login.def` y `/etc/pam.d/login`, las consolas desde las cuales puede ingresar el usuario root se configuran en `/etc/securetty`, otras restricciones de seguridad pueden configurarse en los archivos del directorio `/etc/security` [20].

4.3.1.4. Inicio de sesiones X-Window

Algunas generalidades sobre X-Window y la visión del usuario final se presentaron en el capítulo anterior (ver Lectura Configuración de una sesión), en esta sección presentamos como puede iniciarse un servidor X-Window y algunos aspectos de su configuración.

18. Por defecto todos los usuarios pueden entrar a cualquier hora, puede modificarse esto editando `/etc/security/time.conf`.
19. **motd** abrevia message of day (en español mensaje del día).
20. access.conf login access control table, group.conf para dar acceso a ciertos usuarios a ciertos grupos a los cuales no pertenece, limits.conf para establecer límites en el uso de algunos recursos del sistema (como procesos, memoria, archivos abiertos simultaneamente, tiempo de CPU, número de logins, prioridad de procesos), pam_env.conf donde pueden configurarse variables de ambiente, time.conf donde se configura horarios en el que ciertos usuarios pueden ingresar al sistema

4.3.1.4.1. Servidor X

Un servidor X configurado maneja el modo gráfico de la tarjeta de video, teclado y un apuntador (ratón, tarjeta graficadora) para interactuar con el usuario y atender programas que lo emplean como clientes X-Window. Un cliente X-Window normalmente estará corriendo en la misma máquina donde está el servidor, aunque puede estar en otra máquina ver Servicios de la Intranet. Para iniciar un servidor X se emplea el programa `X` (cuya ruta completa normalmente es `/usr/X11R6/bin/X` y que es un enlace al ejecutable del servidor que haya configurado para su hardware). Cada servidor X tiene una identificación como `natura.micolegio.edu.co:0.0` compuesta por el nombre de la máquina, un número de vista (una vista se refiere a una colección de monitores con un teclado y apuntador común) y un número de pantalla (en caso de que un computador tenga más de un monitor). En un mismo computador pueden iniciarse varios servidores X, cada uno con un númeo de vista diferente. Por ejemplo para iniciar como vista 0 un servidor X en la consola virtual 9 (pasa a ella con **Ctrl-Alt-F9**, ver Sistema de usuarios y manejo de clave):

```
/usr/X11R6/bin/X :0 vt9
```

Simultáneamente podría iniciarse como vista 1 en la consola virtual 3 otro con:

```
/usr/X11R6/bin/X :1 vt3
```

Por defecto si no se especifica una consola virtual por usar, X buscara y empleará la primera libre.

4.3.1.4.2. Administrador de vistas xdm

Se configura en `/etc/X11/xdm/xdm-config` y `/etc/X11/xdm/xdm.options`, este administrador de vistas puede administrar uno o más servidores en una o varias máquinas (configurados en `/usr/lib/X11/xdm/XServer`), y permite usar terminales X con el protocolo XDMCP.

En cada servidor que maneja ejecuta el script `/usr/lib/X11/xdm/XSetup` y después permite el inicio de sesiones presentando el widget Xlogin que pide login y clave a cada usuario y que usa los los recursos de X-Window definidos en `/usr/lib/X11/xdm/Xresources` (en particular la imagen presentada).

Cuando un usuario se autentica se ejecutan los scripts: `/usr/lib/X11/xdm/Xstartup` como root (que por defecto sólo agrega la autenticación a la bitácora `utmp`), después `/usr/lib/X11/xdm/Xsession` como usuario. Este último ejecuta **/etc/X11/Xsession** el cual ejecuta todos los scripts del directorio `/etc/X11/xsession.d` que por defecto mezclan recursos de X-Window (tanto del sistema como de usuario) y ejecutan el script `~/.xsession` el cual debe ser configurado por el usuario (ver Lectura Configuración de una sesión). Algunas variables que afectan la ejecución de los scripts de `/etc/X11/Xsession.d` se configuran en `/etc/X11/Xsession.options`, puede consultarse ayuda sobre estas variables con **man Xsession.options**.

Cuando la ejecución de `~/.xsession` concluye, xdm cierra la sesión con el script `/usg/lib/X11/xdm/Xreset`.

4.3.1.4.3. xinit o startx

xinit es un programa para iniciar desde la línea de comandos una sesión de X-Window teniendo en cuenta la configuración del usuario. Para esto inicia el servidor X configurado en `~/.xserverrc` o en `/etc/X11/xinit/xserverrc` (que normalmente es `/usr/X11R6/bin/X`) y después ejecuta el archivo de configuración `~/.xinitrc` o en su defecto `~/.xsession` o a falta de los anteriores `/usr/lib/X11/xinit/xinitrc`. La sintaxis de `~/.xinitrc` es la misma de `~/.xsession` —sólo que el primero hereda las variables de ambiente del interprete de comandos (la sintaxis de `~/.xsession` puede consultarse en ver Lectura Configuración de una sesión).

startx tiene exactamente el mismo propósito (de hecho ejecuta `xinit`).

Estos comandos son especialmente útiles durante la configuración de X-Window, bien cuando no se ha logrado configurar para los dispositivos de su computador (y en ese caso **xdm** no puede iniciar), o bien para probar cambios más rapidamente.

Eventualmente mientras **xdm** esté operando usted puede querer hacer cambios a la configuración de X-Window, termine en ese caso **xdm** (con **/etc/init.d/xdm stop** o buscando y matando el proceso de **xdm**), términe el servidor X que esté corriendo, haga los cambios necesarios y después ejecute **startx**.

4.3.1.4.4. Administrador de vistas gdm y Gnome

El administrador de vistas gdm es similar a xdm aunque ofrece mejoras en la presentación.

Se configura en el archivo `gdm.conf` y en scripts cuyo nombre sea el número de vista (e.g. `:0`) o `Default` en los directorios:

`/etc/gdm/Init`

: Cuando gdm inicia.

`/ete/gdm/PreSession`

: Cada vez que un usuario abre una sesión.

`/etc/gdm/PostSession`

: Cada vez que un usuario cierra una sesión.

En el directorio `/etc/gdm/Sessions` hay scripts que serán ejecutados de acuerdo a la sesión que el usuario en el menú tipo de sesión durante la autenticación.

El script que inicia la sesión Gnome, por defecto carga algunas variables de ambiente de `/etc/login.defs`, mezcla recursos del sistema `/etc/X11/Xresources` y del usuario (`~/.Xresources`, activa configuración de teclado especializada del sistema y del usuario (`/etc/X11/Xmodmap` y `~/.Xmodmap`), ejecuta el archivo de configuración de Gnome del usuario `~/.gnomerc` e inicia el administrador de sesiones GNOME: `gnome-session`.

4.3.1.5. Variables de ambiente en consolas virtuales y X-Window

La mayoría de intérpretes de comandos, incluyendo bash, leen información de configuración primero del archivo `/etc/profile` y después de archivos de configuración del usuario (ver Lectura Configuración de una sesión).

Sin embargo si emplea un administrador de vistas como xdm, gdm o kdm estos programas por defecto no leerán ese archivo, así que para establecer variables antes de que un administrador de escritorio o un manejador de ventanas inicie debe emplearse un archivo de configuración propio del administrador de vistas. En el caso de gdm el valor de la variable PATH puede establecerse en `/etc/X11/gdm/gdm.conf` (línea DefaultPath) y otras variables pueden establecerse en el archivo que controla el tipo de sesión (con gdm el usuario puede escoger el tipo de sesión al que desea entrar e.g Xsession, Debian, Gnome). Los archivos que controlan las sesiones son scripts localizados en el directorio `/etc/X11/gdm/Sessions` –son scripts normalmente cortos que puede revisar y editar para configurar el inicio de sesiones gráficas.

La solución más general y rápida para establecer variables de ambiente es configurarlas en el archivo `/etc/security/pam_env.conf` que es leido por PAM –que a su vez es usado tanto por **login** como por administradores de displays como xdm y gdm. Si desea configurar PATH tenga en cuenta modificar `/etc/profile` y el archivo de configuración de su administrador de displays para que no la sobreescriban.

4.3.2. Lectura recomendadas: Administración de usuarios

- Otra presentación del sistema de usuarios está disponible por ejemplo en la sección 4.6 de "*Linux Installation and Getting Started*" http://www.linuxdoc.org/LDP/gs/node6.html#secmanageusers
- Páginas del manual del sistema de cada una de las herramientas mencionadas en esta sección (en la primera guía se presentan direcciones en Internet donde puede consultar las páginas **man** (ver Búsqueda y consulta de documentación).
- Puede consultar las políticas sobre numeración de grupos y usuarios en la sección 3.2 de "*Debian Policy Manual*". En un sistema Debian están disponibles en el directorio `/usr/doc/debian-policy`, en Internet el documento está en: http://www.debian.org/doc/debian-policy/
- Puede consultar más documentación sobre el sistema de claves en **info libc** el manual de libc o en Internet en http://www.gnu.org/manual/glibc-2.2.3/html_chapter/libc_29.html#SEC602.
- Si instala en su sistema el paquete libpam-doc (ver Administración de programas), puede consultar más sobre PAM, su configuración y los módulos disponibles en el directorio `/usr/share/doc/libpam-doc`.
- Puede consultar información sobre X-Window y el administrador de vistas x en las páginas man: `X`, `xdm`, `xdm.options`.
- Una buena referencia de X-Window (incluyendo xdm y XDMCP) es: *X-Window System Administrator's Guide*, 1992. Linda Mui y Eric Pearce. O'Reilly & Associates, Inc.
- En Internet puede consultar sobre XDMCP en: http://www.xs4all.nl/~zweije/xauth-9.html
- Puede consultar más sobre GDM en "*GNOME Display Manager Reference Manual*", disponible en su computador en `file:/usr/share/gnome/help/gdm/C/gdm.html` o en Internet en http://www.oswg.org/oswg-nightly/oswg/en_US.ISO_8859-1/articles/gdm-reference/gdm-reference.html

4.3.3. Ejercicios: Administración de usuarios

1. Los números UID y GID del usuario root son fijos, investigue en su sistema cuales son.

2. En el directorio `tarea1` se quiere que todos los miembros del grupo estudiantes puedan escribir, pero que un miembro de ese grupo no pueda borrar o renombrar archivos de otros, ¿cómo puede lograrse? si además se quiere que no puedan ver o modificar el contenido de archivos de otros miembros del mismo grupo que se requiere?

3. Para transformar una clave con el algoritmo DES puede emplear el siguiente script escrito en lenguaje Perl:

```
#!/usr/bin/perl
$sal=join ”, (’.’, ’/’, 0..9, ’A’..’Z’,’a’..’z’)[rand 64, rand 64];
print crypt($ARGV[0],$sal);
print "\n";
```

Si el nombre del script es **enc.pl** y le da permiso de ejecución, para transformar la clave "vida" bastaría ejecutar **enc.pl vida**.

Emplee este script para transformar una clave, después edite `/etc/passwd` o `/etc/shadow` agregue la nueva clave transformada a una cuenta de prueba y finalmente compruebe que la nueva clave funciona entrando a la cuenta de prueba. Nota: Si desea experimentar con MD5 en lugar de DES debe cambiar la "sal" [21], remplazando la línea con la función **crypt** por **print crypt($ARGV[0],"\$1\$$sal");**

4. Cree un usuario en el grupo `users` y agréguelo a grupos que le permitan acceder a la impresora local, a dispositivos de audio y a la unidad de disquette. Después pase a la cuenta del nuevo usuario y compruebe que pueda usar disquettes. Finalmente elimine el usuario creado.

5. Nuestra plataforma de referencia (ver Plataforma de referencia) sugiere 3 grupos básicos: profesores, estudiantes y administración. Considere ventajas y desventajas de esta política (o si lo prefiere de otra ideada por usted de acuerdo a la infomación que se maneja en la red). ¿Qué usuarios deberían tener más de un grupo? ¿qué grupos podrían tener clave, quienes administrarían tales grupos y quienes serían los usuarios?

6. Cree los grupos que decidió en el punto anterior y usuarios de prueba (de forma que al menos un usuario de prueba esté en dos grupos).

7. ¿Qué cambios puede realizar en `/etc/adduser.conf` para facilitar la implementación de la política descrita en los ejercicios anteriores?

8. Piense una máscara de permisos apropiada para todos los usuarios. Describa como la aplica empleando **umask** y los cambios que debe realizar para establecer tal mascara por defecto para todos los usuarios.

9. Haga los cambios apropiados en su sistema, para que todo directorio creado para nuevos usuarios tenga un archivo `ayuda.txt` donde los usuarios podrían escribir sus propias notas para emplear bien el sistema, y un mecanismo que le recuerde a los usuarios la existencia de tal archivo. Inicialmente ese archivo podía tener un mensaje de bienvenida y/o instrucciones para comenzar a usar bien el sistema.

10. Haga una lista de chequeo de detalles que deba tener en cuenta para prevenir que alguien entre a cuentas que no le pertenecen (especialmente como evitar que alguién pueda entrar a la cuenta root).

11. Opcional: Configure gdm para que en vez de presentar el logo típico presente el logo de su institución educativa.

---

21. Tanto MD5 como DES transforman una clave empleando un número, este número (llamado informalmente "sal") produce una transformación diferente y se adjunta al comienzo de la clave transformada. Es útil para hacer más dificil a un intruso descubrir claves (porque si intenta descubrir claves probando con algunas fijas, transformandolas y comparando con `/etc/passwd` debe usar cada una de las sales de cada usuario en sus intentos.

4.4. Administración de archivos

Indicadores de Logros

- Conoce la organización de la jerarquía de directorios de Linux.
- Administra sistemas de archivos.
- Administra el sistema de archivos Ext2.

4.4.1. Lectura: Administración de archivos

En Unix se emplean archivos o directorios para mantener información en medios de almacenamiento (e.g discos duros, disquettes, CD-ROM) pero también para abstraer dispositivos de hardware y mantener canales de comunicación con programas. En una red pueden emplearse directorios o archivos que estén en otro computador (ver Servicio NFS).

Para leer o escribir en un medio de almacenamiento es necesario montarlo antes en la jerarquía de directorios (see Disquetes y CDROM). Aunque antes de poderlo montar ya debe estar formateado a bajo nivel y debe contar con un sistema de archivos. Formatear un medio de almacenamiento significa prepararlo para que el hardware pueda leerlo y escribirlo, al dotarlo de un sistema de archivos se prepara para que el sistema operativo pueda organizar, leer y escribir datos.

4.4.1.1. Ubicación de archivos y directorios

Los directorios y archivos de un sistema Debian tiene una estructura estándar [22], algunos de los directorios son:

`/boot`

Mantiene el kernel e información indispensable para el arranque del sistema.

`/bin`

Mantiene unos pocos programas que estarán disponibles incluso en los modos de ejecución más restringidos (e.g. **bash**, **cat**, **ls**, **login**, **ps**).

`/sbin`

Mantiene programas disponibles sólo para el administrador incluso en los modos de ejecución más restringidos (e.g **fsck**, **getty**, **halt**).

`/usr`

Programas accequibles a usuarios finales y datos de estos programas que no requieren ser modificados (datos de sólo lectura). Algunos de sus subdirectorios son:

`bin`, `games`

Programas disponibles para todos los usuarios.

22. FHS (*Filesystem Hierarchy Standard*) es un estándar que establece "requerimientos y guías para ubicar archivos y directorios en sistemas estilo UNIX".

`sbin`

: Herramientas para el administrador.

`dict`

: Diccionarios –usados por ejemplo por ispell y look.

`share`

: Datos de diversos programas, entre sus subdirectorios están: `doc`, `man` y `info` que contienen documentación; `locale`, `i18n` que mantiene información sobre internacionalización (locale); `zoneinfo` que mantiene información sobre zonas horarias.

`doc`, `info`, `man`

: Enlaces o complementos a la documentación disponible en `/usr/share/doc`, `/usr/share/info` y `/usr/share/man`.

`lib`

: Librerías de funciones comunes a varios programas (algunas como libc indispensables para todos).

`include`, `src`

: Encabezados de algunos componentes del sistema y fuentes de algunos programas.

`X11R6`

: Programas y datos de X-Window.

`local`

: Otros programas y datos compilados por el administrador i.e no provienen de paquetes Debian. En algunos sistemas otros programas se ubican en `/opt`.

`/floppy, /cdrom, /mnt`

: Directorios para montar disquettes, CD-ROMs y otros sistemas de archivos o dispositivos.

`/proc`

: Diversos canales de comunicación con algunos programas (por ejemplo con el kernel).

`/lib`

: Librerías indispensables y módulos (especialmente requeridas durante el arranque del sistema).

`/etc`

: Archivos de configuración de diversos programas.

`/dev`

: Abstracciones a los dispositivos conectados (o que podrían conectarse) al computador.

`/home`

Mantiene información de los usuarios del sistema.

`/root`

Mantiene información del administrador del sistema.

`/tmp`

Archivos temporales creados por algunos programas, que serán borrados por el sistema operativo durante el arranque.

`/var`

En este directorio los programas que lo requieran pueden mantener archivos que deban modificarse frecuentemente. Algunos de sus subdirectorios son: `mail` donde se mantienen colas de correo de cada usuario, `log` que mantiene bitácoras, `run` con archivos que mantienen los números de los procesos iniciados al arrancar el sistema, `spool` colas de impresión y de otros programas, `tmp` archivos temporales (más persistente que `/tmp`).

`/proc`

Este directorio es "virtual", no está presente en el disco, es creado por el sistema para intercambiar información con más facilidad.

El rigor de esta jerarquía hace que memorizandola sea fácil emplear diversos sistemas Unix, aunque puede no ser la más apropiada en algunas circunstancias. Para simular jerarquías diferentes cada proceso iniciado por el usuario root puede tener un directorio raíz diferente al directorio raíz real del sistema. Para establecerlo se emplea el programa **chroot**, que recibe como primer parámetro el nuevo directorio raíz y como segundo parámetro el programa por ejecutar. La ruta del programa se especifica con respecto al nuevo directorio raíz y para que el programa pueda ejecutarse, deben ubicarse las ibrerías que este requiera en la "nueva" jerarquía (por ejemplo en el nuevo directorio `/lib`).

Esta jerarquía de directorios en Linux no necesariamente está en un disco, sino sólo en memoria RAM, por eso pueden tenerse directorios como `/proc` y pueden montarse diversos sistemas de archivos sobre la misma jerarquía como se presenta en la siguiente sección.

4.4.1.2. Montaje y desmontaje de sistemas de archivos

Un sistema de archivos es información almacenada en un dispositivo de almacenamiento organizada siguiendo una convención, la cual permite al sistema operativo manejar la información cuando el usuario lo requiere, es decir, leer, escribir, consultar directorios, etc. Hay diversos tipos (o convenciones) de sistemas de archivos, Linux cuenta con controladores para varios tipos creados específicamente para Linux (e.g ext2, ReiserFS) así como controladores para tipos de sistemas de archivos creados para otros sistemas operativos (e.g affs, hpfs, ntfs, msdos). Los tipos que pueden emplearse con Linux [23] están

23. Los tipos soportados por el kernel 2.2.18, propios de otros sistemas operativos son: minix - Minix, affs Amiga Fast File System, hpfs - OS/2, ntfs - Windows NT/2000, msdos - Dos y Windows 3.1, vfat - Windows 95/98/ME algunas versiones de NT, fat_cvf - volumenes FAT comprimidos (Stacker, DoublSpace), adfs - Acorn Advanced, qnx4 - QNX, sysv - SystemV/Xenix/Coherent, ufs - BSD4.2/FreeBSD/NetBSD/OpenBSD/Solaris/NextStep, efs - Irix de Silicon Graphics, hfs - Apple. Los tipos creados con o para Linux son: ext2; umsdos - extensión a msdos para soportar nombres largos, permisos y propietarios, enlaces y archivos especiales; reiserfs, ext3, proc, usbdev. Los tipos de sistemas de archivos distribuidos son: nfs, coda, smb - protocolo SMB de Windows FWG/95/NT, shmem, ncp - Novell NetWare filesystem. Para ciertos dispositivos hay tipos especiales: iso9660 para leer

incluidos en el kernel o son módulos en el subdirectorio `fs` del directorio con módulos de su sistema (es decir, el directorio de módulos es de la forma `/lib/modules/2.2.x/fs/` remplazando x por la versión de su kernel).

Los que puede usar dependen de la forma como se haya compilado el kernel y de los módulos que estén cargados. Puede examinar los tipos que soporta su sistema viendo el contenido de `/proc/filesystems` y puede emplear otros tipos de sistemas de archivos cargando los módulos apropiados —mientras sean soportados (ver Lectura Inicialización del sistema).

Aunque toda la estructura de directorios presentada en la sección anterior podría estar en un sólo sistema de archivos, es posible distribuirla en varios. Para poder emplear un sistema de archivos primero debe montarse en algún directorio con el comando **mount** o debe configurarse el sistema para que lo monte automáticamente durante el arranque (el primer directorio que debe montarse es la raíz `/`). Para desmontar un sistema de archivos de un directorio se emplea el comando **umount**.

El programa **mount** sin argumentos presenta los sistemas de archivos que están montados [24], de cada uno presenta el dispositivo en el que reside, el directorio en el que está montado, el tipo de sistema de archivos y algunos parámetros del montaje, por ejemplo:

```
/dev/hda1 on / type ext2 (rw,errors=remount-ro,errors=remount-ro)
proc on /proc type proc (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/hda2 on /var type ext2 (rw)
/dev/hda3 on /home type ext2 (rw)
```

En el ejemplo se muestran 5 sistemas de archivos montados en los directorios `/`, `/proc`, `/dev/pts`, `/var` y `/home`. 3 de ellos son de tipo ext2 y residen en tres particiones de un disco duro (`/dev/hda1`, `/dev/hda2` y `/dev/hda3` ver Dispositivos y Linux), los otros dos son virtuales, es decir, no residen en dispositivo de almacenamiento alguno (son creados por el kernel). Las opciones de montaje de todos indican que pueden leerse y escribirse `rw`, el sistema montado en la raíz en caso de fallas será montado con permiso de sólo lectura (`errors=remount-ro`) y el sistema montado en `/dev/pts` tendrá todos sus archivos como pertenencientes al grupo 5 (tty) y permisos de escritura y lectura para el usuario y sólo de escritura para el grupo (modo 620).

Para montar un sistema de archivos se emplea:

```
mount -t tipo -o opciones  dispositivo directorio
```

Donde *`dispositivo`* es el nombre del dispositivo que contiene el sistema de archivos (e.g `/dev/hda1` o `/dev/fl0`), *`tipo`* es el tipo del sistema de archivos [25] y *`directorio`* es la ruta donde se quiere acceder la raíz del sistema montado (si el directorio donde se monta contiene datos, estos serán escondidos mientras el nuevo sistema de archivos permanezca montado).

La información sobre un sistema de archivos, su directorio de montaje y opciones pueden especificarse en el archivo `/etc/fstab`. De hacerlo puede emplear

```
mount directorio
```

CD-ROMs, romfs sistema de archivos de sólo lectura para ROM o RAM para mantener el kernel y módulos.

24. También puede examinar los sistemas de archivos montados en `/etc/mtab` o en `/proc/mounts`.

25. El tipo especial `auto` indica al kernel que intente determinar el sistema de archivos automáticamente. Intentará en el ordén especificado en el archivo `/etc/filesystems` o en su defecto `/proc/filesystems`. Esta opción es muy útil por ejemplo para disquettes.

para montar el sistema de archivos en el directorio especificado (el resto de información debe estar en `/etc/fstab`). Además, por defecto, los sistemas de archivos especificados en ese archivo serán montados automáticamente durante el arranque del sistema. Un ejemplo complementario al ejemplo anterior es:

```
/dev/hda1   /        ext2    defaults,errors=remount-ro   0  1
/dev/hda4   none     swap    sw                           0  0
proc        /proc    proc    defaults                     0  0
/dev/fd0    /floppy  auto    defaults,user,noauto         0  0
/dev/cdrom  /cdrom   iso9660 defaults,ro,user,noauto      0  0
/dev/hda2   /var     ext2    rw                           0  2
/dev/hda3   /home    ext2    rw                           0  2
```

Cada línea especifica un sistema de archivos con: dispositivo, directorio donde se debe montar, tipo del sistema de archivos, opciones de montaje separadas por comas, la frecuencia de volcado (típicamente 0) y un númeo para controlar el chequeo durante el arranque con **fsck**. Además de los tipos soportados por su sistema puede emplear **swap** para especificar particiones que servirán como zona de intercambio [26]. Algunas opciones (comunes a todo sistema de archivos) son: `ro` para montar en modo de sólo lectura, `user` que indica que un usuario puede montar el sistema de archivos (por defecto sólo el administrador puede hacerlo), `rw` para montar en modo lectura y escritura, `ro` modo de sólo lectura, `noauto` para indicar que el sistema no debe montarse durante el arranque —note que los directorios `/floppy` y `cdrom` no son montados automáticamente durante el arranque (ver Disquetes y CDROM). El número que controla el chequeo con `fsck` indica el orden en el cual hacer chequeos, 0 indica que no se debe chequear el sistema, otros números positivos especifian una prioridad (los menores son más prioritarios y los números con el mismo número de ser posible son chequeados en paralelo). Es conveniente emplear 1 con el sistema de archivos montado en la raíz y 2 para los demás que requieran chequeo.

4.4.1.3. Sistemas de archivos en Linux

Aunque cada tipo de sistema de archivos tiene características propias, Linux puede manejar varios representándolos en un módelo de sistema de archivos propio del kernel (virtual file system). Por ejemplo los tipos `vfat` y `msdos` no permiten manejar permisos ni propietarios, pero al montarlos el kernel suple la carencia con valores por defecto [27].

El sistema de archivos del kernel se organiza a partir del árbol de directorios, en este árbol se encuentran nombres de archivos y directorios. Junto con cada nombre hay un número que identifica de forma única la información y que permite ubicarla en algún dispositivo de almacenamiento —estos números pueden examinarse con **ls -i**. Por este diseño pueden montarse y desmontarse diversos sistemas de archivos sobre el mismo arbol de directorios, cada proceso puede tener su propio directorio raíz (**chroot**) y puede haber directorios y archivos que no corresponden a información almacenada en dispositivo alguno e.g el directorio `/proc`.

Cada entrada del árbol de directorios es de uno de los siguientes tipos:

26. Zona de intercambio o memoria swap es espacio en disco que puede emplearse como si fuera memoria RAM en caso de que esta última no sea suficiente (aunque es mucho más lento el acceso a un disco que a RAM). Las zonas de intercambio no se montan en el sistema de archivos, se especifican en `/etc/fstab` con tipo **swap**. La opción `sw` indica al kernel que debe emplear esa zona desde el arranque (posteriormente puede desmontarse o volverse a montar con los programas **swapon** y **swapoff**). Las zonas de intercambio no requieren ser volcadas o chequeadas con **fsck**.
27. El propietario por defecto en sistemas msdos o vfat puede establecerse con la opción `uid=`*`UID`* de `mount`, el grupo por defecto con la opción `gui=`*`GID`*.

regular

Se trata de infomación que está almacenada fisicamente en algún dipositivo montado, es decir programas o datos.

enlace simbólico

Vinculo al nombre de otro archivo en el arbol de directorios (este es un enlace suave, a diferencia de un enlace duro que referencia el nodo-i). Se crean con el programa **ln** y la opción `-s`. La desventaja de los enlaces suaves es que hacen el acceso a la información un poco más ineficiente que los enlaces duros, entre sus ventajas están: si se borra el archivo enlazado el enlace simbólico ya no funciona, pueden hacerse enlaces simbólicos a directorios (no pueden hacerse enlaces duros a directorios).

directorio

Subdirectorio. Se manejan con **mkdir**, **rm -f**. Cada proceso tiene un directorio de trabajo que puede administrarse con **pwd** y **cd**.

Dispositivo por caracteres

No es información almacenada en dispositivo alguno, representa un dispositivo físico que se controla enviando de a un caracter (e.g una línea serial, un modem, una terminal). Se crean con

```
mknod [-m modo] nombre
 num_mayor num_menor
```

Donde el modo puede especificarse con números o letras al igual que con **chmod** (ver Lectura Archivos y permisos), *`num_mayor`* y *`num_menor`* son números que en Linux identifican de forma única el dispositivo [28]. Algunos números de dispositivos se presenta en el siguiente capítulo, la lista completa está en las fuentes del kernel en `include/linux/major.h` y están documentados en `Documentation/devices.txt` o en Internet la documentación más reciente (que corresponde al kernel más nuevo) en: http://www.kernel.org/pub/linux/docs/device-list/devices.txt. Por ejemplo allí se documenta que el dispositivo de caracteres con número mayor 4 es un dispositivo `tty` (por ejemplo una de las consolas virtuales), el número menor indica de que consola virtual se trata (si es un entero entre 1 y 63), para crear una entrada en el directorio de trabajo que se refiera a la primera consola virtual con el nombre cons1:

```
mknod cons1 c 4 1
```

Dispositivo por bloques

Representa un dispositivo al que se le pueden enviar bloques de información (e.g tarjeta de video, disco duro). Para crearlo se usa **mknod** de la forma como se presentó para dispositivos de caracteres pero con `b` después del nombre en lugar de `c`.

Pipes y sockets

No es información almacenada en dispositivo alguno, son canales de comunicación entre programas. Desde el intérprete de comandos pueden crearse pipes con **mkfifo** o con **mknod** seguido del nombre del pipe y la letra ’p’.

28. En lugar de **mknod** también puede emplearse MAKEDEV que recibe nombres de dispositivos conocidos por Linux y crea los archivos apropiados en `/dev`.

Además de los permisos presentados en la sección para usuarios (ver Archivos y permisos) en Linux pueden especificarse los siguientes:

- *setuid bit*: empleado con archivos ejecutables cambia la identificación del usuario por la del dueño del archivo. Con **chmod** se modifica empleando `u` y `s`. por ejemplo:

  ```
  chmod u+s cvs
  ```

  indica que cuando el programa **cvs** se ejecute, debe cambiarse la identificación por la del dueño de ese programa.

  Este permiso puede ser útil para que usuarios normales ejecuten programas que requieren ser ejecutados por `root` y por lo mismo debe usarse con precaución.

- *setgid bit*: tambíen empleado con archivos ejecutables, indica que cuando se ejecute el programa debe cambiarse al grupo del archivo. Se establece con `g` y `s`.
- *sticky bit*: se establece con `t`, no tiene efecto sobre archivos [29], con un directorio evita que un usuario que no sea el dueño del directorio pero que tenga permiso de escritura, pueda borrar o renombrar archivos que no le pertenecen.
- Ejecución condicionada: se establece con `X` en el caso de directorios es equivalente a `x`, en el caso de archivos también es equivalente a `x` sólo si el archivo al que se modifica el permiso ya tenía permiso de ejecución para algún usuario. Es útil por ejemplo:

  ```
  chmod a+X *
  ```

  que da permiso de ejecución a todos los usuarios a los archivos ejecutables y a los directorios.

El administrador puede cambiar los permisos de todos los archivos del sistema y puede fijar una política inicial para la mascara de permisos que todos los usuarios tendrán. Además de **chmod** [30] se emplean los siguientes programas para realizar estas labores:

chgrp *`grupo archivos`*

Cambia el grupo de el o los archivos especificados. Por ejemplo

```
chgrp profesores notas* quiz*
```

cambiará todos los archivos con prefijos "notas" y "quiz" al grupo profesores. Cómo opción puede recibir `-R` para aplicar el cambio recursivamente en un directorio (es decir a subdirectorios también). Este programa puede ser usado por cualquier usuario para cambiar el grupo de sus archivos a otro grupo al que el mismo usuario pertenezca.

29. En otras versiones de Unix con un programa hace que sea mantenido en memoria swap para que pueda cargarse más rápido.
30. El programa **chmod** también puede recibir la especificación de los permisos como un número octal (de 1 a 4 digitos): el primer dígito de izquierda a derecha indica permisos de otros; el segundo, en caso de haberlo, permisos del grupo; el tercero, si lo hay, permisos del usuario y el cuarto permisos de los programas. Para los digitos de usuario, grupo y otros un 0 indica ningún permiso, un 1 indica permiso de ejecución, 2 permiso de escritura y 4 permiso de lecturar; las posibles combinaciones se obtienen sumando estos valores (por ejemplo la máxima suma es 7 que concede los tres permisos). En el caso del digito de permisos especiales 1 es el bit sticky, 2 setgid y 4 setuid. Por ejemplo **chmod 62** da permiso de lectura y escritura al grupo y sólo de lectura a otros. **chmod 4110 prog** marca como ejecutable el archivo prog para usuario y grupo y pone el bit setuid.

chown *usuario*[:*grupo*] *archivos*

Cambia el usuario (y opcionalmente el grupo) de los archivos especificados. Al igual que chgrp con la opción `-R` efectua el cambio recursivamente en subdirectorios. Por ejemplo

```
chown -R paz:administra nomina/
```

Pondrá al usuario paz como dueño de todos los archivos y subdirectorios del directorio `nomina`, todos quedarán con el grupo administra. También puede usarse para cambiar el grupo de un archivo (si no se especifica el nombre del usuario).

umask

Este es un comando del intérprete de comandos [31] , para establecer permisos por defecto en archivos nuevos. En bash pueden especificarse los permisos como con chmod por ejemplo:

```
umask u=rw,g=,o=
```

hará que por defecto grupo y otros usuarios no tengan permiso de lectura ni escritura, sólo el propietario del archivo los tiene [32] . Los permisos de un archivo recién creado se establecen de acuerdo al programa que lo crea y a la máscara. Puede esperar que los programas no restrinjan seguridad y creen nuevos archivos y directorios con permisos de lectura y escritura para propietario, grupo y usuarios; este comportamiento hace que **umask** sea una buena alternativa.

4.4.1.4. Sistema de archivos ext2

ext2 es un sistema de archivos desarrollado para y junto con Linux. Permite todas las características que el sistema de archivos virtual de Linux soporta: directorios y archivos como inodos (con permisos, dueños y tipos especiales e.g dispositivos), enlaces duros y suaves. Entre sus limites se cuentan: nombres de archivos de máximo 255 caracteres, archivos de máximo 2GB, sistema de archivos de máximo 4TB. Aunque ext2 procura evitar fragmentación, en el momento de este escrito no hay una herramienta para defragmentar [33].

Además de los atributos de archivo presentados en la sección anterior, un sistema ext2 tiene otros que pueden administrarse con los siguientes programas:

lsattr *archivos*

Lista los atributos especiales de los archivos especificados. Entre sus opciones `-R` lista recursivamente atributos de subdirectorios y su contenido, `-a` lista también archivos de configuración.

chattr *modo archivos*

Permite modificar los atributos especiales de un sistema de archivos ext2. Como *modo* se especifican los cambios a los permisos, cada permiso se especifica con un caracter: `A` - no actualizar tiempos de acceso, `S` - actualización sincrona (es decir escribe directamente sin emplear caches), `a` -

31. **umask** es un comando al menos de bash, csh y tcsh.
32. Para especificar la mascara de **umask** también puede emplearse un número en octal. El número indica los permisos que por defecto NO deben concederse. El ejemplo análogo al presentado en esta sección es **umask 0177**.
33. Aún cuando en el sitio de desarrollo hay planes para desarrollar una http://e2fsprogs.sourceforge.net/ext2.html.

permite abrir el archivo sólo para agregar datos, `c` - el archivo es mantenido comprimido en disco (sus datos son descomprimidos automáticamente por el kernel), `d` - al emplear el program `dump` el archivo es copiado, `i` - el archivo no puede ser modificado o renombrado, `j` - llevar bitácora de datos, `s` - el archivo debe sobreescribirse con ceros si es eliminado, `u` - en caso de ser borrado por el usuario una copia es salvada para su posible recuperación. Los permisos se especifican como primer parámetro de **chattr**, precediendo uno o más atributos con '+' para activar, '-' para desactivar o '=' para dejar igual.

En el disco la información se organiza en bloques (de 1024, 2048 o 4096 bytes), estos bloques se juntan en grupos (cada uno de máximo 32768 bloques), algunos de los bloques se reservan para el administrador (para que pueda operar si los usuarios llegan a llenar el sistema de archivos). Cada grupo tiene algunos bloques de control y otros de datos, entre los bloques de control está una copia del superbloque (bloque que describe el sistema de archivos completo) y una tabla de los nodos-i cuyos datos están en el bloque. Un nodo-i mantiene información sobre un archivo es decir: atributos (modo, uid, gid, fecha de acceso, fecha de creación, fecha de modificación, fecha de eliminación), cantidad de enlaces y lista de los bloques del grupo en donde se encuentra la información del archivo.

Algunas herramientas para emplear sistemas de archivos ext2 son:

mke2fs *`dispositivo`*

También llamado `mkfs.ext2`, permite crear un sistema ext2 en un dispositivo de almacenamiento. Recibe el nombre del dispositivo, y de no recibir opciones prepara el sistema de archivos calculando los parámetros a partir del tamaño disponible. Como opciones pueden especificarse: `-c` para chequear bloques con errores antes de crear sistema de archivos, opcionalmente pueden pasarse los bloques erróneos —reportados por el programa `badblocks`— en un archivo con `-l` *`arc_blerr`*); `-L` *`etiqueta`* para poner una etiqueta; puede especificar el número de nodos-i bien con `-N` *`num_nodos`* o con `-i` *`bytes_por_inodo`* —tenga en cuenta que no puede cambiarse la cantidad especificada después de que el sistema de archivos ha sido creado; el tamaño en bytes de cada bloque puede ser 1024, 2048 o 4096 y se especifica con la opción `-b` *`tamaño`*; el porcentaje de bloques para el administrador puede especificarse con `-m porcentaje` —por defecto 5%.

debugfs *`dispositivo`*

Permite examinar y modificar a bajo nivel un sistema de archivos ext2.

badblocks *`dispositivo`*

Localiza bloques con errores en un dispositivo y genera un reporte en un archivo.

e2fsck *`dispositivo`*

Verifica un sistema de archivos ext2. Este programa es llamado por **fsck** cuando se chequea un sistema con ext2.

4.4.1.5. Otros sistemas de archivos

Además de poder montar diversos tipos de sistemas de archivos hay paquetes que permiten formatear, revisar o hacer otras operaciones con algunos.

`dosfstools`

Este paquete incluye herramientas para sistemas de archivos FAT12, FAT16 y FAT32, i.e los programas `mkfs.msdos` (también llamado `mkfs.vfat`) y `fsck.vfat` (también llamado `fsck.msdos` o `dosfsck`). El primero permite crear sistemas de archivos para MS-DOS y Windows, por ejemplo para formatear la partición `/dev/hdb1` con sistema FAT32:

```
mkfs.vfat -F 32 /dev/hdb1
```

`fsck.msdos` permite verificar una partición con sistema de archivos para MS-DOS o Windows, por ejemplo para verificar `/dev/hdb1` y reparar errores automáticamente:

```
fsck.vfat -a /dev/hdb1
```

4.4.2. Lecturas recomendadas: Administración de archivos

- FHS. En Debian disponible en el paquete debian-policy. Cuando se instala queda en `/usr/doc/debian-policy/fhs`.
- The Linux System Administrators' Guide: Version 0.6.2. Chapter 4. Overview of the Directory Tree. http://www.linuxdoc.org/LDP/sag/x264.html
- Documentación de los comandos **mount** y **umount**.
- De **info file** la sección *File Permissions*. En Internet esta documentación se encuentra en: http://www.gnu.org/manual/fileutils-3.16/html_mono/fileutils
- Consulte la documentación de **ln**.
- Opcional: documentación de los comandos `chgrp`, `chmod`, `chown` y `umask`.
- Puede resultar interesante para algunos lectores, un artículo escrito por los principales creadores de Unix (Keneth Thompson y Dennis Ritchie) sobre el sistema general y con una porción especial dedicada al sistema de archivos http://cm.bell-labs.com/cm/cs/who/dmr/cacm.html.
- El sistema ext2 está descrito en Design and Implementation of the Second Extended Filesystem Rémy Card, Theodore Ts'o, Stephen Tweedie. http://web.mit.edu/tytso/www/linux/ext2intro.html

4.4.3. Ejercicios: Administración de archivos

1. Entre los progamas del directorio `/bin` hay varios editores de texto. ¿Cuáles son y como opera cada uno?

2. Con el comando **chroot** ejecute el programa **cat** en un directorio diferente. Ayuda: Puede revisar las librerías que **cat** emplea con **ldd /bin/cat** (ver Administración de programas).

3. El ejemplo de **mknod** que referencia la primera consola virtual, en realidad referencia la entrada de teclado de la primera consola virtual. La información desplegada en la primera consola virtual puede verse en el dispositivo con número mayor 7 y número menor 1. En el directorio `/dev` debe existir un archivo que referencie ese dispositivo, identifiquelo y revise su contenido por ejemplo con `cat` o con `less` y la opción `-f`.

4. Hay un archivo en el que se almacena el mensaje de bienvenida que ven los usuarios que entran al sistema (por ejemplo al entrar desde una consola virtual o al hacer **telnet**), cambie el mensaje (Ayuda: el directorio `/etc` contiene la mayoría de archivos de configuración del sistema completo).

5. Examine las especificaciones de la CPU que tiene su sistema (Ayuda: comando **more** y un archivo del directorio `/proc`).

6. En el directorio `/dev` hay varios enlaces simbólicos para dispositivos, por ejemplo `/dev/modem` debe ser un enlace al puerto en el cual está el modem (e.g `ttyS1`), `/dev/cdrom` debe ser un enlace al dispositivo del CDROM (e.g `hdc`), `/dev/mouse` debe ser un enlace al puerto del ratón (e.g `psaux` o `ttyS0`). Revise los enlaces en su computador (si falta alguno agréguelo).

4.5. Administración de programas

Indicadores de Logros

- Sabe donde buscar software para Linux que pueda solucionar necesidades de la institución.
- Puede instalar paquetes `deb` y programas distribuidos como fuentes.
- Puede configurar rutas de directorios donde el sistema busca bibliotecas.

4.5.1. Lecturas: Administración de programas

Otra labor del administrador es buscar, instalar, actualizar y configurar los programas que requiera algún grupo de usuarios (es decir los programas disponibles normalmente en `/usr/bin` `/usr/local/bin` y `/opt`). Eventualmente puede promover el desarrollo de software para la institución.

4.5.1.1. Ejecución de programas y bibliotecas compartidas

El formato binario que soportan las distribuciones actuales de Linux es ELF [34] , el cual permite cargar bibliotecas compartidas [35] durante la ejecución —pueden verse datos del formato de un ejecutable ELF con el programa **readelf**. En este contexto biblioteca compartida se refiere a un conjunto de servicios que uno o más programas puede emplear y que están agrupados en un archivo —para ver las bibliotecas compartidas que un programa requiere puede emplear el programa **ldd** seguido de la ruta del programa (e.g **ldd /bin/cat**) y para ver los servicios que ofrece una biblioteca puede emplear **objdump**.

Para ejecutar un programa, el kernel lo transmite del dispositivo de almacenamiento a memoria RAM (después de que el intérprete de comandos lo ha ubicado empleando la variable PATH y ha decidido que no es un script ver Lectura Scripts básicos para bash).

una vez en RAM identifica el tipo de ejecutable y de ser un ELF emplea como intérprete `/lib/ld-linux.so`.

34. También soporta el formato binario aout en todas las arquitecturas, en Sparc soporta el formato iBCS2, con el módulo binfmt_misc puede agregarse soporte para otros formatos dinánicamente.
35. Biblioteca compartida: en inglés *shared library*.

El programa `/lib/ld-linux.so` es el encadenador dinámico, que encuentra las bibliotecas compartidas requeridas por el ejecutable, prepara el programa para ejecución y lo ejecuta [36] . Para localizar bibliotecas compartidas `/lib/ld-linux.so` busca en orden:

1. Los directorios especificados en la variable de ambiente LD_LIBRARY_PATH que deben separarse unos de otros con el caracter ':'
2. En el archivo `/etc/ld.so.cache`
3. en los directorios `/usr/lib` y `/lib`.

El archivo `/etc/ld.so.cache` es generado por el administrador con el programa **ldconfig**. Este programa genera `/etc/ld.so.cache` a partir de las bibliotecas disponibles en `/lib` y `/usr/lib` y en las rutas especificadas en `/etc/ld.so.conf`. Por esto si se instalan nuevas bibliotecas compartidas en rutas estándar es indispensable ejecutar **ldconfig** —la gran mayoría de programas de instalación lo harán automáticamente— y si se instalan en rutas no estándar asegurarse de que la ruta esté en `/etc/ld.so.conf` antes de ejecutar **ldconfig**.

Una vez se ejecute el programa, este podría leer algunos archivos de configuración (usualmente de `/etc/`) y variables de entorno. También podría cargar datos que requiera para operar (e.g mensajes para el locale apropiado, sonidos o imagenes usualmente del directorio `/usr/share`).

4.5.1.2. Compilación e instalación de programas a partir de fuentes

Aunque en los repositorios oficialesAlgunos programas de libre redistribución son liberados al público junto con el código fuente escrito en algún lenguaje de programación (tipicamente C o C++). Antes de usarlos debe compilarlos e instalarlos siguiendo las instrucciones que acompañan el código fuente; y claro antes debe descomprimirlo y/o desempaquetarlo.

El código fuente podría estar comprimido con diversos programas, dos muy comunes son: **gzip** - extensión .gz, **bzip2** - extensión .bz2. Ambos permiten descomprimir con la opción `-d` seguida del nombre del fichero comprimido. Además de comprimidas, las fuentes podrían estar archivadas con el programa **tar** (extensión .tar), para desarchivar ficheros creados por este programa se emplea **tar xvf *archivo*** (ver Administración de archivos, directorios y enlaces).

Aunque en general debe consultar las instrucciones para lograr la instalación y configuración, varios programas requieren la siguiente secuencia de comandos para compilar (secuencia establecida por autoconf):

```
./configure
make
make install
```

Esta secuencia instalará por defecto en el directorio `/usr/local`, si desea instalar en otra vía puede cambiar el primer comando por **configure --prefix=*nueva_via*** con el directorio destino en lugar de *nueva_via*. Si las fuentes incluyen mensajes en diversos idiomas (locales) se instalarán los especificados en la variable LINGUAS o todos si tal variable no está definida. La variable LINGUAS

36. En Linux recientes, por defecto todos los programas requieren bibliotecas compartidas —al menos requieren libc. Aunque esto puede cambiarse durante la compilación del programa con la opción `-static` del compilador gcc.

debe tener códigos ISO 639 (2 letras como LC_MESSAGE ver Lectura Configuración de una sesión) de los idiomas que se desean, unos separados de otro por espacios.

Cuando busque software, tenga en cuenta que funcione en el computador en el que operará (consulte los requerimientos de hardware y software), y revise la licencia del programa para asegurarse de la posibilidad de usarlo y redistribuirlo sin restricciones (ver Lectura Software de libre redistribución) —si no lee con detalle la licencia, podría infringir la ley.

Una vez consiga software que solucione una necesidad de su institución, debe instalarlo y configurarlo. Es posible que encuentre las fuentes comprimidas o de un archivo precompilado `deb` [37] . Por ejemplo para la distribución Debian, hay un inmenso repositorio de archivos precompilados. Cuando busque un programa, ese repositorio es el primer sitio quee puede consultar: http://packages.debian.org/stable/. Para hacerlo, puede emplear el programa dselect, que le permitirá entrar al repositorio principal de Debian o a otros, le permitirá elegir el o los programas que desee, le indicará si el programa es libre o no y le ayudará a resolver conflictos con otros programas o le indicará otros programas que son prerequisitos; tras esto, dselect se encarga de descargarlos a su máquina, instalarlos y configurarlos, todo automáticamente!

Puede darse el caso que el programa que usted busque no esté en ese repositorio, pero que lo logre encontrar en otra parte ya precompilado como `deb`. En ese caso puede descargarlo y emplear el programa dpkg para instalarlo en su sistema.

4.5.1.3. Paquetes en Debian

Para facilitar el uso de programas, evitando los posibles problemas que se presentan al compilar e instalar se emplean paquetes. Cada paquete es creado y mantenido por un voluntario (normalmente de Debian) e incluye programas precompilados, datos, archivos de configuración, scripts de instalación y desinstalación y datos de control (como nombre, firma digital, descripción, sección, paquetes de los que depende, paquetes con los que entra en conflicto, etc.). Dado que en un computador pueden instalarse muchos paquetes y su administración resulta delicada (porque algunos dependen o entran en conflicto

37. `deb` es la extensión de los archivo precompilados para la distribución Debian, en otras distribuciones el formato y extensión de los archivos precompilados es diferente.

con otros), en Debian se emplean herramientas que evitan conflictos y facilitan la instalación, desinstalación y actualización. La administración del sistema de paquetes (configuración de repositorios), así como las operaciones de paquetes pueden realizarse con el programa dselect que es guiado por menús y ofrece extensa ayuda. Como método de acceso es recomendable apt y especificar los repositorios editando el archivo `/etc/apt/sources.list`. Si planea emplear varios CDs con paquetes para Debian, al emplear `apt-cdrom add` el CD quedará registrado en `/etc/apt/sources.list` y podrá seleccionar e instalar los paquetes con dselect.

En esta sección se presentan detalles del sistema de paquetes de Debian y las herramientas que hay disponibles (las herramientas que usa **dselect** para automatizar la catalogación, obtención, instalación y desinstalación de paquetes).

Un paquete para Debian (archivo con postfijo `.deb`) es un archivo de ficheros creado con el programa **ar** [38] , que sigue diversas convenciones (por ejemplo contiene los datos del programa y datos para controlar instalación y desintalación comprimidos con **gzip** empleando nombres estándar). La herramienta dpkg-deb permite operar estos paquetes como archivos (no es recomendable instalar paquetes usando dpkg-deb).

Para instalar propiamente un paquete se emplea **dpkg -i *`archivo`* ...**. El programa **dpkg** además de verificar dependencias, eliminar versiones previas del mismo paquete, extraer archivos, ejecutar los scripts de instalación y configurar el paquete, actualiza los archivos con información sobre paquetes instalados y disponibles: `/var/lib/dpkg/status` y `/var/lib/dpkg/available` (estos archivos son también llamados caches por `apt-cache`).

Los caches además del nombre, descripción, versión, arquitectura y espacio en disco de cada paquete incluyen información para organizar y controlar los paquetes: estado, estado de selección, sección, prioridad, paquetes de los que depende, paquetes que reemplaza, con los que entra en conflicto y los que recomienda. Los posibles estados de un paquete son: instalado, instalado a medias, no-instalado, desempaquetado (archivos ubicados pero no configurado), configurado a medias, archivos de configuración [39] (sólo los archivos de configuración estan). Los estados de selección (usados por dselect) son: instalar, desinstalar, limpiar (desinstalar incluso archivos de configuración). Las secciones posibles son `main`, `contrib`, `non-US` y `non-free` diferenciadas por los términos de las licencias y la posibilidad de redistribuirlos. Los paquetes en main cumplen con los lineamientos de Debian para decidir si un programa es libre o no [40] —DFSG *Debian Free Software Guidelines*—, los paquetes en `contrib` cumplen con esos lineamientos pero requieren paquetes en `non-free` o `non-US`, los paquetes en `non-free` permiten libre redistribución pero no otros requerimientos de DFSG, los paquetes en `non-US` contienen material criptográfico que no puede exportarse de EUA (estos paquetes pueden ser usados en todo el mundo pero no pueden obtenerse de servidores en EUA). Las secciones se dividen en subsecciones: `admin`, `base`, `comm`, `devel`, `doc`, `editors`, `games`, `graphics`, `interpreters`, `libs`, `mail`, `misc`, `net`, `science`, `shells`, `sound`, `tex`, `text`, `utils`, `web`, `x11`, `comm`, `devel`. Las prioridades posibles son: *required* (mínimo para que el administrador inicie el sistema), *important* (aplicaciones mínimas encontradas en cualquier sistema Unix), *standard* (mínimo para tener un sistema en modo texto), optional (aplicación que no requiere ser instalada) y extra.

dpkg cuenta con varios comandos para operar paquetes y los caches, a continuación se presentan algunos:

38. **ar** se usa para crear archivos con la sintaxis **ar -r *`archivo fichero1 fichero2`* ...**, y para desempaquetarlos con **ar -x *`archivo`***.
39. El estado archivos de configuración es típico tras instalar y desintalar un programa, dpkg no borra archivos de configuración para que en siguientes instalaciones se tomen los valores por defecto de la instalación anterior.
40. La licencia de cada paquete está en el archivo `copyright` del directorio `/usr/doc/`*`paquete`*.

dpkg --configure *`paquetes`*

Configura un paquete cuya configuración esta pendiente. Si en lugar de usar un nombre de paquete se emplea `-a` configura todos los paquetes cuya configuración esté pendiente. Para reconfigurar un paquete que ya está instalado y que empleaba debconf se usa **dpkg-reconfigure *`paquete`***.

dpkg -r *`paquetes`*

Desinstala un paquete. Si se emplea `-a` en lugar del nombre de un paquete desinstala todos los que estén pendientes. La desinstalación de un paquete no elimina archivos de configuración, para eliminarlos puede emplearse `--purge` en lugar de `-r`.

dpkg --clear-avail

Elimina información sobre paquetes disponibles (`/var/lib/dpkg`).

dpkg -A -R *`directorio`*

Agrega al cache de paquetes disponibles, información de todos los paquetes en el directorio especificado y en sus subdirectorios. Pueden emplearse con un propósito análogo las opciones `--update-avail` y `--merge-avail` seguidas del nombre de un archivo con descripciones de paquetes (típicamente `Packages.gz`). Este método NO es el recomendado para actualizar paquetes disponibles, es más flexible emplear **apt** como se explica más adelante.

dpkg -l '*`cadena`*'

Presenta información de los paquetes disponibles que concuerden con la cadena. La cadena puede tener caracteres para expansión de nombres.

dpkg -s *`paquetes`*

Presenta el estado de los paquetes especificados.

dpkg -C

Busca paquetes cuya instalación no este completa.

dpkg -L *`paquetes`*

Muestra la ubicación de los ficheros de los paquetes dados.

dpkg -S *`fichero`*

Busca el paquete que haya instalado el fichero especificado.

dpkg -x *`paquete directorio`*

Descomprime un paquete en el directorio especificado sin instalarlo.

El inconveniente de usar **dpkg -i *`archivo`* ...** para instalar un paquete es que no resuelve problemas de dependencias (por ejemplo si el paquete que se instala requiere otro que aún no se ha instalado). Para resolver dependencias y obtener e instalar paquetes se requiere uno o más repositorios de paquetes y se usa **apt-get install *`paquete`* ...** [41].

41. El programa **apt-get** resuelve dependencias y descarga paquetes de repositirios, para la instalación de un paquete ya descargado usa el programa **dpkg**.

Los repositorios de paquetes contienen un fragmento de los paquetes disponibles para Debian, pueden ser medios de almacenamiento (CD-ROMs, directorios locales o compartidos, disquettes) o sitios en una red a la que su computador está conectado (pueden usarse los protocolos FTP y HTTP). Los posibles repositorios de paquetes se especifican en el archivo `/etc/apt/sources.list` y normalmente el cache de paquetes disponibles en su sistema (`/var/lib/dpkg/available`) es extraido de los archivos `Packages.gz` de cada repositorio de paquetes.

Para actualizar el cache con la información más reciente en los repositorios puede emplear **apt-get update**, para agregar un CD-ROM como repositorio y actualizar el cache de paquetes disponibles emplee `apt-cdrom` con la opción `add`. Para agregar repositorios FTP o HTTP edite el archivo `/etc/apt/souces.list` y agregue líneas de la forma:

```
deb ftp://ftp.debian.org/debian stable main contrib non-free
```

remplazando el URL por el del depósito. A la derecha del URL se especifican las secciones que desea emplear de ese repositorio. La cadena "deb" a la izquierda del URL indica que se trata de un repositorio de paquetes precompilados, puede agregarse un repositorio de fuentes de programas con "deb-src".

Los comandos para `apt-get` son:

`update`

Actualiza caches de paquetes con información de los repositorios especificados en `/etc/apt/sources.list`. De cada repositorio extrae los paquetes disponibles del archivo `Packages.gz`. Debe hacerse antes de *`upgrade`* o *`dist-upgrade`*.

`upgrade`

Instala las versiones más recientes de todos los paquetes, mientras para una instalación no requiera reinstalar otro paquete. Emplea los repositorios especificados en `/etc/apt/sources.list`.

`dselect-upgrade`

Completa los cambios que un usuario haya especificado (pero no completado) con `dselect`.

`dist-upgrade`

Realiza la misma función de `upgrade` tratando de resolver conflictos entre nuevas versiones y existentes.

`install paquete ...`

Obtiene el/los paquete(s) especificado(s) de alguna de los repositorios y lo(s) instala, así como los paquetes que sean requeridos. El nombre del paquete no es una ruta, es la identificación del paquete en el cache o una expresión regular de identificaciones de paquetes. Puede marcarse un paquete para su futura desintalación agregandole al nombre el caracter '-' (con '+' se marca para futura instalación).

`remove paquete ...`

Desinstala el(los) paquete(s) especificado(s). Sigue convenciones análogas a **install**.

`source`

Obtiene las fuentes de un paquete de un repositorio de fuentes (los especificados en `/etc/apt/sources/lists` en líneas `deb-src`). Dejas las fuentes en el directorio de trabajo.

`build-dep paquete ...`

Instala paquetes que satisfacen dependencias de uno o más paquetes.

`clean`

Limpia el directorio donde se dejan paquetes extraidos de repositorios `/var/cache/apt/archives`.

`autoclean`

Realiza la misma función de **clean** pero evita eliminar paquetes útiles.

`check`

Examina el cache de paquetes buscando posibles problemas en las dependencias.

Entre las opciones que **apt-get** recibe están:

`-f`

Intenta solucionar problemas de dependencias.

`-d`

Descarga todos los paquetes pero no los instala. Una vez descargados para instalarlos se usa el mismo comando de **apt-get** que los descargo pero sin `-d`.

`-m`

Ignora paquetes faltantes.

`-no-download`

Evita que se descarguen paquetes de la red (útil junto con `-m`).

`-o`

Opción para establecer una opción que tome precedencia sobre las opciones del archivo de configuración. El archivo de configuración es indicado por la variable APT_CONFIG o en su defecto es `/etc/apt/apt.conf`.

El programa **apt-cache** le permite operar los caches de paquetes instalados y disponibles simultaneamente, algunas de las operaciones que permite son:

show *`paquete`*

Muestra la información de control del paquete especificado.

showpkg *`paquete`*

Muestra información de dependencias de un paquete. Presenta tanto los paquetes de los cuales depende como los paquetes que dependen de ese (dependencias reversas).

search *expreg*

Busca paquetes disponibles e instalados cuyo nombre concuerde con una expresión regular.

4.5.1.4. Depósitos de software de libre redistribución

Cuando busque software, tenga en cuenta que funcione en el computador en el que operará (consulte los requerimientos de hardware y software), y revise la licencia del programa para asegurarse de la posibilidad de usarlo y redistribuirlo sin restricciones (ver Lectura Software de libre redistribución) —si no lee con detalle la licencia, podría infringir la ley.

En Internet hay varios sitios que indexan fuentes de software de libre redistribución que funciona en Linux algunos son:

- http://www.freshmeat.net
- http://www.ibiblio.org/pub/Linux/
- http://www.sourceforge.net

Para la distribución Debian hay varios repositorios de fuentes y precompilados, puede consultar el principal (el repositorio oficial de Debian) en: http://packages.debian.org/stable/.

Para configurar los repositorios que usa apt, puede emplear el programa `apt-setup`, o agregar líneas a su archivo `/etc/apt/sources.list` como:

`deb ftp://ftp.debian.org/debian potato main contrib non-free`

Repositorio principal de Debian. Puede haber un espejo que funcione mejor con la conexión a Internet que tiene, consulte la lista de espejos en http://www.debian.org/misc/README.mirrors

`deb ftp://ftp.debian.org/debian-non-US potato non-US/main`

Repositorio non-US de Alemania. Contiene software que no puede exportarse de EUA por restricciones en criptografía.

`deb http://red-carpet.ximian.com/debian potato main`

Repositorio de Ximian Gnome.

`deb http://kde.tdyc.com/debian potato main`

Repositorio de KDE para Debian 2.2.

`deb http://dtype.org/debian/potato ./`

Repositorio con versiones recientes de algunos paquetes (como el navegador Mozilla).

`deb http://people.debian.org/~synrg/task-junior /`

Repositorio de Debian-Jr, con paquetes enfocados a educación en casa (en especial juegos).

Adicionalmente puede encontrar un listado de otras fuentes no oficiales de paquetes debian en http://www.internatif.org/bortzmeyer/debian/apt-sources/ o http://channel.debian.de/faq/DebianDE-15.html.

4.5.1.5. Su propio depósito de paquetes

En una red puede configurarse un computador para mantener los paquetes que se emplean en el resto y para así hacer actualizacioes por ftp o http. La forma más sencilla es copiar los paquetes en un directorio que pueda accederse por http y emplear tal ruta desde los computadores clientes en `/etc/apt/sources.list`. Por ejemplo si copia paquetes en un directorio que pueda consultarse con el URL `http://servidor.micolegio.edu.co/paquetes` agregue a `/etc/apt/sources.list` la línea:

```
deb http://servidor.micolegio.edu.co/paquetes /
```

También puede copiar los CDs de instalación de Debian en un directorio de ftp (digamos /debian) y agregar en `/etc/apt/sources.list`:

```
deb ftp://servidor.micolegio.edu.co/debian/cd1 stable/non-US main contrib
deb ftp://servidor.micolegio.edu.co/debian/cd1 stable main contrib non-free
deb ftp://servidor.micolegio.edu.co/debian/cd2 stable/non-US main contrib
deb ftp://servidor.micolegio.edu.co/debian/cd2 stable main contrib non-free
deb ftp://servidor.micolegio.edu.co/debian/cd3 stable/non-US main contrib
deb ftp://servidor.micolegio.edu.co/debian/cd3 stable main contrib non-free
```

A continuación se explica como puede crear su propio repositorio de paquetes sólo con los paquetes que usted escoja [42].

Un repositorio http (o ftp) puede constar de uno o más directorios, en cada directorio debe haber paquetes .deb y un índice Packages o mejor Packages.gz. El índice contiene la descripción de cada paquete separada por varias líneas en blanco, por ejemplo una entrada de un índice es:

```
Package: asigna
Version: 1.0-1
Priority: optional
Section: education
Maintainer: Vladimir Tamara <vladimir@tamarapatino.com>
Architecture: i386
Filename: ./asigna_1.0-1_i386.deb
Size: 18016
MD5sum: fef055d6ed02c211e3234aae34efda61
Description: This program helps to arrange a schedule for a school
This program receives information  and restrictions on the schedule of
a school  and tries  to create one.   The information it  receives is:
amount and  labels for teachers,  amount and labels for  rooms, amount
and  labels for  hours. The  restrictions that  can accept are unmovible
teachers (at certain hour in certain room).
installed-size: 104
```

42. Un ejemplo es el repositorio de paquetes de Structio, donde en particular están estas guías. Si cuenta con acceso a Internet puede accederlo con `apt` agregando a `/etc/apt/sources.list` la línea `deb http://structio.sourceforge.net/debian/binary-i386 /`

Este índice preferiblemente debe generarse automáticamente con el programa dpkg-scanpackage, creando antes un archivo `override`. El archivo `override` contiene un mínimo de información de los paquetes, cada línea tiene nombre, prioridad y sección de un paquete separados por espacio. Por ejemplo una línea puede ser:

```
asigna optional education
```

Estando en el directorio del repositorio, si el archivo `override` está allí mismo se genera el índice de paquetes con:

```
dpkg-scanpackages . override > Packages
rm Packages.gz
gzip Packages
```

4.5.1.6. Creación de paquetes

Para crear sus paquetes Debian, debe contar con las fuentes del programa que desea empaquetar e instalar, con las herramientas requeridas para compilarlo e instalar el paquete `task-debian-devel`. El nombre del directorio con fuentes debe ser de la forma *paquete-version* con el nombre del paquete en minúsculas y la versión con digitos, letras, punto y guión. Pase al directorio con las fuentes y ejecute `dh_make`, por ejemplo:

```
cd AA_Linux_colegio-1.1
dh_make -e paz@micolegio.edu.co -f ../AA_Linux_colegio-1.1.tar.gz
```

subdirectorio `debian` en tales fuentes

Cada vez que necesite regenerar el paquete ejecute desde el directorio con fuentes:

```
dpkg-buildpackage -rfakeroot
```

4.5.2. Lecturas recomendadas: Administración de programas

- Para conocer más sobre librerías compartidas puede consultar las páginas `man` de `ld.so`, **ldconfig** y **ldd**. También puede consultar *Program Library HOWTO*. Sección 3: Shared Libraries. http://www.linuxdoc.org/HOWTO/Program-Library-HOWTO/shared-libraries.html
- Otra explicación del sistema de paquetes en Debian, está disponible en: *Debian Guide. Removing and Installing Software*. http://www.newriders.com/debian/html/noframes/node84.html
- Puede aprender más sobre el sistema de paquetes de Debian, consultando las páginas `man` de **dpkg**, **apt-get**, **apt-cache**, **dpkg-reconfigure** y **apt-cdrom**. La guía del usuario de APT en el directorio `/usr/doc/apt/guide.text.gz`, una guía para actualizar varios computadores en red apoyandose

en APT en `/usr/doc/apt/offline.text.gz` y los lineamientos de Debian sobre software libre (DFSG) en el *Contrato Social de Debian* `/usr/doc/debian/social-contract.txt` y en la red en http://www.debian.org/social_contract.

- Para aprender más sobre compresión y archivos, consulte las páginas `man` de **zip**, **unzip**, **gzip** y **tar**.
- El paquete `maint-guide` es una guía para quienes desean crear paquetes Debian (queda instalada al instalar `task-debian-devel`). Una vez instalada puede consultar la guía en diversos formatos en `/usr/doc/maint-guide`, o en Internet en http://www.debian.org/doc/maint-guide/

4.5.3. Ejercicios: Administración de programas

1. Si un usuario instala un programa en el directorio `/home/pedro/bin` y una biblioteca compartida nueva que ese programa requiere en el directorio `/home/pedro/lib`, ¿qué debe agregar él a su archivo `~/.bashrc` para poder ejecutarlo desde un intérprete de comandos sin especificar la ruta completa?

2. Con respecto al administrador no quiere usar LD_LIBRARY_PATH

Con respecto al punto anterior, si el administrador tras acuerdo con el usuario quiere permitir a otros usuarios usar el programa desde un intérprete de comandos (para cargarlo sin especificar la ruta completa), pero quiere además: que el programa se mantenga en el directorio del usuario, que no se deben modificar los archivos de inicialización de los usuarios y que no se use LD_LIBRARY_PATH

3. Hay diversos sitios en Internet donde puede buscarse software para Linux: http://www.freshmeat.net, http://www.linux.org/apps/index.html. Busque otro y compárelos.

4. Puede consultar software enfocado al sector educativo que funciona sobre la plataforma de referencia en el primer apéndice de estas guías (ver Plataforma de referencia) y en Internet en índices especializados: http://ingenieria.sanmartin.edu.co/cgi-bin/slec/slec.cgi/slec.software_librered?style=slec&site=slec&, http://www.ofset.org/freeduc, http://richtech.ca/seul/, http://edvisor.org . Busque las fuentes de un programa de libre redistribución que necesite, instálelo y describa el proceso que siguió.

5. Examine la lista de bibliotecas compartidas que requiere la calculadora gcalc.

6. Algunos programas instalan o usan bibliotecas compartidas. Agregue la vía `/opt/lib` a la lista de directorios donde el sistema buscará bibliotecas compartidas.

7. Consiga un programa pequeño (que quepa en un disquete) en formato `deb`, instálelo, úselo y desinstálelo.

8. Cree un archivo `.tar.gz` de un directorio pequeño (por ejemplo cree uno de prueba con al menos dos archivos).

4.6. Colaboración y soporte

4.6.1. Lectura Colaboración y soporte

Una de las características de Linux y en general del software de libre redistribución es que proviene y se mantiene de la colaboración desinteresada. Como usuario o administrador del sistema usted podrá disfrutarla, pero además puede brindarla para facilitar la continuidad y mejoramiento de este tipo de software. Además está en sus manos difundir este valor entre los usuarios de la red a su cargo.

En esta sección se presentan algunos proyectos y grupos existentes en Internet en los que podrá participar y que podrá difundir entre las personas que emplean su sistema (claro está la mejor forma de difundir el valor de la colaboración es dando ejemplo).

4.6.1.1. Grupos de usuarios

Hay diversos grupos de usuarios en Internet a los que puede recurrir para pedir ayuda para administrar mejor su sistema Linux y en los que podrá ayudar a otras personas que necesiten colaboración.

- Si desea dar y recibir ayuda en aspectos técnicos de la plataforma de referencia S-Helio 1.1, puede emplear la lista de distribución `<structio-sopcol@lists.sourceforge.net>`.

  Puede encontrar información para suscribirse en: http://lists.sourceforge.net/lists/listinfo/structio-sopcol

- Si desea dar y recibir ayuda en aspectos menos técnicos pero relacionados con software de libre redistribución y educación en Colombia, sugerimos de manera especial que se suscriba y participe en la comunidad de voluntarios SLEC (Software de libre redistribución y educación en Colombia). En la lista de este grupo (`<slec@yahoogroups.com>`), podrá pedir ayuda para configurar hardware y software y para administrar mejor la red Linux de su colegio. Podrá también mostrar lo que está haciendo en su institución y al decirlo podría animar a otras personas.

  Puede suscribir su correo desde: http://ar.groups.yahoo.com/group/slec/ (http://ar.groups.yahoo.com/group/slec/) o enviando un correo en blanco a: `<slec-subscribe@egroups.com>`

- Hay otros grupos de usuarios en diversas partes de Colombia que prestan soporte en diversos aspectos relacionados con Linux. Puede consultar un listado en http://groups.yahoo.com/group/slec/links/Grupos_000999083920/.

4.6.1.2. Proyectos

Hay muchos proyectos en curso en los que puede participar. En muchos casos no se requieren amplios conocimientos de programación, cualquier habilidad que usted tenga puede ser aprovechada. Incluso algo tan sencillo como leer documentos, encontrar y reportar fallas es una valiosa ayuda.

- Structio: estas guías son parte del proyecto Structio, podrá consultar más información y consultar formas de ayudar en http://structio.sourceforge.net
- Proyectos de Traducción a Español (manuales, guías, programas): http://www.croftj.net/~barreiro/colabora.html
- Proyecto de Traducción en Castellano, Lucas (Linux en Castellano): http://lucas.hispalinux.es/htmls/participar.html

- También en el proyecto GNOME: http://www.gnome.org/todo/index.php3 y en el proyecto GNOME Hispano: http://es.gnome.org/colabora.html
- También a hacer la distribución Debian: http://www.debian.org/
- Además de permitirle colaborar en proyectos de software y documentación de libre redistribución, el acceso a Internet le da la posibilidad de colaborar en proyectos humanitarios a distancia. Varios de estos proyectos, aunque pueden no tener relación con software de libre redistribución, promueven la ayuda desinteresada: http://www.netaid.org

4.6.2. Ejercicios: Colaboración y soporte

1. Busque grupos de usuarios de Linux en Colombia y comparta su opinión sobre cada uno con sus compañeros de clase.

2. Busque otro proyecto relacionado con software de libre redistribución en el que usted podría participar y comparta la información con sus compañeros durante la clase.

Capítulo 5. Administración de Linux en computadores aislados

Público y logros

Administrador de Red del colegio quién solucionará problemas cuando se presenten o instalará la plataforma de referencia en computadores fuera de red.

- Conoce la secuencia de inicialización y puede configurar algunos detalles.
- Emplea módulos, el kernel y otros programas para configurar un sistema.
- Reconoce los componentes físicos de un computador, puede manipularlos y configurar algunos con Linux.
- Puede instalar la distribución Debian en un computador y configurarla de acuerdo a los lineamientos de la plataforma de referencia S-Helio 1.1.

5.1. Inicialización del sistema

Indicadores de Logros:

- Conoce la secuencia de inicialización.
- Configura algunos detalles de la inicialización del sistema.

5.1.1. Lectura: Inicialización del sistema

Un aspecto que un administrador de un sistema Linux debe considerar en ocasiones, es la secuencia de inicialización del sistema. Esto puede servir para iniciar junto con el sistema servicios importantes (especialmente si el computador se emplea como servidor), para eliminar servicios que no se usen y así aprovechar mejor los recursos o para afinar algunos detalles que afectarán a todos los usuarios del sistema. Por otra parte algunas fallas que pueden ocurrir en un sistema Linux pueden deberse a una secuencia de inicialización errada y para solucionarlas puede ser necesario seguirla de cerca.

5.1.1.1. Secuencia de inicio

Al iniciar un computador con Linux ocurren gran cantidad de procesos, deben ocurrir en un orden preciso porque unos dependen de otros (e.g. el cargador de arranque requiere la BIOS, el kernel requiere la inicialización que hace el cargador de arranque, los módulos requieren el kernel operando, las librerías y programas requieren el kernel y los módulos para interactuar con el usuario).

A grandes rasgos la secuencia de inicialización es:

1. Un programa mantenido de forma permanente en memoria del computador (BIOS [1]) inicializa varios aspectos del hardware, después de acuerdo a su configuración carga el primer sector de un disquete o de un disco duro [2] o de un CD-ROM y ejecuta el programa que allí se encuentra. Si el disquete/ disco/CD-ROM permite arrancar un sistema operativo tal programa se llama un *cargador de arranque* [3].

   Algunos aspectos configurables de la BIOS se mantienen en un tipo de memoria especial (e.g. EPROM que, usando una pila, puede mantener datos cuando el computador está desconectado), pueden configurarse empleando un programa[4].

2. Algunos cargadores de arranque (como LILO o GRUB) pueden permitir al usuario escoger el sistema operativo al cual desea entrar (cuando en el mismo computador haya varios), y carga el kernel del sistema escogido, eventualmente pasando parametros dados por el usuario (ver Parámetros desde el cargador de arranque) [5]. Para sistemas Linux hay varios cargadores de arranque, los más conocidos son LILO y GRUB. También es posible arrancar un sistema Linux desde DOS con el cargador `loadlin`.

3. Si el sistema escogido es Linux, el cargador de arranque, carga el kernel de este sistema. El kernel detecta algunos detalles del hardware (como el procesador, cantidad de memoria y algunos dispositivos), procesa parametros recibidos del cargador de arranque, inicializa algunos aspectos del hardware, prepara áreas de memoria, prepara el sistema de archivos virtual (ver Sistemas de archivos en Linux), prepara el sistema de módulos y lanza como primer proceso **/sbin/init** (a menos que se haya especificado otro programa como parámetro ver Parámetros desde el cargador de arranque). Varios de los mensajes que el kernel produce durante esta etapa pueden ser revisados en el archivo `/var/log/dmesg` y con el programa **dmesg**.

   Como los dispositivos de cada computador son diferentes, el kernel cuenta con controladores [6] para muchos tipos de hardware diferente (tanto para hardware muy antiguo, como para hardware muy nuevo) y para no ocupar tanto espacio en memoria el kernel emplea un sistema de módulos. Si todos los controladores para todos los tipos de hardware que el kernel soporta, se incluyeran directamente en el kernel (i.e estaticamente) desde el arranque, el kernel requeriría bastante memoria RAM (un kernel 2.2.18 con módulos requiere más de 13MB) y la gran mayoría de ese espacio no sería usado, sólo serían usadas las partes que controlan los dispositivos particulares de su computador.

   Un módulo es una parte del kernel que puede cargarse, usarse o quitarse después de que el kernel está funcionando. Algunos módulos ofrecen servicios (e.g soporte para cierto sistemas de archivos) y otros son controladores para hardware específico (por ejemplo hay módulos para tarjetas de sonido, para tarjetas de red, etc.). Durante el proceso de inicialización del sistema, algunos dispositivos son detectados y los módulos apropiados son cargados, sin embargo como se explica en este capítulo algunos dispositivos no pueden ser detectados o configurados automáticamente e incluso algunos no son soportados.

---

1. BIOS es sigla de *Basic Input Output System*.
2. El primer sector de un disco duro tiene la tabla de particiones.
3. Cargador de arranque: del inglés *boot loader*.
4. En PCs el programa para configurar la BIOS normalmente está incluido en la BIOS misma y puede ejecutarse tras encender el computador presionando alguna tecla (e.g. **Del** o **F1**).
5. Además de los parámetros `single` y `init=`*`programa`* pueden emplearse parámetros específicos para módulos compilados estáticamente en el kernel y otros generales como `ro` o `rw` para montar la raíz en modo de sólo lectura o de escritura y lectura respectivamente, `root=`*`dispositivo`* para especificar el dispositivo para montar como raíz, `debug` o `quiet` para que el kernel muestre el máximo o el mínimo de mensajes informativos. Otros pueden verse con `man bootparam`.
6. Controlador: del inglés *driver*.

4.

El programa **/sbin/init**, que normalmente es el primer proceso ejecutado por el kernel completa la secuencia de inicialización ejecutando los procesos de un nivel de ejecución. El nivel de ejecución se asocia con un número entre 0 y 6, o bien con ’S’ de forma que el administrador puede configurar cada nivel de ejecución para que inicie procesos diferentes. Durante un arranque normal init ejecutará primero los procesos del nivel S (scripts del directorio `/etc/rcS.d` en orden). Estos scripts emplean algunas variables definidas en `/etc/default/rcS` para inicializar dispositivos y programas. A continuación presentamos la secuencia configurada en un sistema Debian recién instalado, posiblemente no entenderá todos los términos que serán introducidos en este y el próximo capítulo: instala el mapa de teclado (por defecto `/etc/console-tools/default.kmap.gz` que usted puede modificar de ser necesario), activa swap, revisa y de ser necesario monta la partición raíz y monta el directorio virtual `/proc`, configura dispositivos ISA Plug and Play de acuerdo a al archivo `/etc/isapnp.conf`, calcula dependencias entre módulos y carga los módulos y argumentos especificados en `/etc/modules`, chequea los demás sistemas de archivos que tengan inconsistencias o que hayan alcanzado la cuenta de montajes antes del chequeo [7], configura variables del kernel especificadas en `/etc/sysctl.conf`, configura puertos seriales con la última configuración usada, prepara dispositivos dinámicos `/dev/pts` [8], monta particiones locales y prepara zonas de intercambio, en sistemas que usan PPP limpiar cache de DNS, establece el nombre de la máquina tomandolo de `/etc/hostname`, activa interfaces de red empleando la configuración de `/etc/network/interfaces` y `/etc/network/options`, inicia interfaces que requieran BOOTP o DHCP, activa portmap, monta directorios remotos (con nfs, smb, o ncp), configura fuentes por usar en consolas virtuales [9], ajusta la hora del sistema y realiza otras labores de limpieza y mantenimiento rutinarias.

5.

Después de ejecutar scripts del nivel S, **init** ejecuta los del nivel de ejecución configurado (el nivel se configura en el archivo `/etc/inittab` en una línea que dice `initdefault`). Por defecto es el nível 2 cuyos scripts están en el directorio `/etc/init.d/rc2.d`.

6.

A menos que se haya cambiado la configuración de **/sbin/init**, a continuación se ejecutan los scripts del nivel configurado, se inician terminales tty [10] en diversas consolas virtuales (por defecto en Debian de la 1 a la 6) que permiten a los usuarios iniciar sesiones tipo texto con el intérprete de comandos (ver Lectura Sistema de usuarios y manejo de clave). Para iniciar esas terminales usa el programa `getty` (ver Inicio de sesiones en consolas virtuales).

7.

7. Si existe `/fastboot` no hace el chequeo, si existe `/forcefsck` chequea independiente de la cuenta de montajes. Este chequeo arregla errores sólo si `FSCKFIX` es `yes` en `/etc/defaults`.
8. Se trata de pseudoterminales, por ejemplo usadas por terminales X-Window.
9. Las fuentes por usar en consolas virtuales se configuran en `/etc/console-tools/config`
10. Una terminal tty permite hacer login en un sistema Linux, puede ser una consola virtual, un puerto serial (o modem) o una seudoterminal pty –como las terminales iniciadas en X-Window.

Si X-Window está configurado y si hay instalado un administrador de vistas [11] —como xdm, kdm o gdm— el último script del nivel 2 iniciará X-Window en la consola virtual 7 y permitirá a los usuarios iniciar sesiones gráficas con su login y clave (ver Inicio de sesiones X-Window).

5.1.1.2. Configuración de los niveles de ejecución

A menos que se especifique un proceso inicial diferente desde el cargador de arranque (con el parámetro **init** ver Parámetros desde el cargador de arranque por defecto el primer proceso es `/sbin/init`. Este programa lee el archivo de configuración `/etc/inittab` para iniciar otros procesos organizados en niveles de ejecución y con ellos el resto del sistema operativo.

Cada línea del archivo `/etc/inittab` puede ser un comentario (si comienza con el caracter '#') o de la forma:

```
id:niveles:método:comando
```

Donde id es una identificación arbitraria de hasta de 4 letras, niveles indica el nivel o los niveles en los cuales se aplica, comando es el comando por ejecutar y método permite especificar la forma de iniciarlo o circunstancias bajo las cuales iniciarlo [12]

En un sistema que ya esta inicializado se puede emplear el programa **/sbin/init** (o **/sbin/telinit**) para cambiar el nivel de ejecución, especificando el nuevo nível como parámetro (e.g **/sbin/init 1** ver Encendido).

En Debian un nivel de ejecución es una configuración del sistema en la que sólo están presentes ciertos procesos, tales procesos son iniciados por scripts en un directorio particular (e.g `/etc/rc2.d` para el nivel 2). Los nombres de algunos de esos scripts comienzan con la letra 'S', esos son ejecutados cuando el sistema entra al nivel, los que comienzan con la letra 'K' son ejecutados cuando se sale del nivel (después de la letra 'S' o 'K' sigue un número de dos digitos que especifica el orden en el cual se ejecutan —primero los de menor número—). Al entrar a un nivel los scripts 'S' son ejecutados con el parámetro "start", mientras que al salir los scripts 'K' se ejecutan con el parámetro "stop". Los niveles normales son:

0

Este nivel de ejecución es empleado para detener el sistema

11. Administrador de vistas: del inglés *display manager*.

12. Los posibles métodos para iniciar un comando son: `respawn` que indica que se reiniciará el proceso cuando termine, `wait` que indica que se ejecutará una vez el proceso cuando se ingrese al nivel de ejecución y se esperará que complete antes de continuar, `once` similar a `wait` pero continua ejecutando otros proceso sin esperar a que termine, `boot` ejecutará el proceso durante el arranque (ignorando el nivel de ejecución) y continuará con otros sin esperar a que termine, `bootwait` análogo a `boot` pero espera a que termine antes de ejecutar otro proceso, `off` no hace nada, `ondemand` permite especificar tres niveles de ejecución especiales a, b y c, `initdefault` especifica el nivel de ejecución por defecto (el comando es ignorado), `sysinit` ejecuta durante el arranque antes de otros marcados con `boot` o `bootwait` (ignora nivel de ejecución), `powerwait` ejecuta el comando cuando una UPS indique que se acabó el suministro de energía (`init` espera a que el comando termine) –una UPS es un dispositivo con una batería que puede mantener encendido un computador por un tiempo después de un corte de energía eléctrica—, `powerfail` análogo a `powerwait` pero `init` no espera a que concluya, `powerokwait` se ejecutará el comando cuando el suministro de energía es restablecido, `powerfailnow` ejecuta el comando cuando la UPS indica que su batería está próxima a acabarse, `ctraltdel` proceso ejecutado cuando `init` recibe la señal `SIGINT` –que normalmente ocurre cuando el usuario presiona Ctrl-Alt-Del—, `kbrequest` ejecuta el comando cuando recibe la señal `SIGWINCH` —que puede ser generada por una secuencia de teclas especiales.

Nivel para detener el sistema, es el nivel al que entra con los comandos **halt** o **shutdown -r** (ver Apagado y encendido de una red). Sus scripts están en el directorio: `/etc/rc0.d`.

1

Este nivel de ejecución es apropiado para un sólo usuario, sin conexión a red (modo mono-usuario).

Modo mono usuario [13], apropiado para ciertas tareas administrativas. Entrará a este nivel con el comando **shutdown**). Sus scripts están en `/etc/rc1.d`.

2-5

Operación normal multi usuario. Scripts en `/etc/rcx.d` (con un digito de 2 a 5 en lugar de *x*).

6

Este nivel de ejecución es empleado para reiniciar el sistema por **reboot** y **shutdown**.

Reinicio [14]. Entrará a este nivel con los comandos **reboot** o **shutdown -r**. Scripts en `/etc/rc6.d`.

Los archivos de los directorios `/etc/rcx.d` son enlaces a scripts del directorio `/etc/init.d`, esos scripts tiene el nombre del paquete (ver Paquetes en Debian) al que pertenecen y pueden recibir como parámetro: *start*, *stop*, *restart* (detener y reiniciar), *force-reload* (volver a leer archivos de configuración sin detener o de no ser posibile detener y reiniciar). Pueden crearse directamente con **ln** o en Debian con el programa `update-rc.d`, por ejemplo:

```
update-rc.d xdm start 99 2 3 4 5 . stop 20 0 1 6
```

que indica que el script xdm (el cual debe estar en `/etc/init.d`) debe ser ejecutado al entrar a los niveles 2, 3, 4 y 5 con prioridad 99 (última). Así como cuando el sistema termina normalmente (con halt o `reboot`) o cuando sale de modo monousuario.

5.1.2. Lecturas recomendadas: Inicialización del sistema

- Páginas man de init, inittab, update-rc.d.
- Puede consultarse otra explicación del proceso de inicialización en el Capítulo 7 de "*The Linux System Administrators' Guide* (Version 0.6.2)" disponible en http://www.tldp.org/LDP/sag/x1901.html.

13. Mono usuario: del inglés *single user*
14. Reinicio: del inglés *reboot*

5.1.3. Ejercicios: Inicialización del sistema

1. ¿Qué proceso tiene el identificador 1 en un arranque normal? (puede usar **ps ax** si requiere ayuda).

2. Identifique el script en el que se monta el directorio virtual `/proc` durante el arranque.

3. De forma análoga a la descripción de los scripts del nivel S presentado en esta sección, describa lo que hacen los scripts del nível de ejecución 2.

4. Agregue en el nivel de ejecución 3 un script que le envie un correo a su cuenta de usuario con la fecha y hora.

5. Cambie al nivel de ejecución 3 y compruebe que el script que creo en el punto anterior funciona.

5.2. Kernel y módulos

Indicadores de Logros:

- Maneja sistema de módulos.
- Puede configurar, cambiar o recompilar el kernel.

5.2.1. Lectura: Kernel y módulos

En un sistema Linux la interacción final con dispositivos la realizan los controladores o el kernel. Dicho de otra forma, un dispositivo sólo podrá ser usado si el kernel lo soporta o si existe un controlador capaz de controlarlo y si se configura apropiadamente para hacerlo. Por esto, los dispositivos que se produzcan hoy, no pueden ser operados con controladores ya existentes (a menos que los dispositivos hayan sido diseñados para ser operados con estos —caso que para algunos dispositivos como tarjetas de sonido, tarjetas de video o modems es poco común), por esto mismo, eventualmente, si actualiza el hardware de su computador o si desea usar un cierto servicio del kernel, también deberá actualizar algún módulo o eventualmente el kernel completo. Aunque algunos controladores autodetectan la configuración del dispositivo, esto no siempre ocurre y en algunos casos tendrá que probar (cacharrear), hasta hacer funcionar el módulo que maneja el dispositivo (en estas pruebas la documentación del dispositivo, la del módulo y estas guías, esperamos le sean de utilidad —si desea experimentar rapidamente emplee **modconf**).

Las fuentes en C de cada versión del kernel cuentan con controladores para diversos dispositivos. Cuando se compila una versión, algunos de esos controladores pueden unirse con el kernel mismo (estáticamente), otros pueden dejarse como módulos para cargarse/descargarse cuando la parte estática del kernel este operando, otros pueden ser excluidos del proceso de compilación (y por lo tanto no podrán ser usados ni cuando el kernel esté operando).

5.2.1.1. Módulos

En este contexto, módulo se refiere a un controlador de un dispositivos o servicio que puede cargarse o descargarse cuando el usuario o algún dispositivo lo solicita (i.e dinámicamente). Los módulos que se distribuyen con en el kernel están ubicados en el directorio `/lib/modules/`*`version`*, donde *`version`* es la versión de su kernel [15], con la extensión `.o` [16] organizados en directorios que indican el tipo de dispositivo o el propósito, por ejemplo `fs` - sistema de archivos, `net` - protocolos y hardware para redes.

Para lograr configurar un dispositivo controlado por un módulo, puede emplear las herramientas del paquete `modutils` o **modconf** para:

1. Asegurar que no haya conflictos entre el dispositivo con otros y eventualmente conocer la configuración que usa (algunos controladores autodetectan la configuración del dispositivo, pero no todos).
2. Encontrar un módulo que pueda manejar el dispositivo.
3. Eventualmente pasar opciones al módulo de acuerdo a la configuración del dispositivo (e.g IRQ o dirección base).

En Debian las herraminetas del paquete `modutils` se configuran en los achivos del directorio `/etc/modutils` (con la información de esos archivos y de `/etc/modules` se genera el archivo de configuración de módulos `/etc/modules.conf` [17]). Los programas que ese paquete incluye son:

lsmod

Lista los módulos cargados, de cada uno presenta nombre, tamaño, cuenta de usos y lista de módulos que lo usan (es equivalente a **cat /proc/modules**).

rmmod ***`módulos`***

Descarga uno o más módulos cargados, mientras estos no estén siendo usados. Con la opción `-r` intenta descargar recursivamente módulos de los cuales el módulo especificado dependa. El comando **rmmod -a** descarga todos los módulos que no estén siendo usados.

insmod ***`módulo`*** **`[`*`opciones`*`]`**

Trata de cargar el módulo especificado. Pueden pasarse opciones específicas para el módulo, a continuación del nombre con la sintaxis `símbolo=`*`valor`* (los símbolos posibles dependen del módulo, pueden verse algunos en estas guías o en `/usr/share/modconf/descr.gz` — que es la ayuda presentada por **modconf** — o en último caso en las fuentes del módulo en los macros `MODULE_PARAM` y `MODULE_PARAM_DESCR`). Puede indicarse una ruta no estándar para buscar módulos estableciéndola en la variable MODPATH o en `/etc/modules.conf`. Dado que los módulos se enlazan directamente con el kernel, deben ser compilados para una versión precisa, con la opción `-f` puede evitarse el chequeo de versiones.

depmod

Como un módulo puede requerir otros, hay dependencias que deben respetarse al cargar y descargar módulos. **depmod** permite calcular tales dependencias entre varios módulos o entre todos los disponibles con la opción `-a` [18]. Por defecto **depmod -a** escribe las dependencias en el archivo

15. Versión tal como la reporta **uname -r**
16. Esta extensión es típica de *código objeto* que es código generado por un compilador a partir de un archivo fuente (en el caso de Linux el compilador es `gcc`). Puede "unirse" estáticamente con más código objeto empleando un encadenador (en el caso de Linux el encadenador es `ld`) o dinámicamente usando `/lib/ld.so` (ver Administración de programas) o en el caso de módulos con `insmod`.
17. El archivo de configuración de `modutils` puede variarse especificando uno diferente en la variable MODULECONF
18. La especificación de las rutas para buscar módulos puede hacerse con MODPATH o en `/etc/modules.conf` en las líneas de la forma **path[*`tipo`*]**

`/lib/modules/version/modules.emp` Cada línea de ese archivo tiene el nombre de un módulo seguido del caracter ':' y los módulos de los cuales depende, separados por espacios.

modprobe ***`módulo opciones`***

Emplea la información de dependencias generada por **depmod** e información de `/etc/modules.conf` para cargar el módulo especificado, cargando antes todos los módulos de los cuales dependa. Para especificar el módulo basta escribir el nombre (sin la ruta, ni la extensión `.o`) o uno de los alias definidos en `/etc/modutils/alias` (o en otro archivo del directorio `/etc/modutils`). Si hay líneas `pre-install` o `post-install` en `/etc/modules.conf`, **modprobe** puede ejecutar un comando antes y/o después de cargar el módulo. Como opciones para cargar el módulo usa prioritariamente las dadas en la línea de comandos y después las especificadas en líneas de la forma `options` *`módulo opciones`* en el archivo `/etc/modules.conf` [19]

Puede emplear estos programas para configurar sus módulos y puede hacer permanentes los cambios, agregando el módulo y las opciones en el archivo `/etc/modules` [20].

Para hacer más fácil la configuración de módulos, Debian ofrece las siguientes herramientas:

modconf

Para listar, cargar y descargar módulos con menús. Este programa muestra los módulos disponbiles en categorías y con ayudas sobre su uso y permite cargarlos o descargarlos del kernel, actualizando automáticamente los archivos `/etc/modules` y `/etc/modules.conf` (cambiando los archivos apropiados de `/etc/modutils`) para que los módulos configurados sean cargados automáticamente en el siguiente arranque. La información sobre los módulos disponibles la obtiene del directorio `/lib/modules`, los módulos cargados y sus parámetros los lee de `/etc/modutils` y `/etc/modules.conf` y la ayuda y la información interna de los archivos en `/usr/share/modules.conf` —**modconf** es un script para el intérprete de comandos.

update-modules

Actualiza el archivo `/etc/modules.conf` a partir de la información de los archivos del directorio `/etc/modutils` — en Debian no se edita directamente `/etc/modules.conf`.

Es posible que el kernel del que disponga no cuente con módulos que soporten ciertos dispositivos, en tal caso se puede:

1. buscar en Internet un módulo precompilado para la versión del kernel que tiene,
2. buscar fuentes de un módulo o
3. buscar una versión del kernel que incluya soporte para el dispositivo.

En el último caso las instrucciones de la siguiente sección le serán de utilidad.

---

19. El archivo `/etc/modules.conf` puede tener comentarios en líneas iniciadas con el caracter '#', puede tener líneas para: modificar las dependencias entre módulos (below, above, probe, probeall, depfile); para manejar las rutas donde buscar módulos (path, keep); para ejecutar comandos antes, en vez o después de instalar o descargar los módulos (pre-install, install, post-install, pre-remove, remove, post-remove); para especificar opciones por defecto para algunos módulos o **insmod** (options, insmod_opt); para crear alias o para definir variables o tener en cuenta sólo ciertas partes del archivo (alias, define, if, else, elseif, endif).
20. Durante el arranque las dependencias entre módulos son generadas automáticamente y los módulos especificados (junto con sus opciones) en el archivo `/etc/modules` son cargados.

5.2.1.2. Configuración del kernel

Además de cargar y descargar módulos, algunas características del kernel pueden ser modificadas mientras está en funcionamiento el sistema, bien escribiendo en algunos archivos del directorio `/proc`, o con el programa **sysctl**. Este programa con la opción `-a` presenta todas las variables modificables y su valor, entre otras encuentra variables que mantienen datos autodetectados en dispositivos (en particular del CDROM), datos de la interacción entre el kernel y el sistema de archivos (ver Sistema de archivos ext2), datos sobre los protocolos y dispositivos de redes, y detalles del kernel.

Para cambiar un valor se emplea:

```
/sbin/sysctl -w kernel.hostname=comp
```

con un nombre de variable en lugar de *`kernel.hostname`* y un valor adecuado en lugar de *`comp`*. Pueden hacerse cambios que se vuelven a tomar cada vez que el sistema inicia en el archivo `/etc/sysctl.conf`.

En cuanto a los módulos, las imagenes precompiladas del kernel producidas por Debian incluyen estáticamente varios controladores comunes y prácticamente los demás controladores los incluye como módulos. Por esto, si la versión del kernel que tiene soporta sus dispositivos y los servicios que desea, casi con seguridad usted NO necesita recompilar el kernel (puede confirmar si la imagen que tiene, soporta el controlador como módulo revisando en el directorio de módulos, o puede comprobar si éste se incluye estáticamente, efectuando el segundo paso para una instalación que se explica más adelante, usando como archivo de configuración el de la imagen de su kernel).

Eventualmente, si su dispositivo no es detectado automáticamente o el módulo apropiado no es cargado automáticamente [21], tendrá que pasar los paramétros apropiados de una de las dos siguientes formas:

Controlado estático

Pase los parámetros apropiados desde el cargador de arranque o configurelos de forma permanente con su cargardor de arranque (ver Parámetros desde el cargador de arranque).

Módulo (dinámico)

Pase los paramétros de configuración a **insmod/modprobe** o configurelos de forma permanente en `/etc/modules`, o en un archivo de `/etc/modutils` (vea la sección anterior sobre módulos).

Eventualmente puede encontrar y usar algún programa que le ayude a determinar los parámetros apropiados y/o a realizar las configuraciones (algunos se mencionan en las secciones de este capítulo), o tendrá que consultar los manuales del dispositivo o incluso probar varias opciones hasta determinarlos.

Si tras consultar esta documentación o la del kernel o cualquier otra más actualizada, determina que una versión del kernel diferente a la que tiene, soporta cierto dispositivo o servicio que requiere, el método más sencillo que puede intentar es instalar una imagén precompilada disponible en algún depósito de paquetes de Debian (el nombre es de la forma `kernel-image-`*`version`*). Como parte del archivo oficial de Debian encontrará imagenes recientes de las series 2.2.x y 2.0.x (por defecto Debian 2.2r5 emplea el kernel 2.2.19).

21. **kmod** es el programa que se encarga de cargar módulos automáticamente por demanda i.e. sin intervención del usuario

Fuera del depósito oficial encontrará paquetes para Debian 2.2 de imágenes de la versión 2.4.18 para diversos procesadores e instrucciones en: http://www.fs.tum.de/~bunk/kernel-24.html La instalación es bastante directa y automática (sólo debe tener en cuenta agregar una línea a `/etc/lilo.conf` para usar **initrd**).

En caso de requerir un kernel con una configuración muy especial (por ejemplo, que incluya estáticamente ciertos módulos o que excluya de la compilación módulos para hacer un kernel pequeño) o si requiere una versión para la que que no haya imagenes precompiladas de Debian, recomendamos instalar primero una versión precompilada con una versión cercana a la que desea instalar, para tomar la configuración de esta y modificarla antes de compilar la nueva versión. Además recomendamos esto porque las imagenes precompiladas actualizan otros programas indispensables para las nuevas versiones del kernel —en especial si cambia de la serie 2.2 a la serie 2.4 se deben actualizar : `gcc 2.91.66`, `make 3.77`, `binutils 2.9.1.0.25`, `util-linux 2.10o`, `modutils 2.4.0`, `e2fsprogs 1.19`, `pcmcia-cs 3.1.21`, `PPP 2.4.0`.

Para obtener las fuentes, puede recurrir a paquetes Debian para algunas versiones de las fuentes en depósitos oficiales (paquetes con nombres de la forma `kernel-sources-`*`version`*) y las fuentes de cualquier versión en el sitio de desarrollo del kernel (http://www.kernel.org). Cuando obtenga fuentes del kernel de Linux, tenga en cuenta que el segundo número de la versión sea par —si es impar se trata de una versión para desarrolladores pero NO para usuarios finales.

En esta sección presentamos un breve resumen de los pasos requeridos en una compilación de kernels versiones 2.2.x y 2.4.x, suponiendo que ya ha actualizado todas los programas que estas versiones requieren.

1. Limpieza Para limpiar posibles compilaciones anteriores emplee:

   ```
   make mrproper
   make clean
   ```

2. Preparación de la versión extra La versión de un kernel (que puede examinar con **uname -r**) se compone de 3 números separados por el caracter '.' eventualmente seguidos de una cadena (versión extra). Usted puede establecer esta cadena arbitrariamente y especificarla en el archivo `Makefile` de las fuentes (variable `EXTRAVERSION`), por ejemplo:

   ```
   EXTRAVERSION= -tobias
   ```

   Cuando compile, instale y le funcione un nuevo kernel, esa versión será la reportada por **uname -r**, los módulos que instale por defecto quedarán en `/lib/modules/`*`version`* y serán usados por `modutils` (ver Kernel y módulos).

   Si escoge una versión extra que coincida con una ya existente, los módulos anteriores serán remplazados durante la instalación. Recomendamos emplear una versión extra diferente a las que pueda haber, mantener el kernel original junto con sus módulos en disco y agregar una etiqueta y los datos requeridos por el cargador de arranque) —mientras se aprende, es muy normal producir kernels con configuraciones erradas que no podrán arrancar.

3. Configuración En general el paso más difícil es la configuración de las características que desea del kernel (dada la gran cantidad de preguntas y de términos técnicos, que esperamos estén explicados a lo largo de este capítulo). Para iniciar una configuración completa puede emplear **make menuconfig** que presentará menús y ayuda, tenga en cuenta que algunas características puede incluirlas

estáticamente en el kernel y otras como módulos (las características estáticas aumentan el tamaño). Alternativamente puede emplear **make xconfig** que presenta una interfaz gráfica o **make config** que presenta una serie de preguntas. El resultado de cualquier de estos métodos es un archivo `.config` con las variables de configuración y sus valores.

Otra posibilidad es que emplee la configuración de un kernel ya instalado y la modifique para acomodarla a sus necesidades. En Debian los archivos de configuración del kernel [22] están en el directorio `/boot` con nombres de la forma `config-`*`version_kernel`*. Copie uno de estos que corresponda a una versión cercana al que desea compilar, al directorio con las fuentes con el nombre `.config` después puede modificar la configuración de cualquiera de las siguientes formas: (1) ejecutando **make menuconfig** que leerá el archivo `.config` por defecto, (2) ejecutando **make xconfig** y eligiendo cargar ese archivo o (3) editando `.config` con un editor de texto y ejecutando después **make oldconfig**.

4. Compilación Primero se calculan interdepencias entre archivos fuente de acuerdo a la configuación con **make dep**, después se compila una imagen del kernel con **make bzImage** y finalmente se compilan módulos con **make modules**.
5. Instalación Es necesario instalar la imagen del kernel, los archivos auxiliares y módulos, configurar el cargador de arranque y eventualmente si está compilando un kernel con **initrd** (los kernels 2.4 precompildos de Debian por defecto lo usan) debe crear la imagen apropiada. Antes de instalar módulos. En este punto es recomendable que saque una copia de respaldo del directorio de módulos (en caso de que este remplazandolos por la versión extra que escogió). Para instalar emplee **make modules_install** que copiará todos los módulos y las dependencias entre ellos al directorio `/lib/modules/`*`version_kernel`*. A continuación se presenta un ejemplo que debe adaptar a la versión de su kernel:

```
cd /usr/src/kernel-source-2.2.21
make modules_install
cp arch/i386/boot/bzImage /boot/vmlinuz-2.2.21-tobias
ln -s /boot/vmlinuz-2.2.21-tobias /vmlinuz-2.2.21-tobias
cp System.map /boot/System.map-2.2.21-tobias
cp .config /boot/config-2.2.21-tobias
```

Después deberá editar la configuración de su cargador de arranque para que emplee la nueva imagen del kernel, por ejemplo puede agregarse una sección como:

```
image=/vmlinuz-2.2.21-tobias
root=/dev/hda2
label=Potato-tobias
read-only
append="apm=on"
```

Cambiando el dispositivo `root` por el apropiado, la etiqueta y de no requerirse soporte para APM quitando la última línea (ver BIOS y otras características)

Si está compilando un kernel con **initrd** [23] (por ejemplo los kernel 2.4 para Debian lo emplean por defecto), debe agregar a la misma sección en `/etc/lilo.conf`:

```
initrd=/initrd.img
```

22. Los archivos de configuración del kernel son dejados en `/boot` por los paquetes `kernel-image-`*`version`*.
23. Es una imagen de una partición raíz mínima que reside en memoria RAM, una imagen initrd suele tener controladores y configuraciones que el kernel debe configurar/realizar antes de emplear el dispositivo raiz real.

Después debe ejecutar **/sbin/lilo** para poner un nuevo cargador de arranque en el MBR o en el sector de arranque donde esté configurado.

Después de compilar un kernel, ejecute /**sbin**/**lilo** aún si no modifica `/etc/lilo.conf`

5.2.2. Lecturas recomendadas: Kernel y módulos

- Para manejar mejor las herramientas de módulos, puede consultar las páginas **man** de `modules.conf`, **insmod**, **modprobe**
- En caso de que esté interesado o que necesite más información sobre módulos puede consultar "*Introduction to Linux Kernel Modules*" en http://www.luv.asn.au/overheads/kernelmodules/
- Algunos módulos que se distribuyen separados del kernel requieren ser compilados. Puede encontrar ayuda de como hacerlo en: http://www.scyld.com/expert/modules.html
- Otro documento donde puede consultar el proceso de compilación está disponible en Internet en: http://structio.sourceforge.net/paz/actualiz-kernel.html

5.2.3. Ejercicios: Kernel y módulos

1. ¿Qué versión del kernel Linux tiene instalado su sistema? Ayuda: emplee el programa `kernelversion`

2. Revise los módulos que están cargados en su sistema, y trate de determinar que dispositivo controla o qué servicio presta cada uno.

3. Examine el archivo de configuración de X-Window, si desea experimentar con este, saque una copia y modifíquelo directamente o por medio del programa `XF86Setup` (si no tiene disponible `XF86Setup` puede intentar con `xf86config`).

4. Al cargar un módulo, **insmod** revisa que los símbolos del kernel que el módulo espera puedan ser resueltos. ¿Qué son estos símbolos? Describa también como buscó su respuesta.

5.3. Hardware básico y Linux

Indicadores de Logros:

- Puede abrir la torre de un PC, identificar sus componentes y manipularlos.

- Identifica y configura el hardware básico que soportan los kernels 2.2 y 2.4 en PCs típicos (cercanos a las suposiciones de la plataforma de referencia ver Plataforma de referencia).

5.3.1. Lectura: Hardware básico y Linux

En ocasiones cuando un componente de un computador falla o cuando se requiere instalar uno nuevo, un administrador puede requerir comprar uno compatible con Linux, instalarlo y configurarlo. Estas actividad en general no presentará problemas, mientras se informe bien.

En esta sección presentamos nociones básicas sobre el hardware que típicamente se encuentra en la placa base de un PC, así como su configuración en Linux. Varios dispositivos que no se presentan en esta sección se presentan en la siguiente (ver Dispositivos y Linux) o en el siguiente capítulo (ver Dispositivos para interconexión y Linux). Tenga en cuenta al leer esta sección que nos hemos concentrado en hardware cercano a la recomendación de la plataforma de referencia ver Plataforma de referencia.

5.3.1.1. Precauciones para manipular hardware

Para instalar una tarjeta adaptadora, memoria o diversos dispositivos, tendrá que destapar su computador. A continuación damos algunas recomendaciones:

- Consulte la documentación particular de su computador y guardela para referencia futura.
- Normalmente NO tendrá que abrir el monitor de su computador. Le recomendamos no hacerlo porque los niveles de voltaje de un monitor o un televisor (i.e. de un tubo de rayos catódicos CRT) son bastante elevados (>10000V), permanencen así durante un tiempo después de desconectado y si no toma las precauciones debidas puede sufrir una descarga eléctrica mortal.
- Antes de abrir el computador apague el equipo, así evitará dañar componentes, haciendo cortos accidentales y evitará recibir descargar eléctricas.
- Antes de abrirlo toque un elemento "tierra" [24], normalmente la carcaza de su computador si este tiene un polo a tierra. De esta forma puede sacar la carga eléctrica estática que pueda quedar en su cuerpo. Esto es importante, porque las cargas estáticas puede dañar ciertos dispositivos. Mantenga también los dispositivos en su envoltura antiestática mientras no los use y trate de manipularlos por sus esquinas sin tocar sus circuitos integrados.
- Si por algún motivo ve que el componente no está en óptimas condiciones, no le suministre energía ni lo instale en su computador.

5.3.1.2. Generalidades sobre la placa base

Cuando destape un computador personal verá una lámina sobre la que encontrará entre otros: circuitos integrados (entre ellos el procesador y el BIOS), espacios para la memoria, puertos para discos, ranuras para otras tarjetas, jumpers para configurarla. Esa lámina se llama placa base o tarjeta madre [25]. Separados de la placa, aunque conectados por medio de cables, eventualmente podrá ver unidades de disquete, de CD-ROM, discos duros, una fuente de potencia (por la que se suministra corriente electríca alterna —en Colombia 110V 60Hz— y que convierte para uso del computador a corriente directa).

24. Cualquier elemento por el que usted pueda descargarse pasando la carga estática a la tierra
25. Del inglés *mother board*.

Con el fin de facilitar el cambio y actualización de los dispositivos, una placa base tiene ranuras de uno o más tipos de buses (e.g ISA, PCI, AGP) para conectar en ellos tarjetas que sean o que controlen dispositivos (e.g. tarjeta de sonido, tarjeta de video) o que sirvan como adaptadores para cierto tipo de puertos (e.g adaptadores para dispositivos SCSI), puede también tener integrados controladores y puertos para cierto tipo de dispositivos (e.g IDE, teclado, ratón, puerto paralelo, puerto serial), y eventualmente puede tener integrados dispositivos (e.g. tarjeta de sonido).

5.3.1.3. Generalidades sobre dispositivos

Los dispositivos que pueden conectarse a la placa base por puertos, ranuras o por intermedio de tarjetas adaptadoras tienen direcciones de entrada salida (también llamadas puertos E/S [26]) que les permiten intercambiar datos con el procesador, pueden emplear además unos cables especiales del procesador llamados líneas IRQ (*Interruption Request*) para solicitar atención especial (por ejemplo cuando un dato llega a un modem, este puede activar la línea IRQ que tenga asignada para solicitar procesamiento del dato), eventualmente pueden emplear canales DMA (*Direct Memory Access*) para transmitir información directamente a la memoria sin intervención del procesador y eventualmente si el dispositivo emplea directamente la memoria puede requerir además una dirección base en memoria compartida [27] para comunicarse con los programas.

Normalmente los recursos citados sólo pueden ser empleados por un dispositivo a la vez, si dos emplean el mismo recurso pueden entrar en conflicto y no funcionar. Dado que sólo hay 16 líneas IRQ, 65535 direcciones de E/S, 8 canales de DMA (los primeros 4 de 8 bits y los otros 4 de 16 bits) y la memoria compartida es memoria RAM de su computador (límitada por la cantidad que tenga) debe configurar nuevos dispositivos teniendo en cuenta los que ya hay (algunos dispositivos y controladores permiten compartir IRQ con otros). Para conocer la asignación de recursos puede examinar: `/proc/interrupts` que lista las interrupciones en uso, `/proc/dma` que lista DMAs en uso, `/proc/ioports` que presenta los puertos E/S asignados (tenga en cuenta que normalmente un dispositivo requiere un rango de puertos consecutivos, ningún puerto en ese rango debe estar ocupado —por ejemplo una tarjeta de red Ethernet requiere un puerto base y los 31 puertos siguientes), `/proc/iomem` presenta los rangos de memoria compartida con dispositivos.

El estado de los puertos puede verse o modificarse por intermedio del archivo `/dev/port`.

A nivel de sistema operativo, Linux presenta los dispositivos que puede manejar como "archivos" del directorio `/dev` (ver Ubicación de archivos y directorios), puede consultar la lista de los dispositivos reconocidos con:

```
cat /proc/devices
```

5.3.1.4. Procesador

Es un circuito integrado que realiza algunos cálculos y que controla memoria y dispositivos. Para dar orden a su operación emplea un reloj para ejecutar una instrucción por cada pulsación. Recibe toda la

26. Del inglés *I/O adress* o *I/O ports*.
27. Del inglés *shared memory*.

información por diversos cables (por ejemplo el bus de datos de la arquitectura i386 [28] tiene 32 cables) codificada en cada cable con dos níveles de voltaje: alto y bajo [29]. La velocidad del reloj —que determina la velocidad del procesador— se mide en Hertz [30].

Un kernel compilado para un Intel 386 podría usarse en sus procesadores descendientes, clones o compatibles, aunque el kernel soporta de manera particular algunos de tales procesadores. La serie 2.2 del kernel soporta 80386DX/DXL/SL/SLC/SX [31] de Intel o Cyrix, 80486DX/DX2/DX4 de AMD o Cyrix o IBM o Intel, AMD K5, Intel 5x86/6x86/6x86MX, Intel Pentium Classic, Intel Pentium MMX, AMD K6-3D, Cyrix/VIA, CyrixIII, Intel Pentium Pro, Intel Pentium II, 6x86MX/MII de Cyrix/IBM/National Semiconductor. La serie 2.4 tiene soporte especial además para: Intel Pentium III, Intel Pentium 4, AMD K7 (Athlon/Duron/Thunderbird), Crusoe de Tansmeta, Winchip C6/2/2A y soporta varios procesadores en una misma placa base (SMP - Symmetric Multi Processing). No soporta 80286 o procesadores previos a ese.

En los depósito de Debian hay kernels ya compilados con soporte especial para algunos de estos procesadores. Sus nombres son de la forma `kernel-image-`*`version-procesador`* (ver Configuración del kernel).

Puede examinar información sobre el procesador que tiene con:

```
cat /proc/cpuinfo
```

5.3.1.5. Memoria RAM

La memoria RAM (*Random Access Memory*) permite almacenar información mientras el computador está encendido. La cantidad de información que puede almacenar se mide en bytes [32].

La memoria tiene una velocidad de acceso, menor a la velocidad del procesador, que está limitada por el tipo de memoria, el procesador y el chipset [33] de la placa base.

Una placa base normalmente tendrá módulos de memoria [34]. que pueden cambiarse. Los tipos de conectores —que dependen del procesador y de la placa base— pueden ser SIMM (Single in-line memory module) de 32 bits y 72 pines o DIMM (Double in-line memory module) de 64 bits y 168 pines.

28. Esta arquitectura se refiere a PCs basados en procesadores Intel 80386 sus descendientes, clones y compatibles. Linux soporta otras arquitecutras: Alpha, ARM, IA-64, MIPS, MIPS64, PARISC, PPC, S390, Sparc y Sparc64. Hay distribuciones Debian para las arquitecturas Alpha, ARM, Intel x86, Motorola 680x0 (Atari, Amiga, VME y Macintosh), PowerPC (Macintosh PowerMac) y SPARC (Sun4c, Sun4m y Sun4u) y en preparación hay distribucions Debian para UltraSparc, MIPS, HP PA, IA-64, S/390 y SuperH.
29. Un bit es uno de los niveles de voltaje alto o bajo, puede representarse como un 1 o un 0.
30. Hertz: ciclos por segundo. En este contexto se usa con los prefijos: Mega (un millon) y Giga (mil millones)
31. Linux ofrece emulación de un coprocesador matemático para ciertos procesadores como 80386SX y 80486SX o para procesadores con coprocesador defectuoso.
32. Un byte puede considerarse como un caracter (corresponde a 8 bits). En el contexto de memoria se se usa con los siguientes prefijos: K - Kilo (1024), M - Mega (1048576), G - Giga (1073741824fff) y en vez de byte suele escribirse B. Note que los prefijos para referirse a cantidad de memoria NO corresponden al significado usual (en este caso son potencias de 2 (1024=2^10) y no a potencias de 10)
33. *Chipset*: controla el flujo de información hacia el procesador, por ejemplo desde buses y memoria. Algunos tipos de chipsets y sus velocidades típicas son Slot 1 AGP 66Mhz/100Mhz, Socket 7 AGP (66/100Mhz), Socket 7 PCI (66/75/83Mhz), Socket A 133/266Mhz.
34. Otra clasificación posible para memorias es por la tecnología que emplean para almacenar los datos, algunos tipos según esta clasificación son DRAM (Dynamic Random Access Memory), SDRAM (Static DRAM), ambas soportadas por Linux.

Puede examinar la cantidad de memoria de su computador con el programa `free` o con:

```
cat /proc/meminfo
```

que también presentará información sobre memoria virtual, es decir espacios del disco duro usados para hacer las veces de memoria RAM (estos espacios del disco son particiones especiales ver Discos duros). Si nota que Linux no llega a reconocer automáticamente toda la memoria que tiene instalada puede especificarla manualmente pasando un parámetro como `mem=128M` desde el cargador de arranque (ver Parámetros desde el cargador de arranque).

El contenido de la memoria física puede verse o modificarse con el archivo `/dev/mem` (c,1,1)[35] y la memoria virtual con `/dev/kmem` (c,1,2).

Aviso

No modificar directamente `/dev/mem` ni `/dev/kmem`.

Aunque hay distribuciones más livianas que las sugeridas en estas guías que podrían instalarse con 4MB (e.g Tiny / (http://tiny.seul.org/), SmallLinux http://www.superant.com/smalllinux/), nuestras recomendaciones de software requieren mínimo 32MB en RAM. La serie 2.2 del kernel soporta máximo 2GB, la serie 2.4 soporta máximo 64GB –Debian Potato por defecto soporta máximo 1GB y mínimo 12MB.

5.3.1.6. Buses

Para que un dispositivo pueda ser usado por el procesador es necesario que ambos estén eléctricamente conectados y que se comuniquen empleando un "lenguaje" (protocolo) común. A un tipo de conexión junto con un protocolo para realizar esta comunicación se le llama bus.

Hay diversos tipos de buses, cada uno tiene un tamaño (medido en bits) que índica cuanta información puede transmitirse simultáneamente (8, 16, 32 o 64 bits) y una velocidad de reloj diferente a la velocidad del procesador (e.g 60Mhz, 100Mhz). A continuación presentamos información sobre los tipos de buses más comunes en computadores de escritorio, todos los cuales son soportados por Linux:

5.3.1.6.1. ISA (*International Standard Architecture*)

Aunque este tipo de bus de 8Mhz es de 1980, hay placas base recientes con ranuras que lo soportan y hay una amplia gama de dispositivos tanto de 8 como de 16 bits (tarjetas de red, tarjetas SCSI, controladoras IDE, tarjetas de video, etc). Todas las versiones del kernel soportan este bus y gran cantidad de dispositivos ISA (algunos se presentan en otras secciones).

5.3.1.6.1.1. Configuración de hardware

Algunos de estos dispositivos emplean recursos (puertos E/S, IRQ, DMA) con valores prefijados por quien lo instala antes de que el computador sea encendido. Normalmente cada dispositivo podrá ser configurado para emplear diversos recursos (por ejemplo diversas líneas IRQ) y así evitar conflictos en el uso de estos recursos con otros dispositivos. Algunos dispositivos se pueden configurar con jumpers en el

35. En diversas partes de estas guías usamos la notación (c,*x*,*y*) para denotar el número mayor (*x*) y menor (*y*) de un dispositivo por caracteres. Si el dispositivo es por bloques empleamos `b` en lugar de `c`.

hardware (de acuerdo a las instrucciones que acompañan al dispositivo y mientras el computador está apagado) o bien por software —normalmente con un programa para DOS que guarda la configuración que usted elija en una memoria EPROM del dispositivo. Hay algunos programas para Linux que permiten hacer estas configuraciones para ciertos dispositivos, puede también emplearse software para DOS con una versión de libre redistribución de ese sistema como FreeDOS http://www.freedos.org).

Si el dispositivo usa una IRQ y su BIOS soporta Plug and Play, debe también entrar a la configuración del BIOS y marcar ese IRQ como "Legacy-ISA" en lugar de "PnP".

5.3.1.6.1.2. Configuración del sistema operativo

Después de configurar el dispositivo y tras iniciar el sistema operativo debe emplear el controlador apropiado para manejarlo y pasar parámetros acordes con la configuración que hizo, por ejemplo si configura una tarjeta de red ISA compatible NE2000 para usar la IRQ 12 y la dirección base 0x200 y tiene el módulo **ne**, debe configurarlo por ejemplo con **modconf** (verKernel y módulos) o con:

```
modprobe ne irq=12,io=0x200
```

5.3.1.6.1.3. Dispositivos ISA Plug and Play (PnP)

Algunos dispositivos para puertos ISA denominados *Plug and Play* pueden ser configurados después de que el computador ha iniciado, bien por la BIOS o bien por el sistema operativo. Estos dispositivos son manejados automáticamente por el kernel 2.4 y pueden configurarse manualmente con las herramientas **isapnp** tanto en el kernel 2.2 como en el 2.4. Si usted lo prefiere o el kernel no logra configurar alguno de estos dispositivos tiene estas alternativas:

1. Algunos de estos dispositivos permiten activar/desactivar el funcionamiento como Plug and Play. Si ese es el caso desactívelo y configure el dispositivo como se explicó para dispositivos ISA. Eventualmente también tendrá que desactivar la posible opción de su BIOS: PnP OS.
2.

   Emplee las utilidades **isapnptools** (paquete con el mismo nombre) para asignar una configuración que no entre en conflicto con la de otros dispositivos y que en caso de tener otro sistema operativo en el mismo disco debe coincidir con esa (en tal caso revise primero la configuración que el dispositivo tiene en el otro sistema operativo). Para generar un archivo con las posibles configuraciones de los dispositivos ISA PnP, ejecute el programa **pnpdump** redireccionando la sálida e.g

   ```
   pnpdump > ~/pnp.txt
   ```

   después edite el archivo `~/pnp.txt` para escoger una configuración adecuada que no entre en conflicto y pruebela ejecutando:

   ```
   isapnp ~/pnp.txt
   ```

   El archivo generado consta de líneas con comentarios (iniciadas con el caracter '#') e instrucciones que indican como debe realizarse la configuración general y la configuración particular de cada dispositivo. Cada instrucción debe estar entre paréntesis con sus posibles parámetros. Las primeras instrucciones (configuación general) indican como se realiza el asilamiento de dispositivos ISA PnP, si debe ser realizado por **isapnp** con la instrucción `(ISOLATE)` (esto es necesario con algunas BIOS que no soportan ISA PnP), por el contrario si el aislamiento lo hace el BIOS el archivo debe comenzar con algo como:

```
(READPORT 0x3bb)
(CSN 2)
(IDENTIFY *)
```

A continuación debe haber una sección para cada dispositivo ISA PnP, que indica como se configura, por ejemplo una sección típica (tras quitar comentarios) para configurar una tarjeta de sonido Sound Blaster es:

```
(CONFIGURE CTL0044/1132685 (LD 0
(INT 0 (IRQ 5 (MODE +E)))
(DMA 0 (CHANNEL 1))
(DMA 1 (CHANNEL 5))
(IO 0 (BASE 0x220))
(IO 1 (BASE 0x330))
(IO 2 (BASE 0x388))
(ACT Y)))"
```

donde CTL0044/1132685 es la identificación del dispositivo, LD 0 es un número asignado al dispositivo por Linux (logical device) y las siguientes líneas especifican que el usuario eligió para el dispositivo: 1 línea IRQ (5), 2 canales DMA (1 y 5) y 3 direcciones E/S base (0x220 [36], 0x330 y 0x388).

> La línea con `(ACT Y)` es indispensable para que la configuración sea tenida en cuenta. Después de todas las secciones de dispositivos el archivo debe terminar con la instrucción `(WAITFORKEY)`.

Una vez logre configurar el dispositivo —lo habrá logrado si al ejecutar **isapnp** y al cargar el módulo no se presentan errores— puede hacer el cambio permanente guardando la configuración en el archivo `/etc/isapnp.conf` que tiene la sintaxis de los archivos generados por `pnpdump` y que es cargado automáticamente durante la inicialización del sistema (puede ver el script `/etc/init.d/isapnp` que es usado en `/etc/rcS.d` ver Inicialización del sistema). Además de `/etc/isapnp.conf`, el programa **isapnp** emplea el archivo `/etc/isapnp.gone` en el que se pueden especificar recursos que NO están disponibles para dispositivos Plug and Play.

5.3.1.6.2. MCA o Microcanal

Este bus de 32 bits que opera a 12Mhz fue introducido por IBM en 1987 para su línea de computadores PS/2, es usado por estos y por algunos computadores portátiles. El kernel lo soporta con la variable de configuración CONFIG_MCA (vr Kernel y módulos). De acuerdo a la documentación del kernel (Documentation/mca) cuando se usa puede examinarse en `/proc/mca`: `pos` lista de registros POS, `slot[1-8]` información sobre el adaptador de una ranura, `video` información sobre video integrado,

36. Encontrará en varias partes la notación 0x*número* para referenciar un número escrito en base 16 –es la convención del lenguaje C.

`scsi` información sobre SCSI integrado, `machine` información sobre la máquina. Hay más bien pocos controladores Linux para este tipo de bus, así como también hay relativamente pocos dispositivos que lo soporten. Los dispositivos se configuran por software (ver sección ISA) y los controladores recibien como parámetros los recursos usados.

5.3.1.6.3. *EISA (Enhanced Industry Standard Architecture*)

Esta es una extensión de 32 bits al sistema ISA [37] introducida en 1988, que puede funcionar a 20Mhz. Este bus y muchos dispositivos EISA son soportados por Linux (ver sección sobre el tipo de dispositivo particular).

5.3.1.6.4. VESA o VL-Bus o VLB (*Video Electronic Standard*)

En su versión 1.2, se trata de un bus de 32bits, 33Mhz atado al procesador 486, usado por algunas tarjetas graficadoras producidas especialmente entre 1992 y 1994. La versión 2.0 de VESA es empleada por algunas tarjetas graficadoras (como ATI Video Expression, Matrox Millennium G200, Riva 128 y otras listadas en Framebuffer-HOWTO), en particular permite el uso de framebuffer desde Linux con el controlador `vesafb` (en particular si se activa soporte para dispositivos framebuffer VESA en el kernel y si se cuenta con una tarjeta VESA 2.0, durante el arranque aparecerá el logo de Linux).

5.3.1.6.5. PCI (*Peripheral Component Interconnect*) o *Local Bus*

Este es un diseño de bus de 1993 extensible [38] de 32bits que opera a 66Mhz y que soporta ISA, las líneas IRQ de estos dispositivos son asignadas automáticamente durante el arranque por la BIOS (no hay forma de asignar manualmente una IRQ a un dispositivo PCI). El programa de configuración de algunas BIOS permite evitar que ciertas IRQs sean usadas en la asignación automática (eventualmente en una menú Plug and Play), de forma que esas pueden reservarse para otros dispositivos ISA/EISA.

Los dispositivos PCI tienen una identificación única que permite clasificar el tipo de dispositivo y el fabricante [39]. Estas identificaciones suelen ser presentadas por algunas BIOS durante el arranque del sistema, en Linux puede examinarse junto con los recursos que emplean con:

```
cat /proc/pci
```

o con **lspci**. En Internet puede consultarse la lista de vendedores y dispositivos en: http://www.yourvote.com/pci/ o en caso de sólo conocer el número FCC [40] de un dispositivo puede consultar el vendedor en: http://www.fcc.gov/oet/fccid/

El kernel 2.2.19 y la serie 2.4, durante el arranque identifican también los dispositivos PCI conectados y tratan de cargar los controladores apropiados, normalmente sin intervención del usuario. Si el módulo apropiado para un dispositivo PCI no logra ser cargado automáticamente durante el arranque, puede cargarlo después por ejemplo con **modconf** (normalmente no requerirá parámetros para dispositivos PCI ver Módulos).

37. Una ranura EISA en una placa base soporta también tarjetas ISA.
38. Es extensible porque pueden usarse puentes para conectar otros buses PCI o Cardbus o USB.
39. Las identificaciones conocidas por Linux pueden examinarse en las fuentes del kernel en el archivo `include/linux/pci.h`.
40. FCC es sigla de *Federal Communications Commission* quienes vigilan que los dispositivos empleen eficientemente el espectro electromagéntico.

5.3.1.6.6. USB (*Universal Serial Bus*)

Este tipo de bus que permite tasas de transmisión de 12 Mbits y hasta 127 dispositivos, no requiere la instalación de tarjetas, los dispositivos pueden conectarse y desconectarse mientras el computador está encendido, se conectan a puertos de la placa base o tarjetas adaptadoras o a otros dispositivos USB que actuan como concentradores [41]. Los conectores se distinguen por ser rectangulares, cada dispositivo puede separarse de un concentrador/puerto USB hasta 5 metros.

Las versiones del kernel posteriores a la 2.2.18 soportan dispositivos USB (`CONFIG_USB`, `usbcore`), y dos posibles protocolos empleados por el chipset de la placa base:

UHCI (*Universal Host Controller Interface*)

> Desarrollada por Intel y presente en placas base con chipsets PCI de Intel como 430TX/44FX/440LX/440BX/i810/i820 o en chipsets PCI de VIA como VIA VP2, VP3, MVP3, Apollo Pro /II/133). En el kernel se configura con `CONFIG_USB_UHCI`, módulo `usb-uhci` o bien con el controlador alterno `CONFIG_USB_UHCI_ALT`, `uhci`.

OHCI (*Open Host Controller Inteface*)

> Desarrollado por Compaq/Microsoft/National los controladores USB en la mayoría de arquitecturas con chipsets no Intel (como SIS o ALI) cumplen este estándar. `CONFIG_USB_OHCI`, `usb-ohci`.

De acuerdo a documentación del kernel (Documentation/usb/scanner.txt), los chipsets que soportan UHCI al ejecutar **lspci -V** presentan información de la forma:

```
USB Controller: ......
Flags: .....
I/O ports at ....
```

mientras que los que soporta OHCI dicen:

```
USB Controller: ......
Flags: .....
Memory at ...
```

El controlador básico presenta [42] los dispositivos detectados en `/proc/bus/usb/devices`, los controladores usados en `/proc/bus/usb/001/002` (cambiando 001 por el número de bus y 002 por el número de dispositivo). Para verlos debe montarse antes el directorio con **mount -t usbdevfs none /proc/bus/usb** (ver Montaje y desmontaje de sistemas de archivos).

Los dispositivos USB para interacción con humanos (e.g ratón, teclado, joystick, tarjeta digitalizadora), son manejados con el controlador HID - Human Interface Device (`CONFIG_USB_HID`, `hid`) —si su computador tiene estos dispositivos puede requerir incluir estáticamente este controlador en el kernel,

41. Concentrador: del inglés *hub*
42. Es posible montar `/proc/bus/usb` sólo si se configuró con `CONFIG_USB_DEVICEFS`

para eso debe escoger también durante la configuración soporte básico de entrada (Input core support, `CONFIG_INPUT`).

Hay controladores genéricos para audio (`CONFIG_USB_AUDIO`, `audio`), scanner (`CONFIG_USB_SCANNER`, `scanner`), impresora (`CONFIG_USB_PRINET`, `printer`), modems y dispositivos ISDN (`CONFIG_USB_ACM`, `acm`), dispositivos de almacenamiento (`CONFIG_USB_STORAGE`, `usb-storage`), dispositivos Bluetooth —comunicaciones inalámbricas— (`CONFIG_USB_BLUETOOTH`, `bluetooth`).

Hay soporte para varios adaptadores de USB a puerto serial (`CONFIG_USB_SERIAL`, `usbserial`), y el kernel 2.4 soporta un adaptador de puerto paralelo a USB.

5.3.1.6.7. AGP (*Accelerated Graphics Port*)

Fue introducido en 1997, es un bus especial para tarjetas graficadoras 3-D, que ofrece mayor velocidad que los puertos PCI. Es soportado por versiones recientes del kernel 2.2 (está en el 2.2.19) y por la serie 2.4. `CONFIG_AGP`, módulo `agpgart`. Es útil con algunas tarjetas de vídeo que emplean DRI o GLX de X-Window. Soporta los chipsets: Intel 440LX/BX/GX Intel 815/830M/840/850 (`CONFIG_AGP_INTEL`), Intel 810/815/830M (`CONFIG_AGP_i810`), VIA MPV3/Apollo (`CONFIG_AGP_VIA`), AMD Irongate 761/762 (`CONFIG_AGP_AMD`), SIS (`CONFIG_AGP_SIS`), ALI M1541/M1621/M1631/M1641/M1647/M1651 (`CONFIG_AGP_ALI`), Serverwork (`CONFIG_AGP_SWORKS`).

5.3.1.6.8. PCMCIA (*Personal Computer Memory Card International Association*)

Son puertos disponibles en portátiles desde 1990, para conectar tarjetas PCMCIA como Modems y tarjeta de red, los hay de 16 bits y de 32 bits (Cardbus). Los dispositivos PCMCIA pueden conectarse y descontarse mientras el computador está encendido y son reconfigurados automáticamente por las herramientas PCMCIA complementarias al kernel, paquete `pcmcia-cs`.

5.3.1.6.9. Otros tipos de buses

Linux también soporta otros tipos de buses no tan comunes como: I2O (http://www.i2osig.org), IrDA (Infrarojos), I2C.

5.3.1.7. Interfaz y dispositivos IDE

La placa basa puede tener tener varias interfaces IDE [43] (usualmente 2: `ide0` e `ide1`), cada una de las cuales puede soportar hasta dos dispositivos con una misma correa (uno de los dos dispositivos es llamado maestro y el otro esclavo [44]). Las interfaces IDE usan 7 puertos E/S y una línea IRQ (e.g para ide0 el puerto E/S base por defecto es 0x1f0 y la línea IRQ es la 14, para ide1 0x170 y 15), que de no ser

43. IDE es sigla de *Integrated Drive Electronics*, recibe este nombre porque los dispositivos IDE (discos duros, CD-ROMs) incluyen hardware controlador —por eso pueden conectarse casi directamente a un bus ISA de un PC. Las interfaces que usan están especificadas en los estándares ATA y ATA-2 (Attachment interfaces).
44. Si se usa un disco (o un CDROM) por interfaz debe configurarse como maestro o *single*, si se usan dos discos (empleando una correa con tres conectores) uno de los discos debe configurarse como maestro y el otro como esclavo. La configuración se hace empleando jumpers ubicados en los discos —debe consultar el manual para determinar la ubicación de los jumpers.

estándar deben especificarse como opciones para el kernel pasándolas desde el cargador de arranque e.g `ide1=0x1f0,0x3f6,10` (Parametros cargador de arranque en ver Apagado y encendido de una red).

La versión 2.2 del kernel soporta hasta 6 interfaces IDE, cada una con dos discos y cada disco máximo con 63 particiones, la versión 2.4 soporta hasta 10 interfaces IDE.

El controlador de Linux para interfaces IDE las reconoce automáticamente y las maneja directamente (sin intermediación de la BIOS) [45], pero durante el arranque LILO se apoya en la BIOS para usarlas y en particular para usar los discos duros —por eso al menos con BIOS no tan recientes, es indispensable que la partición `/boot` (o `/` si no se cuenta con una partición propia apara /boot) esté en los primeros 1024 cilindros, que es el límite soportado por diversas BIOS [46].

Los dispositivos asociados con los discos de las dos primeras interfaces IDE son:

`/dev/hda`

: Primario maestro (b, mayor 3, menor 0)

`/dev/hdb`

: Primario esclavo (mayor 3, menor 64)

`/dev/hdc`

: Secundario maestro (mayor 3, menor 0)

`/dev/hdd`

: Secundario esclavo (mayor 3, menor 64)

Podrá emplearlos para referenciar discos duros, CDROMs o los dispositivos que conecte a cada interfaz.

De acuerdo al *Hardware-HOWTO* las siguientes interfaces IDE son soportadas: CMD-640 (soporta interfaces defectuosas), RZ1000 (soporta esta interfaz defectuosa), AEC62XX, ALI M15x3, AMD Viper, CY82C693, Cyrix CS5530 MediaGX, HPT34X, HPT366, Intel PIIXn, NS87415, OPTi 82C621, Promise PDC20246/PDC20262/PDC20267, ServerWorks OSB4, SiS5513, SLC90E66, Tekram TRM290, VIA82CXXX, DTC 2278D, FGI/Holtek HT-6560B VLB, Triton I, Triton II, ALI M14xx, Promise DC4030, QDI QD6580, UMC 8672.

Cuando se compila el kernel las variables de configuración relacionadas con interfaces IDE son: `CONFIG_BLK_DEV_IDE` soporte para interfaz IDE, `CONFIG_BLK_DEV_IDEDISK` soporte para discos IDE ATA-2 (módulo `ide-disk`), `CONFIG_BLK_DEV_IDECD` soporte para CDROM IDE ATAPI (módulo `ide-cd`), `CONFIG_BLK_DEV_IDETAPE` soporte para unidades de cinta ATAPI (módulo `ide-tape`), `CONFIG_BLK_DEV_IDEFLOPPY` floppy IDE con protocolo ATAPI (módulo `ide-floppy`). `CONFIG_BLK_DEV_IDESCSI` permite emular con un dispositivo IDE uno SCSI, esto por ejemplo es útil con quemadoras de CD-ROM IDE (módulo `ide-scsi`, ver Quemadora de CD).

45. De acuerdo a documentación de la configuración del kernel, soporta incluso dos interfaces IDE defectuosas y soluciona los errores: CMD650 y RZ1000

46. De acuerdo a http://freshmeat.net/projects/lilo/?topic_id=139, las versiones de LILO posteriores a 21-3 con BIOS recientes (posteriores a 1998), no tienen esta limitación.

Puede examinar el estado de las interfaces IDE en el directorio `/proc/ide`. Durante el arranque, aunque las interfaces son detectadas automáticamente puede especificar algunos recursos manualmente pasandolos desde el cargador de arranque, por ejemplo: `ide0=0x1f0,0x3f6,14` [47].

5.3.1.8. Adaptadores y dispositivos SCSI

SCSI es un protocolo empleado por algunos discos duros, unidades de cinta, unidades de CDROMs, scanners, quemadoras de CD y otros dispositivos. Para convertir ese protocolo al de un bus de una placa base se requiere una tarjeta adaptadora (las hay para los diversos tipos de buses).

De acuerdo al *SCSI-2.4-HOWTO*, para emplear dispositivos SCSI en Linux deben usarse 3 tipos de módulos:

Controlador general SCSI

Variable `CONFIG_SCSI` en kernel; módulo `scsi_mod`.

Controlador de alto nivel de acuerdo al tipo de dispositivo

Disco duro

Variable CONFIG_BLK_DEV_SD; módulo `sd_mod`; dispositivos: primer disco `/dev/sda` (mayor 8 por bloques, menor 0, la primera partición tiene menor 1 `/dev/sda1`, la segunda 2 `/deb/sda2` y así sucesivamente máximo hasta 15), segundo disco `/dev/sdb` (menor 16), tercero `/dev/sdc` y así sucesivamente máximo hasta 16 discos (empleando los mayores 65 - 71 y un esquema análogo pueden instalarse hasta 128 discos SCSI).

Unidad de cinta

Variable CONFIG_CHR_DEV_ST; módulo `st`; parámetros: `buffer_kbs=32` tamaño en KB del buffer por defecto , `write_threshold_kbs=30` umbral para escritura asincrona, `max_buffers=4` máximo de buffers localizados durante inicialización, `max_sg_segs=4` máximo número de segmentos de dispersión/agrupación; una vez configurado pueden emplearse los dispositivos (con mayor 9): `/dev/st0` (menor 0) modo 0, `/dev/st0l` (menor 32) modo 1, `/dev/st0m` (menor 64) modo 2, `/dev/st0a` (menor 96) modo 3, `/dev/nst0` (menor 128) modo 0 sin retroceso, `/dev/nst0l` (menor 160) modo 1 sin retroceso, `/dev/nst0a` (menor 224) modo 3 sin retroceso (Una segunda unidad de cinta tiene nombres análgos con 1 en lugar de 0, los números menores son los mismos más 1, una tercera tendría nombres con 2 y se sumaría 2 a cada menor, así sucesivamente hasta 32). Las unidades OnStream SC-x0 SCSI en vez de `st` requieren el módulo `osst`; variable de configuración `CONFIG_CHR_DEV_OSST`; recibe los mismos parámetros de `st` excepto `xa_test`; usa el dispositivo `/dev/osst0` (mayor 206) de forma análoga y con los mísmos números menores de `/dev/st0`.

47. Los recursos empleados por los puertos IDE normalment son: `ide0=0x1f0,0x3f6,14`, `ide1=0x170,0x376,15`, `ide2=0x1e8,0x3ee,11`, `ide3=0x168,0x36e,10`.

CD-ROM

Variable `CONFIG_BLK_DEV_SR`; módulo `sr`; como parámetro puede recibir `xa_test=1` para activar chequeo del formato de los sectores (produce fallas en algunos sistemas); dispositivos con mayor 11 `/dev/scd0` (menor 0) para la primera unidad, `/dev/scd1` (menor 1) para la segunda y así sucesivamente hasta 255.

Controladores genéricos SCSI

Por cada unidad de CDROM, unidad de cinta o disco SCSI se crea un dispositivo genérico `sg`, así como para los demás dispositivos SCSI (e.g quemadoras de CD, Scanners) y pseudo-dispositivos (como el producido por `ide-scsi`. En el caso de otros dispositivos debe emplear algún programa auxiliar para manejarlo (como `cdrecord`, `cdrao` en el caso de quemadoras de CD —ver Quemadora de CD— o Sane para Scanners). Variable `CONFIG_CHR_DEV_SG`; módulo `sg`; los dispositivos tienen número mayor 21 son: `/dev/sg0` para el primer dispositivo genérico (menor 0), `/dev/sg1` para el segundo (menor 1) y así sucesivamente hasta 255.

Controlador de bajo nivel para la tarjeta SCSI

El kernel 2.2.19 soporta las siguientes tarjetas (`entre paréntesis se indica el nombre del módulo`): 3ware Hardware ATA-RAID, 7000FASST (`wd700`), ACARD 870U/W (`atp870u`), AHA-1510/1520/1522/2825 AVA-1505 (`aha152x`), AHA-1542 (`aha1542`), AHA-1740 (`aha1740`), tarjetas basadas en el controlador aic777x y aic78xx de Adptec (`excepto 7810`) como 2902, 2910, 293x, 294x, 394x, 3985, 274x, 284x (`aic7xxx`), todos los controladores de AdvanSys (`advansys`), controlador ISA Always IN2000 (`in2000`), adaptadores AM53/79C974 (`am53c974`), adaptador BusLogic (`buslogic`), canal de fibra óptica Compaq (`cqpfcts`), adaptador DTC 3180/3280 (`dtc`), adaptadores que soporten EATA-DMA como DPT PM2011B, PM2021A, PM2012A, PM2021B, PM2022A/9X, PM2122A/9X, PM2322A/PX, PM3021, PM3222, PM3224,PM3334 (`eata`), adaptadores que soporten EATA-PIO como DPT PM2001 y PM2012A (`eata_pio`), adaptores Future Domain de 16 bits como TMC-1660/1680, TMC-1650/1670, TMC-3260, TMC-1610M/MER/MEX (`fdomain`), adaptadores Future Domain MCS 600/700 (`fd_mcs`), controlador genérico para adaptadores NCR (`g_ncr5380`), NEC 53c406 (`ncr53c406`), NEC 53C710 (`ncr53c710`),NEC 53c8xx, 53c9x, Initio 91xx (`initio.o`), Initio A100U2W (`a100u2w`), adaptador NEC 53C406 (`ncr53c406`), adaptador Plug and Play symbios 53c416 (que va con algunos scanners HP, módulo sym53c416, antes debe configurarse con isapnp), adaptador de IBM PS/2 (`ibmmca`), PAS16 (`pas16`), PCI2000I EIDE (`pci2000`), PCI2220i (`pci2220i`), PSI240i (`psi240i`), Qlogic FAS (`qlogicfas`), QLogic ISP (`qlogicisp`), Qlogic isp2100 (`qlogicfc`), controladores SCSI de 8 bits de Seagate ST-02, Future Domain TMC-8xx (`seagate`), adaptadores Tecram DC390, DawiControl 2974 (`tmscsim`), adaptadores Trantor T128/T128F/T228 (`t128`), Ultrastor 14F, 24F y 34F (`u14-34f, ultrastor`). El kernel 2.4.16 soporta además los adaptadores DMX3191D (`dmx3191d`), DTC 3180/3280 (`dtc3x80`), Qlogic QLA 1280 (`qla1280`), adaptadores basados en NCR53c7, 8xx (`sim710`), y soporta las siguientes interfaces PCMCIA (para portatiles): Adaptec AHA152X (`aha152x_cs`), Future Domain PCMCIA (`fdomain_cs`), NinjaSCSI-3/NinjaSCSI-32Bi (`nsp_cs`), Qlogic PCMCIA (`qlogic_cs`).

5.3.1.9. Puertos y dispositivos seriales

Prácticamente todos los computadores con arquitectura i386 tienen 2 puertos seriales en la carcaza a los que se pueden conectar dispositivos seriales externos, estos puertos tienen 9 o 25 pines y en la carcaza son macho [48]. Entre los dispositivos seriales que pueden conectarse se cuentan modems, ratones seriales e impresoras seriales. Estos 2 puertos están conectados a circuitos integrados UART que están en la placa base, tales circuitos convierten la información digital [49] de serial a paralelo (este puerto se comunica con el procesador usando todas las líneas del bus de datos —32 o 64— i.e en paralelo mientras que recibe y envia de a un bit al dispositivo que se conecte).

Algunos dispositivos que pueden conectarse por otros medios (por ejemplo modems internos conectados a algún bus en las ranuras de la placa base), pueden tener sus propios circuitos UART y seguir el mismo estándar de los puertos de la carcaza (RS-232 de 1969 y sus sucesores e.g EIA-232).

Los puertos y dispositivos seriales, tienen varias caracteristicas que deben configurarse de la misma forma tanto en el dispositivo (o puerto) como en el sistema operativo: IRQ, dirección base, velocidad (máximo 115200 baudios[50]).

Los puertos de la carcaza normalmente pueden configurarse desde la BIOS, los dispositivos seriales pueden tener jumpers o pueden configurarse con algún programa especial.

En Linux el controlador del puerto serial (`CONFIG_SERIAL`, `serial`) crea y maneja "archivos" como `/dev/ttyS0` (c,4,64), `/dev/ttyS1` (c,4,65), `/dev/ttyS3` (c,4,66) y `/dev/ttyS4` (c,4,66) (que corresponden en nomenclatura de DOS a los puertos seriales COM1 a COM4). Además de sus funciones estándar, con hardware apropiado, puede soportar más de 4 puertos [51], (`CONFIG_SERIAL_MANY_PORTS`, `CONFIG_SERIAL_SHARE_IRQ`).

Una vez el kernel está operando la configuración de un puerto serial a nivel de sistema operativo puede verse y cambiarse con `setserial`, el siguiente ejemplo muestra como se examina la configuración de `/dev/ttyS0`

```
# setserial -a /dev/ttyS0
/dev/ttyS0, Line 0, UART: 16550A, Port: 0x03f8, IRQ: 4
Baud_base: 115200, close_delay: 50, divisor: 0
closing_wait: 3000
Flags: spd_normal skip_test
```

El ejemplo también presenta la configuración estándar del primer puerto serial, la configuración estándar de los otros 4 puertos es:

`/dev/ttyS1`

Dir. base 0x2f8, IRQ 3.

48. Conector macho se refiere a un conector que tiene pines, a ese se le puede conectar un conector hembra que debe tener ranuras para los pines.
49. Información digital: es información que se codifica con dos níveles de voltaje, puede pensarse que se trata de secuencias de ceros y unos.
50. En comunicaciones se emplea baudio como unidad para medir capacidad de transmisión. Corresponde a número de "estados" por segundo (en algunos casos corresponde a bits por segundo).
51. Los provedores de acceso a Internet requieren conectar más de un modem para atender simultaneamente varias conexiones de los usuarios, un computador con varias terminales conectadas también requiere varios puertos seriales, para lograrlo se usan tarjetas multipuerto, el kernel 2.2.19 soporta las siguientes: Hayes ESP, Digiboard Intelligent Async, Cyclades async, Control Rocketport, Computone IntelliPort Plus, Moxa Intellio, Moxa SmartIO, Multi-Tech multiport, SDL RISCom/8, Specialix IO8+/SXm/RIO y Stallion EasyIO/EC8/64.

`/dev/ttyS2`

Dir. base 0x3e8, IRQ 4.

`/dev/ttyS3`

Dir. base 0x2e8, IRQ 3.

Al emplear puertos seriales tenga en cuenta que una misma línea IRQ normalmente no puede ser usada por más de un dispositivo (verGeneralidades sobre dispositivos).

Para establecer una configuración no estándar en un puerto se emplearía algo como:

```
setserial /dev/ttyS1 uart=16450 irq=10 port=0x3e8 baud_rate=115200
```

`setserial` recibe otros parámetros para configurar más detalles del puerto serial (pueden consultarse en las página `man`).

En Debian **setserial** es ejecutado durante el arranque con la información de configuración de `/etc/serial.conf`. Los cambios a la configuración que se hagan serán salvados automáticamente a ese archivo cuando apague (puede ver el script `/etc/init.d/setserial` ver Inicialización del sistema).

De acuerdo a *Text-Terminal-HOWTO* es posible emplear un puerto serial junto con un modem nulo [52] y **getty** (ver Inicio de sesiones en consolas virtuales) para conectarse desde otro computador (o desde una terminal) y abrir una sesión. Incluso, de acuerdo a la documentación del kernel 2.4 (Documentation/serial-console.txt), hay soporte (`CONFIG_SERIAL_CONSOLE`) para que una terminal conectada al puerto serial pueda ser la consola principal.

De acuerdo a *Laplink-HOWTO* también es posible establecer una conexión PPP sobre una conexión entre puertos seriales (con un modem nulo), por ejemplo para compartir una conexión a Internet entre dos computadores a muy bajo costo.

5.3.1.10. Puerto paralelo

La mayoría de torres cuentan con al menos un puerto paralelo, el conector en el computador es hembra de 25 pines. Se llama paralelo porque transmite simultáneamente por varios cables información digital, mientras que por el puerto serial la transmisión se hace por un sólo cable. Aunque fue diseñado inicialmente junto con el IBM PC para enviar información a impresoras (el protocolo inicial se conoce hoy como SPP - Standard Paralle Port y se basa en el conector Centronics), hoy se pueden conectar muchos otros tipos de dispositivos, por lo mismo hay varios protocolos estándar para este puerto. Eventualmente su BIOS le permite cambiar de uno a otro o configurarlos (la configuración depende del dispositivo que conecte). PS/2, es una extensión a SPP que permite transmitir información desde el dispositivo hacia el computador (el SPP normalmente transmite sólo de computador a dispositivo). EPP (Enhance Parallel Port): permite transmitir información en ambas direcciones, un byte por cada ciclo del reloj del bus, puede emular SPP y eventualmente PS/2. ECP (Extended Capabilities Port): también permite transmitir en ambas direcciones sincronizado con el bus, así como transferencias DMA y compresión de datos. Cada puerto tiene varias direcciones, una línea IRQ y los puertos ECP tienen un canal DMA. De acuerdo a Printing-HOWTO algunos BIOS tratan de manejar el puerto paralelo como

52. Un cable que tiene en sus extremos dos conectores hembra para puerto serial y que intercambia algunas conexiones, si ambos conectores son de 25 pines: (DSR) 6 & (DCD) 8 – 20, (TxD) 2 – 3, (RxD) 3 – 2, (RTS) 4 – 5, (CTS) 5 – 4, (SG) 7 – 7, (DTR) 20 – 6 & 8.

dispositivo Plug and Play, en tal caso es recomendable que cambie la configuración (para que sea "Legacy", "ISA" o "0x378").

En Linux los recursos del puerto paralelos (dirección E/S, IRQ y DMA) son manejados por `parport` (`CONFIG_PARPORT`, módulo `parport`) y `parport_pc` (`CONFIG_PARPORT_PC`, módulo `parport_pc`)[53]. Con estos controladores cargados (como módulos o estáticamente ver Kernel y módulos) puede examinar los recursos que estén siendo usado por dispositivos paralelos en los subdirectorios de `/proc/parport`, por ejemplo la configuración del primer puerto paralelo en `/proc/parport/0/hardware`. Al iniciar `parport_pc` detecta automáticamente la configuración de los puertos, también puede especificarse manualmente con:

```
insmod parport_pc.o io=0x378,0x278 irq=7,5
```

que haría la configuración estándar de un puerto paralelo [54], el primero con dirección base 0x378 [55] e IRQ 7, el segundo con dirección base 0x278 e IRQ 5, puede emplearse en lugar de un número la palabra `auto` para que parport detecte la configuración. Si `parport_pc` es compilado estáticamente en el kernel puede configurarse desde el cargador de arranque con parametros como: `parport=0x378,7 parport=0x278,auto` (puede deshabilitarse con `parport=0` o solicitar autodetección —que de acuerdo a la documentación de parport puede fallar en la detección de IRQs— con: `parport=auto`

Cada dispositivo conectado al puerto paralelo requiere un controlador que interactúe con `parport`, en el caso de impresoras el controlador es `lp` (ver Impresora), otros dispositivos cuentan con controladores particulares (e.g `c-qcam`, `CONFIG_VIDEO_CQCAM` para manejar Connectix Quickcam a color), y otros tipo IDE pueden emplear `paride` como se explica a continuación. Hay también dispositivos SCSI que pueden conectarse (e.g Drives Zip), sobre los cuales puede consultarse en: http://www.torque.net/parport/ppscsi.html.

5.3.1.10.1. Dispositivos IDE por puerto paralelo y adaptadores

De acuerdo a la documentación de las fuentes del kernel, empleando adaptadores especiales pueden conectarse dispositivos IDE a puertos paralelos. Algunos diseños permiten tener hasta 4 de estos dispositivos en un mismo puerto. Para usarlos se requieren 3 tipos de controladores (disponibles tanto en 2.2 como 2.4):

Controlador base (`CONFIG_PARIDE`

Módulo `paride`).

Controladores genéricos de acuerdo al tipo de dispositivo por conectar

Discos duros

Variable de configuración del kernel `CONFIG_PARIDE_PD`; módulo `pd`.

53. `parport` es un controlador genérico para puerto paralelo, sobre este debe usarse soporte particular para la arquitectura, e.g `parport_pc` para i386, `parport_ax` para Sun Ultra/AX.
54. En modo SPP cada puerto paralelo usa 3 direcciones consecutivas, en modo EPP usa 5 también consecutivas, en modo ECP usa también 5 aunque no consecutivas.
55. Algunos puertos paralelos no tan recientes, para el primer puerto paralelo requieren io=0x3bc irq=none

CD-ROMs ATAPI

Variable de configuración `CONFIG_PARIDE_PCD`; módulo `pcd`.

Discos ATAPI

Variable de configuración `CONFIG_PARIDE_PF`; módulo `pf`.

Unidades de cinta

Variable de configuración `CONFIG_PARIDE_PT`; módulo `pt`.

Dispositivo genérico

Variable de configuración `CONFIG_PARIDE_PG`; módulo `pg`.

Controladores para los adaptadores puerto paralelo/IDE paticulares

aten ATEN EH-100, bpck Microsolutions backpack, comm DataStor "commuter" adapter, dstr DataStor EP-2000, epat Shuttle EPAT.

5.3.1.11. BIOS y otras características

APM (Advanced Power Managment) busca ahorrar energía entrando a un estado de menor consumo cuando el computador no ha sido usado durante un tiempo, así mismo permite apagar el computador por software. Debian 2.2 incluye por defecto este controlador estáticamente, debe activarse pasando durante el arranque el parámetro `apm=on` (ver Parámetros desde el cargador de arranque).

Es posible configurar el kernel (`CONFIG_NVRAM`) para que emplee el dispositivo `/dev/nvram` para leer y escribir en el CMOS de la BIOS.

También puede configurarse (`CONFIG_RTC`) para que el el reloj de tiempo real del sistema presente su estado en `/proc/driver/rtc`.

Para diversos programas, por ejemplo los relacionados con criptografía, es importante un generador de números aleatorios. Programar uno de tales generadores no es posible (sólo pueden programarse generadores seudo-aleatorios que repiten una secuencia de números muy larga), el kernel ofrece un generador más aleatorio que depende del "ruido" producido por diversos controladores. Pueden obtenerse sus datos leyendo de los dispositivos `/dev/random` (c, 1, 8) y `/dev/urandom` (c, 1, 9) (el primero más seguro pero un poco más demorado). Hay algunas tarjetas que ofrecen generación de números aleatorios (basadas en "ruido" producido por componentes electrónicos), el kernel soporta los generadores incluidos en tarjetas Intel 8xx (`CONFIG_INTEL_RNG`, `i810_rng`).

5.3.2. Lecturas recomendadas: Hardware básico y Linux

- Puede revisar más sobre riesgos y precauciones al manipular componentes de un computador en: http://home.att.net/~t.f.cook/tc15.htm

- Puede consultar generalidades sobre CPUs, su funcionamiento y proceso de fabricación en en *CPU-Design-HOWTO*. Disponible en su computador en: `/usr/share/doc/HOWTO/en-txt/CPU-Design-HOWTO.gz`. y en Internet en http://www.tldp.org/HOWTO/CPU-Design-HOWTO.html.
- Para comprar hardware recomendamos revisar comparaciones por ejemplo en: / (http://www6.tomshardware.com/), en ese sitio pueden encontrarse entre otras una guía para placas base http://www6.tomshardware.com/mainboard/97q1/970101-1/index.html, una guía sobre tipos de memoria http://www6.tomshardware.com/mainboard/98q4/981024/index.html, una guía sobre tipos de chipset http://www6.tomshardware.com/mainboard/97q1/970101-1/index.html
- Los parámetros que el kernel recibe desde el cargador de arranque pueden consultarse con **man bootparam**.
- Puede consultar más sobre Plug and Play en el *Plug-and-Play-HOWTO* disponible en su sistema en el directorio `/usr/share/doc/HOWTO` y en Internet en http://www.tldp.org/HOWTO/Plug-and-Play-HOWTO.html. Para conocer más sobre *ISA Plug and Play*, puede consultar las páginas man de **isapnp**, **isapnp.conf** el completo FAQ sobre este tema disponible en: http://www.roestock.demon.co.uk/isapnptools/isapnpfaq.html#toc1
- Para conocer más sobre puertos SCSI y su soporte en Linux *SCSI-2.4-HOWTO*, disponible en

  http://www.linuxdoc.org/HOWTO/SCSI-2.4-HOWTO/

- Hay más información sobre dispositivos USB en las fuentes del kernel en el directorio `Documentation/usb`. En Internet puede consultarse una base de dispositivos USB soportados en: http://www.qbik.ch/usb/devices/
- Puede consultar más sobre el soporte de Linux al puerto paralelo en las fuentes del kernel en Documentation/parport.txt o en: http://www.torque.net/linux-pp.html donde en particular encontrará un listado de los dispositivos, diferentes a impresoras, que pueden conectarse a puertos paralelos (e.g Unidades Zip).

  Para conocer más sobre el puerto paralelo, puede consultar el primer capítulo del libro Parallel Port Complete de Jan Alexon o el FAQ del mismo autor: http://www.lvr.com/parport.htm

  Para conectar una impresora puede consultar estas guías (ver Impresora) o *Printing-HOWTO*, en su sistema disponible en `/usr/doc/HOWTO/en-txt/Printing-HOWTO.txt.gz` y en Internet en http://www.tldp.org/HOWTO/Printing-HOWTO/index.html.

  Hay más información sobre PARIDE en las fuentes del kernel en el archivo `Documentation/paride.txt` y en las fuentes del directorio `drivers/block/paride`. En Internet: http://www.torque.net/parport/

- Puede consultarse más sobre el generador de números aleatorios del kernel en las fuentes `drivers/char/random.c`.
- Encontrará más información sobre puertos seriales en *Serial-HOWTO* `/usr/share/doc/HOWTO/en-txt/Serial-HOWTO.gz` y en la página del manual de `setserial`. En *Serial-Programming-HOWTO* encontrará información para programar a bajo nivel un puerto serial. En *Terminal-Text-HOWTO* encuentra información para conectar terminales a un puerto serial y en *Laplink-HOWTO* se describe como crear una conexión PPP sobre un puerto serial.

5.3.3. Ejercicios: Hardware básico y Linux

1. Para referencia futura, o en caso de fallas y para conocer su computador escriba un script que presente las siguientes características:

- Marca y referencia del procesador, así como su velocidad. También la cantidad de memoria RAM disponible.
- Asignación de líneas IRQ, canales DMA y direcciones E/S.
- Dispositivos PCI y recursos que emplean.

2. Opcional: en algunos casos, cuando un componente de su computador falla, el sistema completo deja de funcionar. Para estos casos, una forma de identificar el componente defectuoso es quitando todo el hardware de su computador e instalando una a una las partes y reiniciando el computador con cada parte agregada. Generalmente al iniciar un computador sin memoria ni dispositivos conectados el procesador debe hacer sonar el parlante, puede aprovechar esto y los mensajes que salgan en pantalla (si es que ya la tiene conectada) para determinar que componente falla. Un orden posible para instalar los componentes es: teclado, memoria, tarjeta de video, disco(s) duro(s), unidades de disquete, otras tarjetas. Si tiene la posibilidad y el tiempo saque todos los componentes de su computador y agréguelos uno a uno (siguiendo las precauciones mencionadas al comienzo de esta guía y las que se indiquen en los manuales).

5.4. Dispositivos y Linux

Indicadores de Logros:

- Identifica dispositivos que puede funcionar con Linux.
- Puede configurar dispositivos soportados con Linux.

5.4.1. Lectura: Dispositivos y Linux

En esta sección se presenta información sobre dispositivos que normalmente están separados de la placa base, así como detalles de su configuración en Linux. Esta sección complementa las secciones sobre hardware que típicamente está en la placa base (ver Hardware básico y Linux) y sobre dispositivos para interconexión de varios computadores (ver Dispositivos para interconexión y Linux).

5.4.1.1. Soporte de hardware en Linux

La distribución Debian 2.2 incluye un kernel reciente con todos los módulos compilados y puede detectar hardware en el momento del arranque, de forma que un administrador o un usuario en general no debe encargarse de esto. Sin embargo puede haber algunos casos excepcionales:

1. Si compra hardware diseñado para Windows o que no tenga especificaciones abiertas puede ser difícil usarlo en Linux.
2. Si compra hardware muy moderno para el cual aún no haya controladores para Linux (o si su distribución de Linux no es la más reciente).
3. Si Linux no reconoce automáticamente su hardware (aún cuando pueda soportarlo).

5.4.1.1.1. Hardware diseñado sólo para Windows

Algunos fabricantes de hardware están diseñando componentes que sólo funcionan con el sistema operativo Windows, o se niegan a publicar especificaciones que permitan crear controladores. Un ejemplo son los "Winmodems" y otros componentes que mantienen parte de la información que usualmente va en el hardware en software, disminuyendo con esto el desempeño y haciendo difícil su uso en otros sistemas operativos.

Para sobrellevar este problema usted puede considerar:

- Cuando compre hardware nuevo, pregunte siempre de forma explícita al vendedor si el hardware que le está ofreciendo funciona con Linux y en particular pregunte la versión del kernel con la que funciona (para que usted sepa si funciona con el que usted tiene). Su pregunta no sólo le permitirá saber a usted si el hardware le servirá, también le permitirá saber al vendedor que hay un mercado para hardware que funciona con Linux.
- Para determinar si un dispositivo particular es soportado, puede consultar la sección respectiva de esta guía o las lecturas que recomendamos.
- También puede consultar en los empaques del hardware, el sitio web del fabricante, en el sitio de la compra, en grupos de usuarios o en listados de hardware disponibles en Internet. Si en el website del fabricante, encuentra forma de escribir un correo electrónico, hágalo y hágales saber que usted quiere emplear ese hardware con Linux, así ellos se enterarán que hay un mercado potencial para Linux.

Si desea conocer más sobre este problema y formas de solucionarlo, puede consultar la sección "Adquisición de hardware específicamente para Linux" de "*Instalación de Debian Linux 2.2 para Intel x86*": http://www.debian.org/releases/2.2/i386/ch-hardware-req.es.html#s2.5

5.4.1.1.2. Hardware no soportado por su distribución de Linux

A medida que se produce nuevo hardware, los desarrolladores de Linux crean controladores. Es posible que usted compre hardware muy nuevo, que aunque pueda ser soportado por versiones nuevas de Linux, no sea soportado por la que usted tiene.

Para determinar si tiene este problema busque el dispositivo en la sección respectiva de esta guía o en las lecturas que recomendamos. Si no encuentra el dispositivo, puede:

- Buscar un módulo (precompilado o fuentes) que pueda funcionar con la versión del kernel que tiene, note que un módulo para una versión diferente del kernel NO servirá (ver Kernel y módulos).
- Determinar si una versión más reciente del kernel soporta el hardware —o en el caso de tarjetas graficadoras, una versión más reciente de XFree86. Si encuentra una versión más nueva que lo soporte, puede obtenerla de Internet e instalar sólo el kernel (ver Configuración del kernel) o XFree86 (verMonitor y tarjeta graficadora) sin actualizar la distribución completa.

Puede ver el hardware soportado por la distribución Debian 2.2 en estas guías la sección que corresponde al tipo de dispositivo, en las lecturas que se recomiendan en cada sección, en la sección "Hardware soportado" de "Instalación de Debian Linux 2.2 para Intel x86": http://www.debian.org/releases/2.2/i386/ch-hardware-req.es.html o en el *Hardware-HOWTO*, disponible en su sistema en `/usr/share/doc/HOWTO/en-txt/Hardware-HOWTO.gz` o en Internet en: http://www.linuxdoc.org/HOWTO/Hardware-HOWTO.html

5.4.1.1.3. Hardware no reconocido por su distribución de Linux

Si su distribución de Linux no reconoce automáticamente un dispositivo, pero usted sabe que lo soporta, intente determinar y/o configurar usted mismo los recursos de hardware que emplea —sin que entre en conflicto con otros dispositivos— y configure después Linux bien

- Activando el módulo apropiado pasando como parámetro los recursos si es que el controlador es un módulo
- pasando los parámetros desde el cargador de arranque (ver Configuración del kernel) si el controlador está incluido en el kernel. Puede consultar los parámetros en la sección correspondiente de esta guía o en las lecturas que recomendamos en cada sección.

5.4.1.2. Teclado

Los conectores de teclado típicamente son redondos de 5 pines o 6 pines [56], aunque también hay teclados USB. Linux soporta gran variedad de distribuciones de teclado, el soporte para el tipo de teclado debe estar incluido estáticamente en el kernel (aunque la mayoría de kernels NO incluyen soporte para teclados USB, puede consultar más adelante en esta sección como emplear tal tipo de teclado).

Un teclado produce en los puertos del teclado, secuencias de 1 a 6 bytes por cada tecla oprimida o soltada, estas secuencias se conocen como *scancodes* [57]. El kernel recibe estas secuencias y normalmente [58] las traduce a código internos (keycodes), que después son traducidos a ASCII o Unicode de acuerdo al mapa del teclado.

Los mapas de teclado asocian a códigos de teclado (keycode [59]) caracteres que deben producirse. Pueden definirse secuencias simultáneas de teclas (por ejemplo **Ctrl-Alt-A**) o secuencias con teclas muertas (para producir por ejemplo á al oprimir primero ’ y después a). Encontrará mapas de teclado para diversas distribuciones en el directorio `/usr/share/keymaps/i386`, el mapa que se carga durante el arranque está en `/etc/console-tools/default.kmap.gz`, archivo que puede remplazarse o editarse (descomprima antes de editar y comprima cuando termine ver Administración de archivos directorios y enlaces). X-Window tiene su propia configuración del teclado, que durante el arranque de X-Window toma del archivo `/etc/X11/XF86Config` (ver Monitor y tarjeta graficadora) pero que cada usuario puede personalizar con `xmodmap` (ver Una sesión con X-Window).

56. Los teclados no USB emplean la línea IRQ 1, y las direcciones de E/S 0x60 a 0x6F.
57. por ejemplo al oprimir **s** se produce 0x1f y al soltarlo 0x9f, puede ver estas secuencias con **showkey -s**.
58. Cada consola virtual puede tener un modo que controle la traducción de acuerdo a la aplicación que corra: por ejemplo bash –modo ASCII o Unicode— o X-Window —modo Raw— o showkeys —modo keycode).
59. La asociación entre scancodes y keycodes puede verse con **getkeycodes** y puede cambiarse con **setkeycode**.

En Colombia las distribuciones de teclado más usadas son QWERTY (la primera fila de teclas con letras de izquierda a derecha tiene las teclas **q**, **w**, **e**, **r**, **t**, **y**), y en particular:

US

que no tiene **ñ**, ni símbolos de idiomas diferentes a los requeridos en inglés, recomendamos que use el mapa `/usr/share/keymaps/i386/qwerty/us-latin1.kmap.gz` con algunas modificaciones (ver Localización y configuración de programas).

Español

Tiene **ñ** y a la derecha de esta tiene el apostrofe **'**. Mapa: `/usr/share/keymaps/i386/qwerty/es.kmap.gz`

Latinoamericano

Tiene **ñ** y a la derecha de esta tiene el paréntesis cuadrado izquierdo **[**. Mapa: `/usr/share/keymaps/i386/qwerty/la-latin1.kmap.gz`

Antes de cambiar el mapa de teclado que se carga en el arranque, puede probar el nuevo mapa desde una consola virtual con el programa **loadkeys**:

```
loadkeys /usr/share/keymaps/i386/qwerty/es.kmap.gz
```

Al editar un mapa de teclado del kernel tenga en cuenta que las líneas iniciadas con ! o # son comentarios, cada definición debe estar una línea o en varias sólo si cada línea se terminen con el caracter '\'. Pueden incluirse otros mapas con líneas como:

```
include "euro"
```

Cada tecla debe configurarse (o estar configurada en un archivo incluido) con líneas de la forma:

```
keycode  39 = +ntilde +Ntilde
        alt     keycode  39 = semicolon
```

la primera línea configura la tecla con código 39, como la ñ (i.e ntilde), la segunda indica que al oprimir **Alt** con esa tecla debe producirse el caracter ';'. +ntilde por estar en la "columna 0" de la tecla 39 es el caracter que debe producirse si la tecla se usa sola, +Ntilde el que debe producirse si se emplea con Shift [60] El caracter '+' antes de ntilde y Ntilde indica que Shift debe invertir el efecto de Caps Lock (i.e si Caps Lock está activado Shift pasa a minusculas).

El código de cada tecla puede consultarlo con el programa **showkey**, que mostrará el código de cada tecla que oprima (el teclado no podrá usarse para nada más, este programa terminará después de 10 segundos sin oprimir tecla alguna). Los nombres de los símbolos que pueden producirse pueden consultarse desde una consola texto con **dumpkeys**, por ejemplo: **dumpkeys –long-info**.

El administrador, puede configurar la rata de repetición del teclado con **kbdrate**, por ejemplo:

60. Pueden especificarse, combinaciones de teclas bien con nuevas líneas iniciadas con modificadores (e.g `shift keycode 39=+Ntilde` o bien en otras columnas de la definición sin modificadores (i.e iniciada con **keycode**), el número de columna corresponde a la suma de las siguientes posibles teclas modificadoras: Shift 1, AltGr 2, Control 4, Alt 8, ShiftL 16, ShiftR 32, CtrlL 64, CtrlR 128. Pueden omitirse algunas de las 256 columnas, con una línea que indique que columnas se especifican como `keymaps 0-2,4-5,8,12`.

```
kbdrate -r 30 -d 250
```

Cambiará la rata a 30 caracteres por segunda, con un intervalo de espera de 250mseg antes de empezar a repetir.

Aunque el mapa de teclado y la rata de repetición es común a todas las consolas virtuales, cada una tiene su propio modo y estado de leds. Puede configurarse el modo en el que el teclado produce caracteres en una consola virtual con el programa kbd_mode y el estado de los leds con setleds. Para el caso de español el modo ASCII que está configurado por defecto es suficiente, pero para otros idiomas o eventualmente para algunos programas el modo Unicode [61] es apropiado, lo puede establecer con **kbd_mode -u** [62]. Cada consola tiene su propio estado con respecto a mayúsculas sostenidas (Caps Lock), detención de desplazamiento (Scroll Lock) y teclado númerico (Num Lock), estos estados pueden cambiarse con los parametros `-num`, `+num`, `-caps`, `+caps`, `-scroll` o `+scroll` de **setleds** ('+' indica activar). Estos estados pueden o no verse en los leds del teclado físico, con el parámetro `-F` se cambia el estado de la consola pero no los leds, con `-D` se cambia el estado de ambos y con `-L` sólo el de los leds (puede haber algunos teclados cuyos leds no pueden ser controlados por `setleds`).

En X-Window, además de poder configurar el teclado con **xmodmap**, el estado de los leds puede controlarse con **xset**, e.g. **xset led 3** o **xset -led 3** para activar o desactivar el led de mayusculas sostenidas.

Teclado USB

Dado que el teclado es uno de los primeros dispositivos detectados y manejados por el kernel y dado que por defecto los kernels disponibles para Debian 2.2 no incluyen estáticamente ninguno de estos controladores, si su teclado es USB recomendamos instalar un kernel 2.4 y compilarlo con el controlador `hid` o `usbkbd` incluido estáticamente (ver Kernel y módulos).

Hay dos controladores, uno general que se apoya en el controlado HID (`CONFIG_INPUT_KEYBDEV`; `keybdev`) y uno restringido pero independiente de HID (`CONFIG_USB_KBD`; `usbkbd`) (sección sobre buses USB ver Buses). HID requiere más memoria porque además de teclados USB controla otros dispositivos para interacción con humanos (como ratón y game pads), para incluirlo estáticamente en el kernel debe configurar también Input Core Support (`CONFIG_INPUT`).

5.4.1.3. Ratón

El tipo de puerto de un ratón puede ser serial (conector rectangular de 9 o 25 pines [63]), PS/2 (conector redondo de 6 pines, también llamado Mini-DIN [64]), USB (conector rectangular) o Bus Mouse (tiene una tarjeta dedicada o se conecta a un puerto de algunas tarjetas graficadoras –ATI).

Como recursos de Hardware un mouse emplea algunas direcciones de entrada sálida y una línea IRQ (ver Generalidades sobre dispositivos), recursos que son atendidos por un controlador, de acuerdo al puerto donde se conecte. El controlador puede ser usado por algunos programas con un archivo para dispositivos (normalmente `/dev/mouse` que debe ser un enlace al dispositivo apropiado). Por ejemplo

61. Unicode es un estándar que codifica con números de 16 bits los caracteres de diversos idiomas.
62. En realidad establecerá modo UTF-8, UTF-8 es una codificación para Unicode que resulta compatible con ASCII para los primeros 256 caracteres.
63. Si un ratón serial maneja también el protocolo PS/2, puede emplearse un conversor (por ejemplo de D-SUB 9 a Mini-DIN: DTR+RTS+RI,4+7+9 - 4,+5V; CD,1 - 1,Data; TXD+GND, 3+5 - 3, Gnd; DSR,6 - 5, Clock. http://www.hardwarebook.net/adapter/serial/mouseserialps2.html
64. Si un mouse PS/2 maneja también el protocolo serial, puede emplear un conversor de PS/2 a Serial: GND, 3 - 5, GND, RxD,2-2,RxD; TxD,6-3,TxD; +5V,4-7,RTS. http://www.pchardware.org/HWB/ad_MousePs2Serial.html

es usado por `gpm` que permite emplear el ratón en consolas virtuales (verSistema de usuarios y manejo de clave) y por X-Window.

Ratón serial

Es manejado por el controlador de puertos seriales del kernel, el dispositivo puede ser por ejemplo `/dev/ttyS0` (primer puerto serial c, 4, 64) o `/dev/ttyS1` (c, 4, 65), que normalmente será configurado automáticamente por el controlador incluido estáticamente en el kernel. De requerir una configuración no estándar del puerto puede emplear el programa **setserial** (ver Puertos y dispositivos seriales).

Ratón PS/2

El puerto para ratón PS/2 emplea algunas direcciones de entrada sálida y la interrupción 1el En el kernel original de Debian 2.2, este controlador está incluido estáticamente (`CONFIG_PSMOUSE`), el dispositivo es `/dev/psaux` (c,10,1).

Bus mouse

Las versiones recientes del kernel 2.2 (posterior o igual a 2.2.19) y del kernel 2.4, soportan los siguientes bus mouse, cuyos controladores son por defecto módulos en Debian 2.2:

Inport

(`CONFIG_MS_BUSMOUSE`), se conecta a una tarjeta dedicada con un conector de 9 pines. Módulo `msbusmouse`, dispositivo `/dev/inportbm` (c,10,2). El módulo recibe el parámetro `mouse_irq` seguido del número de la interrupción, si se incluye estáticamente en el kernel puede configurarse desde del cargador de arranque (ver Parámetros desde el cargador de arranque) con `msmouse=3` (cambiando 3 por el número de interrupción).

Logitech

El conector es redondo de 9 pines, (`CONFIG_BUSMOUSE`), módulo `busmouse`, dispositivo `/dev/logibm` (c,10,0). Al igual que Inport el módulo recibe el parámetro `mouse_irq=3` y desde el cargador de arranque puede usarse `bmouse=3` (remplazando 3 por el número de interrupción).

ATI-XL

(`CONFIG_ATIXL_BUSMOUSE`) módulo `atixlmouse`, dispositivo `/dev/atibm` (c,10,3). Por defecto emplea la interrupción 5 (que puede configurarse en la tarjeta), de requerir otra debe editarse y recompilarse el módulo (en las fuentes `driver/char/atixlmouse.c` línea con `ATIXL_MOUSE_IRQ`).

C&T 82C710 del TI Travelmate

(`CONFIG_82C710_MOUSE`), módulo `qpmouse` emplea por defecto la interrupción 12, emplea el mismo dispositivo de PS/2 (i.e `/dev/psaux`, c,10,1).

PC110 digitizer pad

Usado para emular un mouse PS/2 con la tableta digitalizadora del palmtop IBM PC110 (`CONFIG_PC110_PAD`), módulo `pc110pad`. Dispositivo `/dev/pc110pad` (c,10,9).

Ratón USB

Hay un controlador general apoyado en HID (ver Buses, sección USB), (`CONFIG_INPUT_MOUSEDEV`, `CONFIG_INPUT_MOUSEDEV_SCREEN_X`, `CONFIG_INPUT_MOUSEDEV_SCREEN_Y`), módulo `mousedev`. Maneja los dispositivos `/dev/input/mouse0` (13,32) y `/dev/input/mice` (13,63) que emulan un mouse PS/2. En situaciones especiales pueden emplearse en lugar de este un controlador restringido que no requiere HID: `CONFIG_USB_MOUSE`; módulo `usbmouse`. De requerir el controlador basado en HID incluido estáticamente en el kernel recuerde agregar también Input core support `CONFIG_INPUT`.

5.4.1.3.1. Programas específicas para el ratón

Es recomendable que cree el enlace `/dev/mouse` al dispositivo de su ratón:

```
ln -s /dev/psaux /dev/mouse
```

El programa **gpm** (paquete `gpm`) le permitirá copiar y pegar textos entre consolas virtuales. Se trata de un programa que se inicia durante el arranque y permanentemente monitorea el estado del mouse para copiar y pegar textos [65]. Para configurar **gpm** —una vez el controlador del puerto al que está conectado el ratón esté operando— puede emplear el programa gpmconfig, que le pedirá algunos parámetros de configuración como el puerto al que está conectado (e.g `/dev/mouse`) y el protocolo que usa (normalmente `ms` para ratones seriales y `ps2` para ratones PS/2).

X-Window emplea el ratón de acuerdo a la configuración del archivo `/etc/X11/XF86Config` (ver Monitor y tarjeta graficadora), en la sección `Pointer`. Soporta diversos protocolos (el más común para puertos seriales es el protocolo Microsoft y para puertos PS/2 el protocolo PS/2). Puede emular 3 botones con mouse que sólo tienen dos (el tercero se "obtiene" oprimiendo simultaneamente los dos) agregando `Emulate3Buttons`. También permite configurar la rueda para desplazamiento (junto con 3 botón) de algunos ratones, agregando en la sección `Pointer` una línea con `ZAxisMapping 4 5`, aunque también se requieren programas que puedan aprovechar esta última característica (e.g Mozilla, programas de Gnome y KDE).

5.4.1.4. Monitor y tarjeta graficadora

El tipo de monitor más común en la actualidad tiene un tubo de rayos catódicos (CRT), que emplea imanes para apuntar un haz de electrones sobre una capa de fluorescente en el interior del monitor. Este haz recorre permanentemente la pantalla de de izquierda a derecha (con una frecuencia horizontal) y de arriba hacia abajo (con una frecuencia vertical) haciendo brillar algunas zonas durante un corto período de tiempo. De un monitor a otro tanto la frecuencia horizontal [66] como vertical [67] varian, consulte las precisas de su monitor en el manual o si X-Window ya está configurado puede verlos en `/etc/X11/XF86Config` en líneas iniciadas con `HorizSync` y `VertRefresh`.

65. Si se inició con el parámetro `-R`, gpm retransmite la información al dispositivo `/dev/gpmdata` (con el protocolo de ratones Microsoft) lo cual puede ser util para emplear gpm simultaneamente con otros programas (como X-Window)
66. La frecuencia horizontal es un rango que suelen estar entre 30Hz y 150Hz —en televisores en Colombia es 15,735Hz.
67. La frecuencia vertical es un rango que suele estar entre 50Hz y 135Hz —en televisores en Colombia es 60Hz.

La tarjeta graficadora puede tener memoria RAM específica para gráficas, que le permitirán soportar diversas resoluciones (las más comunes son 640x480, 800x600, 1024x768, 1152x864, 1280x1024, 1600x1200 y 1800x1440), y una cantidad máxima de colores (2,16,256, 65536 y 16 millones).

Para operar un sistema Linux en un i386, no se requiere configurar la tarjeta graficadora porque puede operarse en modo texto. Sin embargo para emplear características de internacionalización del kernel (como fuentes para alfabetos diferentes al del español), consolas en modo gráfico o para iniciar X-Window se debe configurar la tarjeta graficadora y el monitor.

5.4.1.4.1. X-Window

Esta sección presenta configuración de la tarjeta graficadora y el monitor en X-Window, operación que normalmente debe hacer una sola vez (otros aspectos de X-Windows se documentan en otras secciones, ver Inicio de sesiones X-Window).

Como implementación de X-Window, Linux y otros sistemas de libre redistribución para i386 emplean XFree86 (http://www.xfree86.org). Por esto, si no planea usar fuentes de otros idiomas en la consola, para la mayoría de tarjetas graficadoras sólo requerirá configurar XFree86 (aunque para algunas tarjetas recientes con aceleración 3D que emplean DRI/DRM puede requerir soporte del kernel i.e un módulo).

Debian 2.2 incluye el servidor XFree86 3.3.6, el cual en el momento de este escrito ya ha sido descontinuado (ver más adelante sobre como actualizar a uno de la serie 4.x de requerirse), aunque soporta gran variedad de tarjetas recientes y no tan recientes (hay algunas tarjetas que 3.3.6 soporta, pero que la serie 4.x no), en cuanto a memoria aunque el mínimo requerido por un XFree86 3.3.6 es 4MB de RAM, sólo puede trabajarse confortablemente con 16MB o más.

Para que X-Window pueda emplear los dispositivos que tiene, deben configurar: 1) El servidor por defecto 2) el archivo `/etc/X11/XF86Config` que incluye información sobre el monitor, el mouse (ver Ratón), la tarjeta de video y el teclado (aunque por defecto cuando X inicia emplea la configuración del teclado del kernel ver Teclado). Para algunas tarjetas la configuración puede realizarse facilmente con el programa `anXious`, que es ejecutado durante la instalación de Debian, pero que sólo soporta algunas tarjetas PCI (paquete `xviddetect`).

En la mayoría de casos el servidor por defecto que debe usar es XF86_SVGA (paquete `xserver-svga`), si su hardware lo requiere o lo soporta puede emplear otro como se presenta más adelante. Para cambiar la configuración puede emplear el programa XF86Setup (paquete xf86setup) que funciona en modo gráfico y requiere el servidor `XF86_VGA` instalado (paquete `xserver-vga16`), o puede emplear en modo texto el programa xf86config o eventualmente puede editar directamente el archivo `/etc/X11/XF86Config`. En todos los casos, no siempre es indispensable que escoja la tarjeta, ni el mouse que tiene (al iniciar X-Window intentará autodetectarlos), y normalmente será muy importante que escoja correctamente la frecuencia horizontal y vertical de su monitor (consulte el manual o intente con diversos valores sin sobrepasar los límites máximos de su monitor). Cada vez que cambie la configuración puede intentar arrancar el servidor con X, o con el nombre del servidor (e.g XF86_SVGA), o con startx, o con `initx` (ver Inicio de sesiones X-Window), mientras hace pruebas con la configuración, es recomendablep que redireccione errores a un archivo para buscar allí eventuales fallas:

```
startx 2> errores.txt
```

El servidor por defecto es un programa del directorio `/usr/X11R6/bin` apropiado para su tarjeta y que debe enlazarse desde `/usr/X11R6/bin/X`, así mismo debe ser especificado en la primera línea del archivo /etc/X11/Xserver. Así, si su servidor es /usr/X11R6/bin/XF86_SVGA (que es el más común porque soporta tarjetas con chips Cirrus 542x, 543x, Western Digital 90c3x y Oak) debe ejecutar:

```
ln -s /usr/X11R6/bin/XF86_SVGA /usr/X11R6/bin/X
```

y editar /etc/X11/Xserver.

Si el servidor XF86_SVGA no logra emplear su tarjeta, eventualmente puede intentar primero con /usr/X11R6/bin/XF86_VGA (paquete xserver-vga) que es un servidor genérico que soporta una resolución máxima de 640x480 y 16 colores (existen unas pocas tarjetas que no pueden usarse con este servidor). Eventualmente su tarjeta puede contar con soporte especial con otro servidor (también ubicados en el directorio /usr/X11R6/bin/ con nombres que comienzan con XF86_) disponible en alguno de los siguientes paquetes:

xserver-rage128

Para tarjetas ATI con chipset Rage128: ATI Rage Fury, ATI Rage Magnum, ATI Xpert99, ATI Xpert128, ATI Xpert2000 y ATI All-in-Wonder 128

xserver-mach32

Para tarjetas basadas en ATI Mach32.

xserver-mach64

Para tarjetas basadas en ATI Mach64: 3D Rage, 3D Rage II, y 3D Rage Pro

xserver-i128

Para tarjetas Number Nine Imagine 128, Ticket 2 Ride, Revolution 3D y Revolution IV.

xserver-3dlabs

Para tarjetas 3DLabs GLINT y basadas en Permedia.

xserver-agx

Tarjetas basadas en IBM XGA e IIT AGX.

xserver-ggi

Tarjetas LibGGI (no emplean XF86Config, consultar documentación del paquete).

xserver-8514

Tarjetas basadas en ATI 8514/A.

xserver-s3v

Tarjetas basadas en S3 ViRGE y ViRGE/VX, aunque el servidor XF86_SVGA también las soporta.

xserver-w32

Tarjetas basadas en Tseng ET4000/W32 y ET6000-based.

xserver-mono

Tarjetas y/o monitores monocromáticos.

`xserver-sis`

Para algunas tarjetas SiS como SiS 86c201, SiS 86c202, SiS 86c205, SiS 5597, SiS 5598, SiS 6326 AGP, SiS 530/620, SiS 540/630 and SiS 300 (otras son soportadas por XF86_SVGA).

`xserver-s3`

Tarjetas basada en el chipset S3.

`xserver-p9000`

Tarjetas basadas en Weitek P9000.

`xserver-mach8`

Tarjetas basadas en ATI Mach8.

`xserver-fbdev`

Tarjetas que emplean controlador framebuffer.

Los kernels recientes de la serie 2.2 y los de la serie 2.4 ofrecen soporte para tarjetas AGP (verBuses), y para la infraestructura DRI/DRM[68] introducida en XFree86 4.x (`CONFIG_DRM`). Ambos soportan las tarjetas: 3dfx Banshee/Voodoo3+ (`CONFIG_DRM_TDFX`, `tdfx`), 3dlabs GMX 200 (`CONFIG_DRM_GAMMA`, gamma), ATI Rage 128 (`CONFIG_DRM_R128`), Intel 810 (`CONFIG_DRM_I810`, `i810`), Matrox g200/g400 (`CONFIG_DRM_MGA`, `mga`) y el kernel 2.4.16 soporta también ATI Radeon (`CONFIG_DRM_RADEON`).

Si su tarjeta no es soportada por el servidor 3.3.6 de Xfree86 puede actualizar a la versión 4.1, empleando paquetes no oficiales para Debian 2.2, para eso agregue al archivo `/etc/apt/sources.list` las líneas:

```
deb http://people.debian.org/%7Ecpbotha/ xf410_potato/i386/
deb http://people.debian.org/%7Ecpbotha/ xf410_potato/all/
```

y despues ejecute (ver Paquetes en Debian):

```
apt-get update
apt-get -u dist-upgrade
apt-get install xserver-xfree86
wget http://people.debian.org/%7Ecpbotha/xf4_potato_deps/libfreetype6_potato/libfreetype6_2.
1_i386.deb
dpkg -i libfreetype6_2.0.2.20010514-1_i386.deb
apt-get -u dist-upgrade
```

Después de actualizar o si lo requiere puede ejecutar el programa de configuración con:

```
dpkg-reconfigure xserver-xfree86
```

o emplear el nuevo programa de configuración de XFree 4.x `xf86cfg`. El archivo de configuración en Debian es `/etc/X11/XF86Config-4`, si emplea un teclado US puede habilitar teclas muertas en la sección Keyboard con:

```
   Option          "XkbLayout"     "us-intl"
```

68. Direct Rendering Interface/Direct Rendering Manager

Si su tarjeta de video no es soportada por las versiones 3.3.6 ni 4.1 de XFree86, puede intentar compilar la versión más reciente (ver Compilación e instalación de programas a partir de fuentes). Para compilar la versión 4.2 (y eventualmente otras) puede aprovechar los tipos de letra del paquete 4.1 (que debe instalar primero) y compilar sólo la parte mínima de XFree86. En tal caso instale los paquetes `libncurses5-dev`, `zlib1g-dev`, `libpam0g-dev` y una vez descargue las fuentes de XFree86 4.2 (http://www.xfree86.org), edite el archivo `xt/config/cf/site.def` y agregue la línea `#define BuildFonts NO`, después puede compilar, instalar y probar sus cambios. Para compilar e instalar emplee `make World` (tomará más de 30 minutos) seguido de `make install` (eventualmente tendrá que borrar los archivos `/etc/X11/lbxproxy/AtomControl` y `/etc/X11/proxymngr/pmconfig` y eliminar la línea `case ABS_WHEEL` de la fuente `programs/Xserver/hw/xfree86/input/wacom/xf86Wacom.c`

5.4.1.5. Impresora

En esta sección se presenta como configurar una nueva impresora local (conectada a uno de los puertos parelelos del computador en el que se configura) o como reconfigurar una existente con el sisteam lpd [69],

el uso de una impresora ya configurada y algunas nociones básicas se presentan en el capítulo 3 (ver Impresión y formatos para impresión) mientras que la configuración de impresoras remotas en el siguiente capítulo (ver Impresora en red).

5.4.1.5.1. Impresora local

La mayoría de impresoras se conectan al puerto paralelo (por ejemplo utilizables con dispositivos `/dev/lp0` o `/dev/lp1`). Eventualmente el kernel cargará por demanda los controladores apropiados, cuando intente usarla (e.g. cuando inicie `lpd`). E n ese caso o si ya ha configurado manualmente el controlador apropiado (verBuses, ver Puertos y dispositivos seriales, ver Puerto paralelo), puede probar que su impresora funciona enviando una cadena sencilla al puerto apropiad por ejemplo `echo "Hola" >/dev/lp0` o `lptest > /dev/lp0`.

Si su impresora se conecta al puerto paralelo y requiere configuración manual tenga en cuenta que se requiere soporte del kernel (`CONFIG_PRINTER`, módulo `lp`), el cual en el caso de Debian 2.2 está incluido como módulo, así que de requerirlo sólo debe cargarlo durante el arranque y eventualmente pasar los parámetros apropiados (use por ejemplo `modconf` ver Módulos). Dado que `lp` emplea `parport` para el manejo a bajo nivel del puerto paralelo (verPuerto paralelo) los parámetros que este controlador recibe hacen referencia a dispositivios configurados con `parport`:

`parport=0,none,1`

Para configurar `/dev/lp0` (c, 6, 0) como el primer dispositivo de `parport`, deshabilitar `/dev/lp1` (c, 6, 1) y configurar `/dev/lp2` (c, 6, 2) con el segundo dispositivo de `parport`. En caso de tener el controlador estáticamente en el kernel, el parámetro análogo que debería pasar desde el cargador de arranque es `lp=parport0,none,parport1`. Note que con este esquema `dev/lp0` podría referirse a un puerto paralelo arbitrario.

69. Para atender trabajos de impresión hay disponibles varios sistemas: lpd, LPRng, CUPS, QPD. El más popular en sistemas Unix es **lpd** que permite a varios computadores en red compartir una misma impresora.

`parport=auto`

Intenta autodetectar impresoras y crear los dispositivos necesarios. Si el controlador está incluido estaticamente `lp=auto`.

`reset=1`

Reinicializa impresora cuando carga el módulo. Si el controlador está incluido estaticamente `lp=reset` o para desactivar por completo el controlador `lp=off`.

Para atender solicitudes de impresión hechas por `lpr` (ver Impresión y formatos para impresión) debe configurar `lpd` durante el arranque, esto ocurrirá automáticamente cuando instale el paquete `lpr` y debe configurar la impresora en `/etc/printcap`. El siguiente es un ejemplo de tal archivo:

```
lp|local line printer:\
        :lp=/dev/lp0:sd=/var/spool/lpd/lp:lf=/var/log/lp-errs:

psprn|rp|remote line printer:\
         :lp=:rm=rojo:rp=psprn:sd=/var/spool/lpd/psprn: \
:lf=/var/log/lp-errs:
```

que configura dos impresoras, la primera local llamada `lp` o `local line printer` y la segunda remota llamada `psprn` o `rp` o `remote line printer`. La impresora local está conectada a `/dev/lp0`, la cola de impresión la mantiene en `/var/spool/lp` y envia errores a `/var/log/lp-errs`. La impresora remota está conectada a `rojo.micolegio.edu.co` (suponiendo que `micolegio.edu.co` es el dominio del sistemas donde está `/etc/printcap`), el nombre en la máquina remota es también `psprn` y la cola de impresión en la máquina local es `/var/spool/lpd/psprn`.

El ejemplo anterior no presenta como se configura filtros (para que `lpd` pueda imprimir documentos PostScript o PDF), ni muchos otros detalles configurables en `/etc/printcap` (ver `man printcap`). Para configurar un filtro de PostScript y PDF y ayudarle en la configuración de una impresora local puede emplear el programa `magicfilter`, el cual con base en los filtros de Ghostscript le puede ayudar a elegir el apropiado para su impresora. `magicfilterconf` le hará algunas preguntas sobre su impresora, modificará el archivo `/etc/printcap` y creara los directorios apropiados.

Si no logra determinar un filtro apropiado para su impresora puede revisar la base de datos de impresoras soportadas en Linux en: http://www.linuxprinting.org/database.html donde además de filtros para algunas impresoras que Ghostscript no soporta, encontrará concejo sobre que impresora comprar.

Mientras cambia su configuración puede probar imprimir un texto simple por ejemplo con **lptest | lpr**. Una vez configurada una impresora los usuarios locales podrán imprimir con el programa **lpr** (tanto textos como PostScript como PDF) y podrá controlar las colas con el programa **lpc**.

5.4.1.6. Discos duros

Un disco duro consta de varias placas circulares sobre las que se almacena información magneticamente. La organización o geometría de un disco suele especificarse como cantidad de cilindros [70], cantidad de cabezas [71] y cantidad de sectores. Para poder emplear un disco en Linux, se requiere un controlador que lo maneje, el disco debe estar formateado a bajo nivel, debe estar particionado, una o más particiones

70. cilindros: del inglés *cylinder*.
71. cabeza: del inglés *head*.

deben ser para Linux (tipo 83), una puede ser para swap (tipo 82) y las particiones para Linux deben tener un sistema de archivos apropiado para montarlo (ver Montaje y desmontaje de sistemas de archivos), como ext2 (ver Sistema de archivos ext2).

Además de esto para iniciar un computador con un disco duro, debe tener una partición marcada como iniciable en la tabla de particiones o debe emplear un cargador de arranque (e.g LILO o GRUB).

Todo disco duro cuya interfaz sea soportada por Linux debe funcionar sin requerir configuración manual. Debian 2.2 incluye estaticamente controladores para interfaces de discos RLL, MFM diversos IDE/EIDE y tiene módulos para diversos discos SCSI, así como módulos para discos conectados a puerto paralelo (ver Puerto paralelo) y controladores para arreglos de de discos RAID (0,1,4/5) —para respaldar información [72].

5.4.1.6.1. Discos IDE

Como se presentó en la sección sobre dispositivos IDE (ver Interfaz y dispositivos IDE) el controlador de esta interfaz está incluido estáticamente en el kernel y reconoce automáticamente los discos conectados (probando o acudiendo a CMOS/BIOS), si ustede requiere configurar manualmente el disco conectado a un puerto IDE puede hacerlo pasando como parámetro desde el cargador de arranque (ver Parámetros desde el cargador de arranque) la geometría del disco, e.g: `hdd=cdrom` o `hdc=1050,32,64` remplazando 1050 por número de cilindros, 32 por número de cabezas y 64 por número de sectores.

Los nombres de los dispositivos son: `/dev/hda` (primaria maestro), `/dev/hdb` (primario esclavo), `/dev/hdc` (secundario maestro) y `/dev/hdd` (secundario esclavo).

> Cuando lea o escriba en estos dispositivos (sólo lo puede hacer el usuario `root`), estará leyendo y escribiendo directamente sobre sectores de su disco

Para verificar la detección de su(s) disco(s) IDE puede consultar los mensajes del arranque (con **dmesg** o **less /var/log/dmesg**) o puede emplear el programa **ide_info** seguido del nombre de un dispositivo IDE, por ejemplo:

```
$ ide_info /dev/hda
MODEL="IBM-DTCA-24090"
FW_REV="TC6OAB4A"
SERIAL_NO="KY0KYJC0917"
```

Las particiones de cada disco se enumeran comenzando en uno, por ejemplo las tres primeras particiones de un disco primario maestro se identifican con: `/dev/hda1`, `/dev/dha2` y `/dev/hda3` (dispositivo por bloques con mayor 3, menor 0 para el disco entero 1 para la primera partición, 2 para la segunda y así sucesivamente hasta máximo 63).

72. El kernel también cuenta con controladores para tarjetas controladoras para discos duros XT (`CONFIG_BLK_DEV_XD`, módulo `xd`) y para discos ESDI de PS/2 (`CONFIG_BLK_DEV_PS2`, módulo `ps2esdi`), así como 2 discos con interfaces defectuosas (cuyos problemas soluciona): CMD-640 y RZ1000.

5.4.1.6.2. Discos SCSI

Los discos con interfaz SCSI son en general más rápidos que los IDE pero más costosos y típicamente requieren una tarjeta controladora. Tal como se presentó en otra sección (ver Adaptadores y dispositivos SCSI), con una controladora SCSI soportada, se puede emplear el módulo `sd_mod` (`CONFIG_BLK_DEV_SD`) para manejar uno o varios discos SCSI.

El primero disco SCSI es `/dev/sda` y su primera partición es `/dev/sda1`, el segundo, `/dev/sdb` y así sucesivamente hasta `/dev/sdh`.

5.4.1.6.3. Particiones

Una partición es una porción de un disco duro destinada para un sistema de archivos. Un disco duro puede particionarese para:

- Mantener varios sistemas operativos.
- Destinar varias particiones a Linux montando cada partición como un directorio (y limitando así el espacio de esos directorios), por ejemplo `/var` (donde está la cola de correo), `/usr` donde se ubican programas, `/home` donde cada usuario tiene su espacio personal (ver Ubicación de archivos y directorios).
- Destinar alguna partición como zona de intercambio (swap) para emplear espacio de disco como si fuera memoria RAM.
- Destinar alguna partición al directorio `/boot` para facilitar el arranque de Linux en algunos computadores con discos duros de más de 1024 cilindros, como se explica a continuación.

La división en particiones de un disco duro se mantiene en una *tabla de particiones* que está en el primer sector físico, que además puede tener un cargador de arranque (ver Inicialización del sistema).

Linux en un PC puede manejar a lo sumo 4 particiones **primarias** (en el caso de un disco IDE las particiones primarias están asociadas a los dispositivos `/dev/hda1`, `/dev/hda2`, `/dev/hda3` y `/dev/hda4`). Dado que pueden requerirse más de 4, una de las particones primarias puede remplazarse por una partición extendida, y tal partición extendida puede entonces dividirse en una o más particiones lógicas (que en el caso de un disco IDE primario se referencian como `/dev/hda5`, `/dev/hda6`, y así sucesivamente).

Para la operación de Debian se requiere al menos una partición de 300MB (o de 800MB para un sistema básico o 2GB para una instalación completa), aunque consideramos recomendable al menos una partición más para *swap* (memoria virtual [73]) de un tamaño cercano a la cantidad de RAM del computador. En los casos de un servidor o un cliente para una red en un colegio lo invitamos a consultar nuestra sugerencia para la división del espacio en particiones Ver Plataforma de referencia.

Aunque en un mismo disco duro pueden dejarse diversos sistemas operativos, por razones históricas, con diversas BIOS (previas a 1998 o que no soporten LBA32) es indispensable dejar el arranque de cada sistema operativo en los primeros 1024 cilindros. Para facilitar esto, en caso de requerirse, el arranque básico de Linux (directorio `/boot`) puede dejarse en una partición pequeña (e.g 10MB) en los primeros 1024 cilindros o incluso como un directorio en DOS, mientras que el resto del sistema puede estar en una o más particiones en cualquier ubicación del disco.

73. Memoria virtual es un espacio en disco que puede emplearse para mantener información que típicamente está en memoria RAM, dotando de más memoria RAM un computador a cambio de una disminución en el desempeño —el acceso a un disco duro (ms) es mucho más demorado que el acceso a memoria RAM (ns).

Para cambiar la tabla de particiones de un disco en Linux pueden emplearse los programas **cfdisk** o **fdisk** [74]. Ambos se inician pasando como parámetro el dispositivo del disco que desea editar (e.g `/dev/hda` o `/dev/sda`), le permiten modificar la partición hasta que este satisfecho con la distribución y finalmente permiten salvar la partición configurada en el disco.

> **Aviso**
>
> Al modificar una partición el sistema de archivos que en ella hubiera no podrá usarse, es muy recomendable que mantenga en un escrito la tabla de particiones de su disco incluyendo tipo, inicio y fin de cada una para recuperarla de requerirse.

5.4.1.6.4. Zonas de intercambio

Normalmente se destinará una partición como zona de intercambio. Una vez en funcionamiento puede desactivarse el uso de memoria virtual con **swapoff** y reactivarse con **swapon**. De requerirse más memoria virtual puede destinarse una nueva partición para esto (tipo 82) y puede inicializarse con el programa **mkswap**. El dispositivo debe ser entonces agregado a `/etc/fstab` en una línea de la forma:

```
/dev/hdb6    none   swap    sw    0   0
```

para que pueda ser usada durante el arranque por `swapon -a`.

5.4.1.6.5. Cambio del tamaño de un sistema de archivos

En otras secciones se presentan las generalidades sobre sistemas de archivos (ver Montaje y desmontaje de sistemas de archivos, ver Sistemas de archivos en Linux) y sobre el sistema de archivos ext2 típico de Linux (ver Sistema de archivos ext2), aquí encontrará información para cambiar el tamaño de un sistema de archivos.

Después de que un disco ha sido particionado y se le ha instalado un sistema de archivos, puede ser difícil cambiar el tamaño de una partición. Si se desea una partición más grande sin borrar el sistema de archivos es necesario liberar espacio a continuación de la partición, aumentar el tamaño de la partición (desde Linux puede usarse **fdisk** o **cfdisk**) y después aumentar el tamaño del sistema de archivos. Si se desea una partición más pequeña, primero debe disminuirse el tamaño del sistema de archivos y después debe disminuirse el tamaño de la partición recortando el final —teniendo cuidado de no sobreescribir sobre otro sistema de archivos. El tamaño de un sistema ext2 puede cambiarse con **rezisee2fs**, el tamaño de otros sistemas de archivos (entre otros vfat de Windows) puede cambiarse desde Linux con el programa `parted`.

Al instalar Linux, si ya tiene otro sistema operativo que ocupa el disco entero puede emplear un programa análogo a **parted** que funcione en el sistema operativo ya instalado. Para el caso de Windows y DOS puede emplearse **fips** que disminuye el tamaño de un sistema de archivos FAT12, FAT16 o FAT32, así como el de la partición en la que está, liberando con esto espacio que puede asignarse a una o más particiones para Linux. De acuerdo a la documentación de **fips**, la forma más precisa de usarlo es desde un disquete de arranque (para DOS) después de defragmentar el disco y de evitar dejar bloques inamobibles al final de la partición (puede ser quitando temporalmente soporte para swap a Windows o

74. Otro programa para particionar discos, menos interactivo es **sfdisk**.

borrando los archivos image.idx o mirorsav.fil que volverán a generarse posteriormente: `attrib -r -s -h image.idx`, `attrib -r -s -h mirorsav.fil` y `del image.idx`, `del mirrorsav.fil`).

5.4.1.6.6. Cargador de arranque

LILO puede instalarse en el MBR de un disco, o en el sector de arranque de alguna partición (incluso de particiones lógicas) o en el sector de arranque de un disquete. Se configura en el archivo `/etc/lilo.conf`, después de modificar ese archivo debe ejecutarse el programa **/sbin/lilo**. Un ejemplo de este archivo con comentarios (líneas iniciadas con el caracter '#') que explican su uso es:

```
# Soporte LBA para discos grandes (más de 8GB)
lba32

# En este dispositivo LILO se instalará
boot=/dev/hda3

# Dispositivo que debe montarse como raiz
root=/dev/hda3

# Archivo con nuevo sector de arranque
install=/boot/boot.b

# Localización del mapa creado por lilo
map=/boot/map

# Milisegundos por esperar antes de arrancar imagen por defecto
delay=20

# Imagen por iniciar por defecto
default=Linux

# Primera imagen
image=/vmlinuz
label=Linux
read-only
initrd=/initrd.img

# Segunda imagen, un kernel de repuesto
image=/vmlinuz.old
label=LinuxOLD
read-only
optional

# Otro sistema operativo
other=/dev/hda1
        label=OpenBSD
        table=/dev/hda
```

Además de estas posibilidades de configuración, se presentan otras y la forma de pasar parámetros al kernel en otra sección (ver Parámetros desde el cargador de arranque).

5.4.1.7. Disquetes

Una unidad de disquete permite almacenar datos magnéticos en un disquete, emplea una línea de DMA (por defecto 2), una interrupción IRQ (por defecto 6). Dependiendo de la cantidad de información que puede almacenar una unidad o un disquete pueden ser de densidad doble, alta y extra alta, las dimensiones pueden ser de 3 1/2 pulgadas o 5 1/4.

El controlador de Linux (`CONFIG_BLK_DEV_FD`, módulo `floppy`) soporta unidades de baja, alta y extra alta densidad, puede soportar más de dos y soporta algunos formatos de capacidad extra [75].

Los parámetros cuando está incluido estáticamente son los mismos del módulo: `floppy=daring` mejor el desempeño de algunas unidades, `floppy=two_fdc` indica que tiene dos conroladoras de floppy, la seguda en la dirección 0x370 (puede cambiarse poniendo la dirección antes de `two_fdc`), `floopy=thinkpad` para usuarios de Thinkpad, `floppy=nodma` para evitar uso de DMA, `floppy=nofifo` deshabilita FIFO, `floppy=`*`unidad`*`,`*`tipo`*`,cmos` establece tipo CMOS de una unidad (1 - 5 1/4DD, 2 - 5 1/4HD, 3 - 3 1/2DD, 4 - 3 1/2HD, 5 - 3 1/2 ED, 6 3 1/2 ED), `floppy=broken_dcl` no usar línea de cambio de disco, `floppy=6,irq` para establecer IRQ, `floppy=2,dma` para establecer DMA. Documentación completa en `driver/block/README.fd`.

Aunque hay varios dispositivos para unidades de disquete (dependiendo del formato), los dispositivos `/dev/fd0`, `/dev/fd1`, etc. autodetectarán el formato. Para acceder a un disquete pueden usarse estos con **mount/umount** o con las herramientas del paquete `mtools`. Si se emplea **mount** es recomendable agregar a `/etc/fstab`:

```
/dev/fd0  /floppy  auto  defaults,user,noauto   0  0
```

que permitira a los usuario montar sus disquetes (sistema de archivo autodetectado) y que no montará la unidad durante el arranque (ver Lectura Disquetes y CDROM).

Para formatear un disquete con sistema de archivos vfat (leible también en Windows) puede emplearse:

```
mformat a:
```

o `superformat` (herramienta del paquete `fdutils`).

El paquete `fdutils` incluye entre otros los programas: **diskseekd** para mover la cabeza de la unidad periodicamente (para quitar el polvo), **xdfcopy** para copiar discos con formato XDF (1840MB), **floppymeter** para medir capacidad y velocidad de rotación de la unidad, **fdmount** monta un disquete, **floppycontrol** para configurar la unidad, **fdumount** para desmontar un disquete, **superformat** para formatear un disquete (puede ser a capacidades no estándar), **fdrawcmd** para enviar comandos a la unidad, **xdfformat** para formatear un disquete con formato XDF, **setfdprm** establece parámetros para un disquete, **fdlist** lista de unidades de disquete, **getfdprm** permite examinar parámetros establecidos con **setfdprm**. **fdmountd** monta un disquete autodetectando el formato.

5.4.1.8. Unidades de CD-ROM y DVD

Una vez configure su unidad de CD o DVD podrá montar CDs de datos —si se trata de un CD con datos— o con un programa apropiado podrá escuchar CDs de audio (ver al final de esta sección). Los

75. De acuerdo a la documentación de `superformat`, puede formatearse un disquete de alta densidad a 1734K (83 cilindros, 21 sectores) con `superformat /dev/fd0 sect=21 cyl=83` y a 1992K (83 cilindors, pistas de 12KB) con `superformat /dev/fd0 tracksize=12KB cyl=83 mss`

controladores para los tipos más comunes de unidades de CD y DVD también soportan CDs multisesión (PhotoCD) y algunos soportan lectura de datos digitales.

En el momento de este escrito tanto la serie 2.2.x como la 2.4.x del kernel soportan las unidades de CD-ROM y DVD más estándares IDE/ATAPI y SCSI así como algunas unidades no tan recientes con sus propias interfaces. No soporta unidades Vertos, Genoa y algunos modelos Funai.

5.4.1.8.1. Unidades IDE/ATAPI y SCSI

Muchas unidades de CD-ROM y DVD comunes se conectan a un puerto IDE y siguen la especificación ATAPI. La configuración del hardware es análoga a la de discos IDE (maestro o esclavo) y el nombre del dispositivo en Linux sigue la misma convención de discos duros IDE (`/dev/hda`, `/dev/hdb`, etc. Interfaz y dispositivos IDE). El controlador de estas unidades es **ide-cd**, que por defecto está incluido estáticamente en el kernel y que reconoce y configura automáticamente su(s) unidad(es) mientras empleen dirección base estándar (ie. 0x1f0, 0x3f6 para ide0 y 0x170, 0x376 para ide1). Si la autodetección no reconoce su unidad puede pasar como parámetro al kernel (desde el cargador de arranque): `hdb=cdrom` cambiando `hdb` por el dispositivo IDE apropiado, o puede especificar direcciones base e IRQ del puerto IDE: `ide1=0x170,0x376,15` remplazando 0x170 por la primera dirección base, 0x376 por la segunda 15 por la IRQ (ver Parámetros desde el cargador de arranque). Puede consultar más documentación sobre CD-ROMs IDE y el soporte en Linux en `Documentation/cdrom/ide-cd`.

Si emplea una unidad SCSI debe antes verificar que Linux reconozca la tarjeta controladora. Después emplee el dispositivo adecuado para referenciar la unidad de CD-ROM (e.g `/dev/scd0` o `dev/scd1`).

5.4.1.8.2. Otras unidades de CD-ROM

Hay otras unidades que emplean una tarjeta controladora especial para conectarse al bus, o se conectan por medio de una tarjeta de sonido (aunque varias tarjetas de sonido recientes siguen el estándar IDE/ATAPI, así que las unidades conectados a estas se configuran como se presentó en la sección anterior).

`aztcd (CONFIG_AZTCD)`

: Aztech,Orchid,Okano,Wearnes. Parametros de arranque aztcd=dirbase. Dispositivo: `/dev/aztcd0`. Documentación en `Documentation/cdrom/aztcd`.

`gscd (CONFIG_GSCD)`

: Goldstar GCDR-420. Dispositivo: `/dev/gscd0`. Documentación: `Documentation/cdrom/gscd`.

`mcd, mcdx (CONFIG_MCD, CONFIG_MCD_IRQ, CONFIG_MCD_BASE, CONFIG_MCDX)`

: Ambos soportan unidades Mitsumi modelos LU-005, FX-001 y FX-001D, pero el primero no soporta CD multisesión, mientras el segundo tiene soporte experimental. Parámetros para insmod: `mcdx=0x300,11,0x304,5`. Dispositivos: `/dev/mcd` o `/dev/mcdx` (mcdx soporta hasta 4 drives con nombres de la forma `/dev/mcdx1`). Documentación: `Documentation/cdrom/mcd` y `Documentation/cdrom/mcdx`.

`cm206 (CONFIG_CM206)`

: Philips/LMS CM206 con tarjeta adaptadora CM206. Documentación: Documentation/cdrom/cm206. Autodetecta configuración. Parámetro del kernel o **insmod**:

cm206=0x300,11. Dispositivo: /dev/cm206cd.

optcd (CONFIG_OPTCD)

Optics Storage Dolphin 8000AT (interfaz de 34 pines). Dispositivo: /dev/optcd. Parámetro para cargador de arranque e insmod: optcd=0x340 (dirección base en lugar de 0x340). Documentación: Documentation/cdrom/optcd. Este CD-ROM puede estar conectado a una tarjeta de sonido ISP16, MAD16 o Mozart, en tal caso también debe instalarse isp16 (ver más adelante).

sjcd (CONFIG_SJCD)

Sanyo CDR-H94A. Documentación: Documentation/cdrom/sjcd. Dispositivo: /dev/sjcd (18 0). Parámetros del kernel sjcd=0x340 (dirección base en lugar de 0x340). Parámetros para insmod: sjcd_base=0x340. Este CD-ROM puede estar conectado a una tarjeta de sonido ISP16, MAD16 o Mozart, en tal caso también debe instalarse isp16 (ver más adelante).

isp16 (CONFIG_ISP16_CDI)

Permite configurar tarjetas de sonidos que usan los chips 82c928 o 82c929 como interfaces de CD-ROM. ISP16, MAD16, Mozart. Ha funcionado con CD-ROMs que emplean los módulos sjcd y optcd. Documentación: Documentation/cdrom/isp16. Parámetros para el kernel: isp16=0x340,11,5,Sanyo –no se requieren todos– remplazar 0x340 por dirección base (0x340, 0x320, 0x330 o 0x360), 11 por IRQ (3, 5, 7, 9, 10, 11 o 0 que indica no usar IRQ), 5 por DMA (0, 3, 5, 6 o 7) y Sanyo por tipo de unidad (noisp16, Sanyo, Panasonic, Sony o Mitsumi). Parámetros para insmod: isp16_cdrom_base=0x340 isp16_cdrom_irq=0 isp16_cdrom_dma=0 isp16_cdrom_type=Panasonic.

sbpcd (CONFIG_SBPCD)

Matsushita/Panasonic/Creative CR52x, CR56x, CD200, Longshine LCS-7260, TEAC CD-55A. Parámetros para el kernel: sbpcd=0x230, LaserMater con la dirección base en lugar de 0x230 y en lugar de LaserMater puede ser 0 o SoundBlaster (1), SoundScape (2) y Teac16bit (3). Parámetros para insmod: sbpcd=0x230,1 Dispositivo: /dev/sbpcd [76].

cdu31a (CONFIG_CDU31A)

Para las unidades Sony CDU31A/CDU33A. Documentación: Documentation/cdrom/cdu31a. No autodetecta, debe especificarse dirección base e IRQ (la descripción de la configuración del hardaware está en la documentación). Parámetros para el kernel: cdu31a=0x320,5 (remplazando 0x320 por la dirección base y 5 por la IRQ), si se usa con una tarjeta de sonido PAS-16 cdu31a=0x1f88,0,PAS. Parámetros para insmod: cdu31a_port=0x340, cdu31a_irq=5. Dispositivo /dev/sonycd (15,0).

sonycd535 (CONFIG_CDU535)

Sony CDU-531/535, CDU-510/515. Documentación: Documentation/cdrom/sonycd535. No debe activarse DMA, ni IRQ en la unidad, debe ser la unidad 1, la dirección base debe pasarse como parámetro del kernel: sonycd535=0x320. Si la unidad está apagado durante el arranque no será reconocida, pero puede después cargarse el controlador como módulo; el parámetro para insmod es también sonycd535=0x320. Dispositivo /dev/cdu535.

76. Este driver soporta hasta 4 unidades de CD-ROM los nombres son /dev/sbpcd1, /dev/sbpcd2 y /dev/sbpcd3

También existen unidades que se conectan al puerto paralelo, a tarjetas PCMCIA (en computadores portatiles) y a puertos USB. Puede consultase más sobre estas unidades en el CDROM-HOWTO.

5.4.1.8.3. Uso de CDs y DVDs

Dado que varios programas usan `/dev/cdrom` para comunicarse con la unidad de CD o DVD, ese archivo debería ser un enlace al dispositivo al que esté conectado su unidad (creelo por ejemplo con **ln -s /dev/hdb /dev/cdrom** remplazando `/dev/hdb` por el dispositivo de su unidad).

Para almacenar datos en un CD hay diversos sistemas de archivos, el más usado es ISO9660 (`CONFIG_ISO9660_FS` incluido por defecto en el kernel), para DVDs el sistema de archivos es UDF.

Es usual montar un CD o un DVD en `/cdrom`, para esto puede agregar al archivo `/etc/fstab` (ver Montaje y desmontaje de sistemas de archivos) la línea:

```
/dev/cdrom /cdrom iso9660 defaults,ro,user,noauto  0  0
```

que también especifica que no debe montarse durante el arranque el CD y que otorga permiso a los usuarios para montarlo y desmontarlo con **mount /cdrom** y **umount /cdrom** respectivamente.

En Debian por convención los usuarios que requieran emplear la unidad de CD-ROM o DVD para leer datos deben estar en los grupos `disk`, `cdrom` y `tape`, los que requieran escuchar CDs de música además deben estar en el grupo `audio`.

Para escuchar CDs de música no es necesario (ni posible) montar el CD antes, se debe emplear un programa como `workbone` que funciona en modo texto. Para abrir una unidad de CD-ROM puede emplear el comando `eject`.

5.4.1.9. Quemadora de CD

El kernel 2.2 y posteriores incluye soporte para quemadoras IDE y SCSI, el kernel 2.4 soporta algunas quemadoras USB: HP CD-Writer 82xx y otras basadas en ISD-200 como Sony CRX10U CD-R/RW, CyQ've CQ8060A CDRW.

Si su quemadora es SCSI será especialmente fácil de usar con el programa por excelencia para quemar CDs en ambientes Unix: **cdrecord**. Si la quemadora es IDE debe "emular" una quemadora SCSI; para lograrlo debe:

1. Activar los módulos del kernel: `ide-scsi` y `sg` (puede por ejemplo emplear `modconf` y buscar ambos módulos en la sección SCSI ver Kernel y módulos).
2. Indicar al kernel que para manejar la quemadora IDE emplee ide-scsi en lugar de un driver IDE usual. Esto puede hacerse pasando al kernel durante el arranque el parámetro `hdd=ide-scsi` (en vez de `hdd` el IDE al cual está conectada la quemadora ver Interfaz y dispositivos IDE). Puede pasar el parámetro durante el arranque desde el prompt de LILO, o puede configurar LILO para que lo pase automáticamente durante el arranque (ver Parámetros desde el cargador de arranque) (i.e agregando una línea cómo `append="hdd=ide-scsi"` en la sección apropiada y después ejectuando `/sbin/lilo`).
3. Finalmente, reinicie su sistema y vea entre los mensajes de arranque el número de drive SCSI que se le asigna a su quemadora (puede revisarlos posteriormente con **dmesg**) o averiguelo ejecutando

cdrecord -scanbus. Cree entonces el enlace simbólico `/dev/cdrom` o `/dev/quemadora` para que referencie el dispositivo SCSI asignado por el kernel, por ejemplo suponiendo que le hubiera asignado el 0 (caso que se da cuando no hay más dispositivos SCSI):

```
rm /dev/cdrom
ln -s /dev/scd0 /dev/cdrom
```

5.4.1.9.1. Quemado de CDs

Hay algunos programas gráficos para ayudarle a quemar un CD, estos programas emplean **cdrecord**. Este programa espera la especificación de la quemadora como dispositivo SCSI, puede ver la de su quemadora (una vez bien configurada) con **cdrecord -scanbus**.

cdrecord también permite controlar todos los detalles del proceso de escritura. A continuación presentamos dos ejemplos de su uso, suponiendo que su quemadora es reportada por cdrecord -scanbus como dispositivo 0,0,0 y que soporta velocidad 10x. Para transferir una imagen de un CD-ROM emplee:

```
cdrecord -v speed=10 dev=0,0,0 -data imagen.iso
```

Puede crear una imagen ISO a partir de un CD con:

```
dd
```

O puede crear una imagen ISO-9660 con extensión Rock Ridge a partir de la información de un directorio `/home/paz/cd` con:

```
mkisofs -R -o imagen.iso /home/paz/cd
```

Para montar una imagen ISO en el sistema de archivos en el directorio `/mtn/im` puede emplear:

```
mount -o loop -t iso9660 imagen.iso /mnt/im
```

y desmontarlo cuando no necesite ver más el contenido con **umount /mnt/im**.

Para transferir audio emplee:

```
cdrecord -v speed=10 dev=0,0,0 -data musica.wav
```

o para copiar un CD de audio (de acuerdo al manual de cdrecord):

```
cdda2wav -v255 -D2,0 -B -Owav
cdrecord -v dev=2,0 -dao -useinfo *.wav
```

5.4.1.10. Tarjetas de sonido

Una tarjeta de sonido puede convertir una sonido almacenado como señal digital, en una señal análoga que por medio de un parlante produce el sonido (conversos digital/análogo) , típicamente también puede convertir información análoga que recibe de un micrófono en una señal digital (conversor análogo digital) para almacenarla en un archivo en algún formato apropiado.

Algunas tarjetas de sonidos además pueden tener: mezclador para controlar fuentes y volumen, secuenciador MIDI, sintetizador FM, sintetizador Wavetable o ofrecer funcionalidad no directamente relacionada con sonido: interfaz para Joystick, interfaces IDE o para CD-ROMs (ver Unidades de CD-ROM y DVD).

Para la mezcla, reproducción y captura de audio una tarjeta normalmente emplea un codec (e.g manejado por ac97_codec) usa una línea IRQ, uno o dos canales DMA y dos direcciones E/S. Si la tarjeta tiene un secuenciador MIDI (e.g manejado por mpu401) puede requerir otra línea IRQ y una dirección base, si tiene un sintetizador FM (e.g manejado por opl3) requiere una dirección base (normalmente 0x388), si tiene un sintetizador Wavetable (e.g manejado por **awe_wave**) puede requerir 3 direcciones base.

En el momento de este escrito hay tres conjutos de controladores para tarjetas de sonido que le podrían ayudar a configurar manual o automáticamente su tarjeta de sonido (aunque hay tarjetas no soportadas por controlador alguno):

1. El que se distribuye con el kernel, cuyos controladores siguen normalmente el estándar Open Sound System (empleado en diversos sistemas Unix).
2. Los controladores del proyecto ALSA (la versión 0.4.1 está disponible en los paquetes `alsa-base`, `alsadriver`, `alsautils`, `alsaconf`), estos controladores siguen la arquitectura ALSA (particular para Linux). La herramienta `alsaconf` autodetecta y trata de configurar diversas tarjetas. Puede obtenerse una versión reciente (en el momento de este escrito la más reciente es 0.9) en: http://www.alsa-project.org aunque debe compilarse antes de usarse.
3. Los controladores comerciales del proyecto Open Sound System http://www.opensound.com

La configuración manual empleando los controladores del kernel se presenta más adelante en esta sección. Una vez configurado el sonido, puede usar (con el kernel 2.2 y con algunos controladores) **cat /proc/sound** o **cat /dev/sndstat** para ver los dispositivos reconocidos y su configuración. Puede verificar rápidamente si está funcionando la reproducción de sonido instalando el paquete **sox** y reproduciendo un sonido:

```
play /usr/share/sounds/login.wav
```

Algunas aplicaciones para sonido se presentan al final de esta sección.

> Las aplicaciones de audio pueden ser usadas por todo usuario de Debian que pertenezca al grupo `audio`

5.4.1.10.1. Controladores del kernel

Dado que los controladores de Linux siguen la arquitectura Open Sound System[77], algunos de los siguientes dispositivos podrán ser usados una vez configure su tarjeta:

`/dev/sndstat`

: Presenta información sobre los dispositivos de sonido en forma legible para humanos, por ejemplo con `cat /dev/sndstat`.

`/dev/dsp /dev/audio`

: Voz digitalizada, los datos enviados a esta interfaz son reproducidos por la tarjeta. Cuando se lee, el sonido capturado por la tarjeta de sonido es retornado. Mientras que `/dev/dsp` emplea codificación lineal de 8 bits sin signo, mientras que `/dev/audio` usa codificación logaritmica mu-law (por ejemplo empleada por el formato `.au` de Sun).

`/dev/mixer`

: Permite a los programas que lo usen controlar niveles de reproducción y grabación.

`/dev/sequencer /dev/music /dev/sequencer2`

: Acceso a sintetizadores FM y MIDI de la tarjeta.

`/dev/midi`

: Envio de datos a un sintetizador MIDI.

Para configurarla debe:

1. Identificar un módulo que pueda servir para el chip de su tarjeta, puede buscar en esta guía ayudandose del nombre de la tarjeta y preferiblement del nombre del chip principal (puede verlo en la tarjeta misma o en la documentación o si se trata de una tarjeta PCI puede consultar la identificación ver Buses). También puede consultar el Sound-HOWTO referenciado en las lecturas recomendadas de esta sección.
2. Si es una tarjeta ISA o ISA PnP, debe determinar los recursos de hardware que usa (sección ISA PnP ver Buses). Puede revisar el hardware y los manuales o la configuración en otro sistema operativo donde ya esté configurada.

Con estos datos puede intentar cargar el módulo apropiado y pasarle, de ser necesario, los parámetros de acuerdo a la configuración y al módulo. Si usa `modprobe` sólo necesitará especificar el controlador para el chip principal de su tarjeta y `modprobe` cargará automáticamente los otros (aunque no necesariamente todas las características opcionales ver Módulos).

Si la tarjeta es PCI normalmente será configurada durante el arranque y no necesitará especificar parámetros (sección PCI ver Buses) ni cargar el módulo, si la tarjeta es ISA PnP necesitará configurarla manualmente con `isapnptools` y cargar el módulo apropiado (sección ISA PnP ver Buses).

Los controladores del kernel están organizados en una jerarquía de módulos:

77. Los controladores de ALSA también pueden soportar OSS.

- Controladores generales que deben cargarse primero (`CONFIG_SOUND`): `soundcore` funciones para registrar dispositivos por tipo, `soundlow` controladores de bajo nivel que no hacen parte de OSS/Lite, `sound` funciones comunes requeridas por todos los módulos.
- Controladores específicos para cada tarjeta de sonido, dependen del chip principal de la tarjeta de sonido. Pueden requerir que otros módulos estén cargados.
- Controladores generales para soportar o emular características. Las tarjetas que incluyen un codec AC97[78], requieren que se carge primero `ac97_codec`. Las tarjetas compatibles con la Roland MPU-401, pueden emplear el módulo `mpu401` como secuenciador MIDI, este módulo puede recibir como parámetros: `irq`, `io`. Otras tarjetas con sintetizador midi pueden emplear bien `uart401` (que puede recibir como parámetros: `io`, `irq`) o el sintetizador MIDI `uart6850` (`CONFIG_SOUND_UART6850`) que puede recibir los mismo parámetros. Diversas tarjetas con sintetizador FM OPL3 emplean el módulo `opl3` (que debe cargarse de forma independiente).

A continuación listamos los controladores disponibles junto con el kernel 2.2.19 (incluido en Debian 2.2r6) , con algunas tarjetas que los usan y en algunos casos detalles de la configuración, la información ha sido extraida especialmente de las fuentes del kernel y su documentación.

C-Media PCI audio chip (CMI8338/8738)

(CONFIG_SOUND_CMPCI, módulo `cmpci`). Tarjeta PCI que usa el chip CMI8338 o CMI8337, este chip de cmedia (ID vendedor 0x13f6, ID dispositivos 0x0100, 0x101, 0x111) El controlador puede recibir los parámetros: `spdif_loop` (1 para habilitar ciclo SPDIF), `four_ch` (1 para habilitar modo de 4 canales), `rear_out` (1 si la tarjeta tiene un conector de salida separado), `mpu_io` `synth_io`. El controlador disponible en el kernel 2.4 soporta más características.

Crystal CS4281

(módulo `cs4281`). Integrado de Cirrus (ID Vendor 0x1013, ID Dispositivo 0x6005), el controlador no requiere parámetro alguno.

Crystal SoundFusion (CS4280/461x)

(CONFIG_SOUND_FUSION, módulo `cs46xx`). Son chips con codec AC97. El controlador puede recibir los parámetros: `external_amp=1` para activar soporte para Crystal EAPD, `thinkpad=1` para habilitar soporte especial para Thinkpad.

Creative SBLive (EMU10K1)

(`CONFIG_SOUND_EMU10K1`, módulo `emu10k1`) El integrado emu10k1 está en tarjetas PCI como Creative SBLive!, SB PCI512 o emu10k1. El controlador no requiere parámetros.

Ensoniq AudioPCI (ES1370) (`CONFIG_SOUND_ES1370`, módulo `es1370`)

Identificación PCI 1274:5000 (ver con lspci -n). Las tarjetas Sound Blaster 64/PCI pueden emplear este chip. El controlador de Linux no soporta `/dev/audio` (usar el programa `play`). Los posibles parámetros son: `joystick=1` para habilita puerto de joystick, `lineout=1` para convertir una línea de entrada en línea de salida, `micbias=1` para establecer desviación +5V para un microfono electret.

78. AC97 es un estándar para mezcla análoga/digital.

Ensoniq ES1371 (`CONIFG_SOUND_ES1371`, módulo `es1371`)

PCI ID 1273:1371 o 1273:5880. Las tarjetas Sound Blaster 64/PCI pueden emplear este chip. El controlador está más depurado en el kernel 2.2.21 o en la serie 2.4. (El controlador incluido en 2.2.19 no soporta bien 1273:5880). El controlador de Linux no soporta `/dev/audio` (usar `play`), sus posibles parámetros son: `joystick=<io>` para establecer dirección E/S del puerto de joystick, `spdif=1` para habilitar salida en modo digital S/PDIF.

ESS Maestro (`CONFIG_SOUND_MAESTRO`, módulo `maestro`)

Usado en tarjetas Maestro 1 (Vendedor ID 0x1285, Dispositivo 0x100), Maestro 2 (Vendedor ID 0x125D, Dispositivo 0x1968) y Maestro 2E (0x125D:0x1978). Los parámetros que este controlador puede recibir son: `debug=1` para habilitar mensajes de depuración [79], `dsps_order=2` para usar esta tarjeta con tarjeta de sonido secundaria, `use_pm` 0 o 1 para deshabilitar/habilitar manejo de potencia en la tarjeta Maestro 2E. Hay disponible más documentación en `Documentation/sound/MAESTRO`

ESS Maestro 3 (`CONFIG_SOUND_MAESTRO3`, módulo `maestro3`)

Tarjeta PCI Maestro 3 (0x1998 o 0x199a, vendedor 0x125D) y Allegro (vendedor 0x125D, dispositivo PCI ID 0x1988). Los parámetros para el controlador pueden ser: `debug=1` para habilitar mensajes de depuración, `global_dsp_speed=`*`velocidad`* por defecto 49, `external_amp` 1 o 0 para habilitar/deshabilitar amplificador externo. Más documentación en `Documentation/sound/MAESTRO3`.

ESS_SOLO1 (`CONFIG_SOUND_ESSSOLO1`,módulo `esssolo1`)

PCI ID 125D:1969. El controlador de Linux no soporta `/dev/audio` (usar `play`). Más información en `Documentation/sound/solo1`.

Intel ICH (810,820,440MX) (`CONFIG_SOUND_ICH`, módulo i810_audio)

En algunos computadores donde no se escucha bien pueden emplearse los parámetros `ftsodell=1` o `clocking=`*`frecuencia`*. Hay más documentación en el archivo `drivers/sound/i810_audio`.

S3 Sonic Vibes (`CONFIG_SOUND_SONICVIBES`, módulo `sonicvibes`)

PCI ID 5333:CA00. El controlador puede recibir los parámetros: `reverb=1` para habilitar reverberación, `wavetable=1` para habilitar sintetizador wavetable y `dmaio=`*`dir_ES`* para asignar dirección E/S inicial para DMA (si el BIOS no lo hace). Hay más documentación en `Documentation/sound/sonicvibes`.

Trident 4DWave-DX/NX (`CONFIG_SOUND_TRIDENT`,

módulo `trident`). Incluido en tarjetas Trident 4Dwave (PCI ID 1023:2000 o 1023:2001), Sis7018 (1039:7018), ALi 5451 (10b9:5451).

Turtel Beach MultiSound Pinnacle Fiji (`CONFIG_SOUND_MSNDPIN`, módulo `msdn_pinnacle`) y Turtle Beach MultiSound Classic, Tahiti, Monterrey (`CONFIG_SOUND_MSNDCLASS`, módulo `msnd_pinnacle`)

Requiere código para el firmware. El controlador requiere los parámetros: `io`, `irq` y `mem`, y puede recibir también: `write_ndelay`, `major`, `fifosize`, `calibrate_signal` (0 o 1), `digital`, `cfg`

79. Mensajes útiles para desarrolladores del controlador.

(para especificar puert 0x250, 0x260 o 0x270 cuando no opera en modo PnP), `reset`, `mpu_io`, `mpu_irq`, `ide_io0`, `ide_io1`, `ide_irq`, `joystick_io`. Documentación y utilidades en `Documentation/sound/MultiSound`.

Codec VIA 82C686 (`CONFIG_SOUND_VIA82Cxxx`, módulo

`via82cxxx_audio`) Normalmente incluido en placas base como Kinetiz 7E-A.

ProAudioSpectrum 16 support (`CONFIG_SOUND_PAS`, módulo `pas2`)

Soporta las tarjetas Pro Audio Spectrum 16, ProAudio Studio 16, Logitech SoundMan 16. El controlador puede recibir los siguientes parámetros: `io`, `irq`, `dma`, `dma16`, `sb_io`, `sb_irq`, `sb_dma`, `sb_dma16`, `joystick`, `symphony` que puede ser 0 o 1, `broken_bus_clock` que puede ser 0 o 1.

100% Sound Blaster compatibles (SB16/32/64, ESS, Jazz16) (`CONFIG_SOUND_SB`,módulo `sb`)

Soporta las tarjetas SoundBlaster, Thunderboard, SM Games, Avance Logic ALS-007, ESS, SB AWE 32, SB AWE 64, IBM Mwave [80]. Otras tarjetas que dicen ser compatibles con SoundBlaster normalmente no funcionarán con este controlador. Si la tarjeta es ISA Plug and Play debe usarse isapnptools para la configuración (ver Buses), un ejemplo de `/etc/isapnp.conf` sin comentarios es:

```
(CONFIGURE CTL0044/1132685 (LD 0
(INT 0 (IRQ 5 (MODE +E)))
(DMA 0 (CHANNEL 1))
(DMA 1 (CHANNEL 5))
(IO 0 (BASE 0x220))
(IO 1 (BASE 0x330))
(IO 2 (BASE 0x388))
(ACT Y)))
```

Cómo módulos podrían cargarse:

```
modprobe sound
insmod uart401
modprobe sb io=0x220 irq=5 dma=1 dma16=5 mpu_io=0x330
```

La mayoría de tarjetas SoundBlaster y compatibles soportan sintetizador FM con:

```
insmod opl3 io=0x388
```

Eventualmente si su tarjeta no soporta DMA de 16 bits, puede emplear como parámetro `dma16=-1`. Si su tarjeta soporta Wavetable, en `/etc/isapnp.conf` podría agregarse los puertos E/S 0xA20 y 0xE20 en la sección sobre WaveTable:

```
  (CONFIGURE CTL0048/58128 (LD 2
  #     ANSI string ->WaveTable<-
(IO 0 (BASE 0x0620))
(IO 1 (BASE 0x0A20))
(IO 2 (BASE 0x0E20))
(ACT Y)
  ))
```

80. En `Documentation/sound/mwave`, hay documentación particular para IBM Mwave que permite usarlo, cargando primero el firmware desde DOS.

y para cargar el controlador (después de cargar `sb`): `insmod awe_wave`. El módulo `awe_wave` puede recibir los parámetros: `io=`*`dir_ES`* puerto E/S base de Emu8000, `memsize=`*`mem`* cantidad de memoria DRAM de la tarjeta en Kbytes. Hay más documentación en `Documentation/sound/AWE32`, `INSTALL.awe`, `README.awe`. El mismo controlador `sb` puede usarse con algunas compatibles empleando las opciones:

`mad16=1 , trix=1 , pas2=1 , sm_games=1`

para Logitech soundman

`acer=1`

para detectar tarjetas en algunos portatiles ACER

`esstype=`*`tipo`*

De acuerdo a Documentation/sound/ESS puede tener los siguientes valores: -1 sólo autodetectar ES688 o ES1688, 0 intentar autodetectar el chip ESS (puede fallar para ES1688), 688 chip ES688, 1788 chip ES1788, 1688 chip ES1688, 1888 chip ES1888, 1868 chip ES1868, 1869 chip ES1869, 1887 chip ES1887 (soporta Full Duplex). ES1868 es compatible con 1688 PnP, debe configurarse con `isapnp`:

```
(CONFIGURE ESS1868/-1 (LD 1
(IO 0 (BASE 0x0220))
(IO 1 (BASE 0x0388))
(IO 2 (BASE 0x0330))
(DMA 0 (CHANNEL 1))
(INT 0 (IRQ 5 (MODE +E)))
(ACT Y)
```

El mismo driver soporta tarjetas AVance Logic ALS007 y Avance Logic ALS100. Ambas se configuran con isapnptools, dejando un DMA de 8 bits (0,1 o 3), pero DMA de 16 bits debe dejarse en -1. Más documentación en `Documentation/sound/ALS`.

Gravis Ultrasound (`CONFIG_SOUND_GUS`, módulo `gus`)

Tarjetas GUS y GUS MAX. El controlador puede recibir como parámetros: `io`, `irq`, `dma`, `dma16`, `type=1` para indicar que es PnP, `gus16`, `no_wave_dma`, `db16=1` para habilitar Gus16 AD1848.

Personal Sound System (ECHO ESC614) (`CONFIG_SOUND_PSS`, módulo `pss`)

Funciona con las tarjetas Orchid SW32, Cardinal DSP16, Beethoven ADSP-16, PSS y otras basadas en ECHO. Los posibles parámetros del módulo son: `pss_io=0x220` puerto E/s base (0x240), `mss_io=0x530` o 0x604 o 0xE80 o 0xF40 WSS (audio) E/S base, `mss_irq=3` o 5 o 7 o 9 o 10 o 11 o 12 WSS (audio) IRQ, `mss_dma=0` 1 o 3 WSS (audio) DMA, `mpu_io=0x330` puerto E/S base para Midi, `mpu_irq=3` o 5 o 7 o 9 o 10 o 11 o 12, IRQ Midi, `pss_cdrom_port=0x340` puerto base para el CDROM PSS, `pss_enable_joystick=1` o 0 para habilitar/deshabilitar puerto de Joystick en tarjeta PSS, `pss_no_sound=1` para no configurar componentes de sonido, útil para quienes sólo quieren configurar CDROM o Joystick de la tarjeta, `pss_keep_settings=1` para mantener configuración de hardware después de descargar módulo, `pss_firmware=/etc/sound/pss_synth` localización del archivo con el firmware (está en los discos de instalación para DOS), `pss_mixer=1` o 0 para habilitar o deshabilitar mezclador PSS. Más documentación en `Documentation/sound/PSS` y `Documentation/PSS-updates`.

Microsoft Sound System (`CONFIG_SOUND_MSS`, módulo `ad1848`)

Funciona con varias tarjetas en particular Windows Sound System, Aztech SG 16 Pro o NX16 Pro, CMI8330 (ver Documentation/sound/CMI8330). El codec ad1848/cs4248 es un chip usado en el MS Sound System, compatible con CS4231 (usada en GUS MAX) y CS4231A, AD1845, CS4232, CS4232A, Los parámetros, junto con valores típicos son: `io=0x530`, `irq=11`, `dma=0` canal primario, `dma2=5` canal secundario, `type=1` si no tiene registros IRQ/DMA, `deskpro_xl=1` habilitar soporte especial para Deskpro XL, `deskpro_m=1` habilitar soporte especial para Deskpro M, `soundpro=1` habilitar soporte especial para chips SoundPro (ver Documentation/sound/SoundPro).

Ensoniq SoundScape (`CONFIG_SOUND_SSCAPE`, módulo `sscape`)

Algunas tarjetas de Ensoniq, Spea y Reveal. El controlador puede recibir los parámetros: `dma`, `irq`, `io`, `spea` (1 o 0), `mpu_irq`, `mpu_io`, `mss` (si es 1 habilita soporte para Microsoft Sound System).

MediaTrix AudioTrix Pro (`CONFIG_SOUND_TRIX`, módulo `trix`)

El controlador recibe los parámetros: `io`, `irq`, `dma`, `dma2`, `sb_io`, `sb_dma`, `sb_irq`, `mpu_io`, `mpu_irq`, `joystick` (si es 1 habilita interfaz para Joystick).

OPTi MAD16, Mozart (`CONFIG_SOUND_MAD16`, módulo `mad16`)

Mozart (OAK OTI-601), MAD16 (OPTi 82C928, 82C929, 82C931). Un uso típico es:

```
modprobe mad16 io=0x530 irq=7 dma=0 dma16=1 mpu_io=816 mpu_irq=5
```

además puede recibir los siguientes parámetros: `joystick=0` (deshabilitado) o 1 (habilitado), `cdtype=0x00` (deshabilitado), 0x02 (Sony CDU31A), 0x04 (Mitsumi), 0x06 (Panasonic), 0x08 (Secondary IDE) 0x0a (Primary IDE), `cdirq=0` (deshabilitado), 3 (IRQ3), 5,7,9,10,11, `cddma=0` (deshabilitado), 1,2,3 (DMA3 para Sony o Panasonic), 5, 6,7, (DMA5 para Mitsumi o IDE), `opl4=0` (OPL3), 1 (OPL4). Más documentación en `Documentation/sound/MAD16` y `Documentation/sound/Opti`.

TurtleBeach WaveFront

(CONFIG_SOUND_WAVEFRONT, módulo `wavefront`), Tropez Plus, Tropez o Maui. Una configuración típica de una tarjeta Tropez+ es:

```
modprobe wavefront io=0x200 irq=9
modprobe cs4232 synthirq=9 syntio=0x200 io=0x530 irq=5 dma=1 dma2=0
modprobe opl3 io=0x388
```

El controlador `wavfront` también puede recibir los parámetros: `io`, `irq`, `wf_raw=1` para evitar revisión de estado limpio antes de cargar firmware, `fx_raw=0` para evitar limpiar después de cargar firmware, `debug_default` para depuración, `wait_usecs=150`, `sleep_interval=100`, `sleep_tries=50`, `ospath=/etc/sound/wavefront.os` vía donde encontrar firmware, `reset_time=2`, `ramcheck_time=20`, `osrun_time=10`. Más documentación en `Documentation/sound/TROPEZ+`. Para tarjetas Maui o Tropez también puede emplearse el módulo `maui` que recibe los parámteros: `io` e `irq`.

Crystal CS4232

(CONFIG_SOUND_CS4232), tarjetas basada en CS4232, usan Plug and Play. Un uso típico es:

```
modprobe cs4232 io=0x534 irq=5 dma=3 dma2=0
```

Además de los mostrados recibe los parámetros `mpuio`, `mpuirq`, `syntirq`, `syntio`. Para tener sintesis FM: `insmod opl3 io=0x388` Ver también TurtelBeach WaveFront

Yamaha OPL3-SA2, SA3 y SAx PnP (`CONFIG_SOUND_OPL3SA2`, módulo `opl3sa2`)

El controlador recibe los parámetros: `io` para establecr dirección E/S base (e.g 0x370), `mss_io` establece base E/S de MSS (audio) (e.g 0x530, 0xE80, las direcciones posibles deben terminar en 0 o 4 y deben estar entre 0x530 y 0xF48), `mpu_io` base E/S para MIDI (e.g 0x330, debe ser una dirección entre 0x300 y 0x334), `irq` línea IRQ para irq MSS (5, 7, 9, 10, 11, 12), `dma` primer canal DMA MSS (0, 1, 3), `dma2` segundo canal DMA MSS (0, 1, 3).

Aztech Sound Galaxy (no PnP) (`CONFIG_SOUND_SGALAXY`, módulo `sgalaxy`)

Soporta Waverider Pro 32 - 3D y Galaxy Washington 16. El controlador puede recibir los parámetros: `io`, `irq`, `dma`, `dma2`, `sgbase` (para especificar la dirección E/S básica).

Tarjetas basadas en AD1816

(`CONFIG_SOUND_AD1816`, módulo `ad1816`) Tarjeta basada en el chip Analog Devices AD1816, algunas tarjetas que lo usan son: Terratec Base 1, Terratec Base 64, HP Kayak, Acer FX-3D, SY-1816, Highscreen Sound-Boostar 32 Wave 3D, Highscreen Sound-Boostar 16 , AVM Apex Pro card y las siguientes pasando el parámetro `option=1` al cargar el módulo: Aztech SC-16 3D, Newcom SC-16 3D, Terratec EWS64S. Cuando este controlador se incluye estáticamente puede configurarse con un parámetro del kernel de la forma: `ad1816=0x530,5,1,3`, remplazando los valores por la dirección E/S base, línea IRQ, primer y segundo canales DMA. Como módulo los parámetros podrían ser:

```
modprobe ad1816 io=0x530 irq=5 dma=1 dma2=3 ad1816_clockfreq=33000
```

Debe usarse `isapnp` para configurarla, líneas como las siguientes son relevantes:

```
(INT 0 (IRQ 5 (MODE +E)))
(DMA 0 (CHANNEL 1))
(DMA 1 (CHANNEL 3))
(IO 0 (BASE 0x0220))
(IO 1 (BASE 0x0388))
(IO 2 (BASE 0x0530))
```

Más documentación en el kernel `Documentation/sound/AD1816`.

Yamaha OPL3-SA1 (`CONFIG_SOUND_OPL3SA1`, módulo `opl3sa`)

Para emplear este chip en modo WSS (Windows Sound System) puede usarse:

```
modprobe  opl3sa io=0x530 irq=11 dma=0 dma2=1 mpu_io=0x330 mpu_irq=5
```

también podrían cargarse módulos para soportar sintetizador FM (opl3 o SoftOSS) con:**insmod op3 io=0x388r** o **insmod softoss**. Más documentación en `Documentation/sound/OPL3-SA`.

NeoMagic NM256AV/NM256ZX (`CONFIG_SOUND_NM256`, módulo `nm256`)

Chip NeoMagic 256AV/256ZX, usado en laptops como: Sony Z505S/SX/DX/RX, F150, F160, F180, F250, F270, F28, PCG-F26, Dell Latitude CPi, CPt. El controlador soporta los parámetros: `usecache`, `buffertop`, `nm256_debug`, `force_load`. Más documentación en `Documentation/sound/NM256`.

Adlib (`CONFIG_SOUND_ADLIB`, módulo `adlib_card`)

El controlador recibe la dirección E/S base con el parámetro `io`.

Yamaha YMF7xx PCI (`CONFIG_SOUND_YMFPCI`, módulo `ymfpci`)

Tarjetas YMF711, el controlador puede recibir como parámetros: `mpu_io` y `synth_io`.

Tarjetas Gallant (SC-600, SC-6600) (`CONFIG_SOUND_AEDSP16`, módulo `aedsp16`)

Para tarjetas que incluyen el DSP Audio Excel. El controlador establece los recursos del DSP desde Linux, un uso típico es:

```
modprobe aedsp16 io=0x220 irq=5 dma=1
```

También podría pasar como parámetros: mss_base (e.g 0x530 o 0xE80), mpu_base (0x300, 0x310, 0x320 o 0x330) mpu_irq (5, 7, 9, 10 o 0).

5.4.1.10.2. Programas que emplean la tarjeta de sonido

Hay gran variedad algunos gráficos otros para consola, puede buscarlos con **apt-cache search sound | less** (ver Paquetes en Debian).

sox

Incluye el program sox que permite convertir de un formato de sonido a otro, play que permite reproducir un sonido almacenado prácticamente en cualquier formato y rec que permite grabar un sonido. Por ejemplo para convertir de formato raw (PCM 11025 Hetz, 16 bits) a .wav:

```
sox -r 11025 -s -w voz.raw voz.wav
```

playmidi

Útil si su tarjeta incluye un sintetizador MIDI que ha logrado configurar. Incluye los programas playmidi (consola) y xplaymidi (X11), así como algunos ejemplos. Si su tarjeta no tiene sintetizador MIDI puede emplear Timidity.

`timidity timidity-patches`

Emula sintetizador MIDI por software.

aumix

Mezclador modo texto, se inicia con **aumix**.

soundtracker

Editor y reproductor gráfico de archivos MOD para Amiga.

xmms

Aplicación gráfica para reproducir MP3 y otros formatos.

5.4.2. Lecturas recomendadas: Dispositivos y Linux

- Para conocer más sobre el teclado y su configuración puede consultar las páginas **man** de keymaps, loadkeys, dumpkey, showkey, kbd_mode, setleds, kbdrate. Si desea ver diversos tipos de teclados soportados por Linux y sus scancodes puede consultar en Internet: http://www.win.tue.nl/~aeb/linux/kbd/scancodes.html
- Hay más detalles sobre configuración de Bus Mouse en el *Busmouse-HOWTO*, disponible en Internet en: http://www.ibiblio.org/pub/Linux/docs/HOWTO/unmaintained/Busmouse-HOWTO Y sobre la configuración del mouse con XFree86 3.3.6 puede consultar "*Mouse Support in XFree86*": http://www.xfree86.org/3.3.6/mouse.html
- Puede aprender más sobre las posibilidades de internacioanlización de teclado y pantalla del kernel consultado las páginas **man** de: unicode, console, charsets.
- Puede aprender más sobre las secuencias de control de la consola de Linux consultado la página `man` de: console, console-codes. Y el Keyboard and Console HOWTO, disponible en su sistema en `/usr/share/doc/HOWTO/en-txt/Keyboard-and-Console-HOWTO.txt.gz` y en Internet en: http://www.tldp.org/HOWTO/Keyboard-and-Console-HOWTO.html
- Para configurar impresoras (incluyendo seriales y algo de USB): http://www.linuxdoc.org/HOWTO/Printing-HOWTO.html y las páginas `man` de `lpd`, `lpr` y `printcap`. Dado que `lpd` es un sistema de impresión común en Linux y otros sistemas Unix, también puede consultar sobre este tema en el manual de FreeBSD: http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/printing.html
- Los discos SCSI en general son más veloces que los discos IDE, aunque son más costosos. Puede conocer más sobre características que afectan la velocidad de un disco en: http://www4.tomshardware.com/storage/97q3/970728/index.html
- Puede consultarse información completa sobre fdutils (y superformat) con **man superformat**, **info fdutils** o en Internet en: http://www.tux.org/pub/knaff/fdutils/Fdutils.html La información completa sobre **mtools** con **info mtools** o en Internet en: http://mtools.linux.lu/mtools.html#SEC1
- Para conocer más sobre quemadoras de CD y su configuración en Linux puede consultar El *CD-Writing HOWTO* que debe estar en su computador en `/usr/share/doc/HOWTO/en-txt/CD-Writing-HOWTO.txt.gz`, o en Internet en http://www.linuxdoc.org/HOWTO/CD-Writing-HOWTO.html
- Puede consultar más sobre CD-ROMs en Linux en las fuentes del kernel, los archivos del directorio Documention/cdrom.

> **Aviso**
>
> Esa documentación ha sido escrita en diversos momentos del desarrollo del kernel e infortunadamente no siempre ha sido actualizada. La distribución Debian incluye soporte como módulos para todos los controladores de CD-ROM, no debe ser necesario recompilar el kernel (excepto para algunos módulos que no pueden ser configurados desde el arranque ni con parámetros para **insmod**). Los dispositivos para todos estos módulos son creados automáticamente durante la instalación de Debian. También puede consultar *CDROM-HOWTO* en su sistema `/usr/share/doc/HOWTO/en-txt/CDROM-HOWT.gz` o en Internet: http://www.tldp.org/HOWTO/CDROM-HOWTO/index.html.

- Encontrará más información para configurar una tarjeta de sonido en el *Sound-HOWTO*, disponible localmente en `/usr/share/doc/HOWTO/en-text/Sound-HOWTO.gz` o en Internet: http://www.linuxdoc.org/HOWTO/Sound-HOWTO.html y para emplear su tarjeta de sonido en el Sound-Playing-HOWTO: `/usr/share/doc/HOWTO/en-text/Sound-Playing-HOWTO.gz` o en Internet: http://www.linuxdoc.org/HOWTO/Sound-Playing-HOWTO.html. Si desea conocer más sobre los dispositivos de sonido (`/dev/dsp`, `/dev/mixer`) y como programarlo puede consultar la guía de programación del Open Sound System disponible en: http://www.opensound.com/pguide/index.html
- Si necesita información adicional para configurar el sistema X-Window, puede consultar el FAQ de XFree86 disponible en http://www.xfree86.org/FAQ/, o las secciones 5.3 a 5.7 de "*Linux Installation and Getting Started. The X Window System*." http://www.linuxdoc.org/LDP/gs/node7.html#SECTION00700000000000000000 Puede consultar un glosario corto sobre terminos relacionados con monitores y tarjetas graficadoras en: http://www.algonet.se/~lear/modes/glossary.html#freq

5.4.3. Ejercicios: Dispositivos y Linux

1. Continue el ejercicio 1 de la sección anterior, completando el script para que reporte:

- Marca y referencia de cada disco duro. Capacidad de cada uno y forma como cada uno está particionado. (Ayuda: **fdisk -l**)
- Marca y referencia de la tarjeta de video, así como el nombre del controlador que Linux emplea. (Ayuda: `/etc/X11/XF86Config`).
- Marca y referencia del monitor, así como los parámetros de configuración del mismo para X-Window. (Ayuda: `/etc/X11/XF86Config`).
- Marca del ratón, cantidad de botones, tipo de puerto al que está conectado y recurso que emplea (puerto COM en caso de que sea serial).
- En caso de que tenga tarjeta de sonido, marca y referencia de la misma junto con los recursos de hardware que emplea (IRQs, DMA, puertos). Módulo que usa y parámetros que recibe el módulo.
- Si el computador tiene modem, marca y referencia, así como recursos de hardware que emplea (puerto

serial, IRQ). Si el modem requiere controladores especiales, también nombre y los parámetros de los mismos.

- Marca y referencia de otros dispositivos externos que estén conectados al computador, así como controladores y parámetros de configuración. (e.g. de scanners, unidades para quemar CD-ROM, camaras, etc.).
- Si el computador está conectado a una red, también es importante que el script extraiga:
  - Marca y referencia de la tarjeta de red así como los recursos de hardware que emplee (IRQ, DMA, puertos). De ser necesario (si el kernel no lo configura automáticamente), el nombre del controlador y los parámetros que requiera.
  - Dirección IP del computador y nombre.
  - Dirección del servidor DNS, del gateway y mascara de red.

2. Además del algo de información en estas guías y del Hardware-HOWTO, hay algunas bases de datos de hardware soportado por Linux. Busque alguna(s) y comparta la información con sus compañeros.

3. Opcional: busque controladores para Linux de algunos componentes de su computador en cada casa que hace el hardware, en caso de que no encuentre, es recomendable que les envíe un correo preguntando por el controlador específico. Si ellos no tienen el controlador, tendrán la oportunidad de enterarse de la necesidad del controlador para el hardware que usted está utilizando, y los instará a crearlo o liberarlo al público.

5.5. Instalación de Debian

Indicadores de Logros:

- Puede instalar Debian Linux siguiendo los lineamientos de la plataforma de referencia (verPlataforma de referencia).

Hay diversas distribuciones de Linux, cada una con debilidades y fortalezas propias. Nuestra recomendación es la distribución Debian 2.2. Que si bien no es la distribución más fácil de instalar, ofrece a cambio otras ventajas:

- Es una distribución de caracter social, que antepone los intereses de la comunidad a la publicidad o el beneficio de los desarrolladores.
- Técnicamente está muy bien afinada y provee un excelente manejador de paquetes.
- Con respecto a otras distribuciones ampliamente conocidas, es la única sobre cuyo futuro podemos decidir, pues su desarrollo es abierto a quién quiera participar y es democrático.

5.5.1. Lectura: Instalación de Debian

- Antes de iniciar la instalación haga un listado del hardware que tiene disponible y de los recursos de hardware que emplean (verHardware básico y Linux).
- Planee la forma como distribuirá el espacio en su(s) disco(s) duro(s), las particiones que destinará para los directorios de Linux y la forma como arrancará Linux y eventualmente otro(s) sistema(s) operativo(s) (verDiscos duros). Tenga en cuenta que podrá configurar el cargador de arranque para cargar más de un sistema operativo, ubicados en distintas particiones o en distintos discos, pero que con algunas BIOS la información para arrancar Linux debe estar en los primeros 1024 cilindros de un disco (verDiscos duros)
- De requerirlo cree espacio para las particiones que planeó, si tiene DOS o Windows puede defragmentar y emplear el programa `fips` (disponible en el directorio `/tools` del CD 1), si tiene Linux puede crear espacio con el program parted (Cambio del tamaño de un sistema de archivos ver Discos duros).
- De los métodos de instalación posibles el más sencillo es usar un CD-ROM. Si su BIOS lo permite, configúrela para arrancar el sistema por el CD-ROM y emplee el CD 1 de Debian. Si su BIOS no lo permite o si requiere otro método de instalación cree dos disquetes (de 1.44) con las imagenes: `rescue.bin` y `root.bin`[81] disponibles en el CD 1 en el directorio `/install`. Por ejemplo para copiar la imagen `rescue.bin` desde DOS/Windows en un disquete (unidad A en terminología DOS), puede emplear el programa `rawrite2` disponible en el mismo directorio `/install` del CD 1:

  ```
  rawrite2 -f rescue.bin -d a
  ```

  Desde Linux y otras versiones de Unix podría usar **dd**:

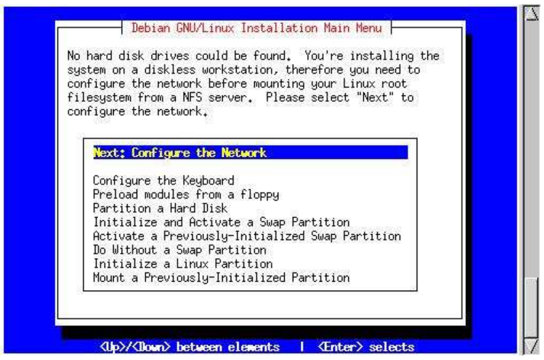
  ```
  dd if=rescue.bin of=/dev/fd0 bs=1024
  ```

  Después arranque el sistema iniciando por el disquete con `rescue.bin`.

81. Estos disquetes permiten iniciar un sistema con hardware usual, hay otros disquetes para sistemas con discos o dispositivos no tan comunes, ver manual de instalación.

Este disquete permitirá iniciar un sistema Linux mínimo para iniciar una instalación o para recuperar un sistema que pueda fallar [82].

- El programa de instalación le permitirá configurar un teclado sencillo, (`qwerty/us` o `qwerty/es`) después de completar la instalación podrá refinar su elección (verTeclado).



- En la mayoría de computadores no requerirá cargar módulos adicionales para realizar la instalación, esto podría ser necesario si desea hacer una instalación en red (para configurar la tarjeta) o si su unidad de CD-ROM no es IDE/ATAPI (verUnidades de CD-ROM y DVD).
- Durante la instalación eligiendo "Partition a Hard Disk" podrá emplear el programa `cfdisk` (verDiscos duros), para dividir el espacio que abrió y marcar las particiones nuevas como Linux (tipo 83) y una como Linux swap (tipo 82). Con `cfdisk` para dividir una partición en dos, primero debe borrarla y después crear dos nuevas.
- Formatee cada partición Linux para que pueda usar el sistema de archivos ext2, para esto eliga la opción del menú "Initialize a Linux Partition" (verSistema de archivos ext2). Podrá especificar también el directorio en el que desea montar cada partición que inicialice (si tiene otras particiones Linux en algunos de sus discos también podría escoger directorios para montarlas con "Mount a Previously-Initialized Partition" ver Montaje y desmontaje de sistemas de archivos). Formatee y active la partición para swap con la opción "Initialize and Activate a Swap Partition".
- Cuando termine de inicializar y activar todas las particiones instale el kernel y los módulos con la opción del menú "Install Operating System Kernel and Modules", después podría cargar algunos módulos de acuerdo a los dispositivos de su computador[83], pero esta operación puede aplazarse hasta tener un sistema mínimo funcionando —podrá realizarla posteriormente con el programa `modconf` (verMódulos).
- Si el computador que instala funcionará en red LAN (por ejemplo en una red de computadores de un colegio), es recomendable que intente cargar el módulo apropiado para la tarjeta de red, así el programa de instalación le permitirá especificar más información sobre la red. Eventualmente puede querer emplear los siguientes datos generales:

  `Dirección IP: 192.168.0.1`

82. Para recuperar un computador con una partición Linux, puede emplear el disquete de arranque, después de configurar el teclado pase a la segunda consola virtual, monte el disco duro con la raiz de Linux e.g `mount /dev/hda3 /mnt`, establezcalo como nueva raíz `chroot /mnt` y haga los ajustes necesarios para recuperar el sistema (e.g editar `/etc/lilo.conf` y ejectuar `/sbin/lilo`.
83. Entre los módulos presentados en la sección misc, dice `sb` "VIA 82Cxxx audio support", debería ser `sb` "SoundBlaster and 100% compatible support".

```
Netmask: 255.255.255.0
Gateway: 192.168.0.1
Servidor DNS: 192.168.0.1
```

Con esta configuración puede escoger cualquier nombre para su máquina y su dominio.

- Si elige instalar el cargador de arranque (opción del menú "Make Linux Bootable directly from H.D"), podrá instalar LILO bien en el MBR o en el primer sector de la partición `/`. Si tiene sólo Windows en su disco puede ser mejor instalarlo en el MBR —cuando reinicie entrará a Linux y podrá configurar LILO para permitirle entrar también a Windows, ver más adelante—, si tiene otras particiones con Linux que usen LILO o algún otro cargador de arranque puede dejar LILO en el sector de arranque de `/` y configurar el otro cargador de arranque (ver Cargador de arranque ver Discos duros).
- Cuando reinicie el computador entrará al Linux que instaló y el programa de instalación continuará. Por seguridad es mejor emplear claves MD5 y Shadow (verUsuarios). Al elegir las tareas que realizará con su sistema es recomendable que instale herramientas de desarrollo (para compilar programas ver Compilación e instalación de programas a partir de fuentes). Si por algún motivo se interrumpe o usted desea reiniciar el proceso de instalación, use desde la cuenta root:

  ```
  dpkg-reconfigure base-config
  ```

  (paquete `base-config`), las acciones realizadas durante la instalación se registran en la bitacora `/var/log/installer`. También podrá entrar al programa `dselect` para refinar aún más la selección de paquetes, o para instalar paquetes de otras fuentes (ver Paquetes en Debian)
- Mientras se instalan y configuran los paquetes, tendrá que intervenir en ocasiones, por ejemplo para configurar X-Window. Al igual que con módulos para dispositivos, si no resulta fácil, no es indispensable que lo configure durante la instalación (para configurar X-Window posteriormente vea Monitor y tarjeta graficadora). En ocasiones la instalación de paquetes no se completa, es recomendable que entre al programa `dselect` y eliga la opción `Install`.
- Una vez la instalación de paquetes concluya, si instaló LILO en el MBR puede configurarlo para permitirle entrar a otros sistemas operativos. Por ejemplo si tiene Windows en la partición `/dev/hda1` (ver Discos duros), edite al archivo `/etc/lilo.conf` y agregue:

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

  cambiando dos por la etiqueta con la que desee entrar al otro sistema, después ejecute `/sbin/lilo`.
- Para completar la instalación puede:
  - Instalar otros paquetes que pueda requerir, configurando de ser necesario otras fuentes en `/etc/apt/sources.list`, por ejemplo si tiene conexión a Internet:

    ```
    deb http://non-us.debian.org/debian-non-US stable/non-US main contrib non-free.
    ```

    (ver Paquetes en Debian).
  - Refinar configuración del teclado para español (ver Teclado en español)
  - Instalar locale para Colombia y configurar mensajes en español (ver Locale para Colombia y mensajes en español)
  - Configurar X-Window (verMonitor y tarjeta graficadora).
  - Cargar controladores que requiera, por ejemplo para tarjeta de sonido (verTarjetas de sonido).

- Configurar otros dispositivos/servicios como impresión (verImpresora).
- Contarle a quienes desarrollan esta guía sobre su experiencia `structio-guias@lists.sourceforge.net`.

5.5.2. Lecturas recomendadas: Instalación de Debian

Recomendamos de forma especial *Instalación de Debian Linux 2.2 para Intel x86* http://www.debian.org/releases/stable/i386/install.es.html

Además de otras secciones de este capítulo, en caso de que las necesite, sugerimos algunas lecturas que pueden ayudarle a tomar decisiones al instalar Linux en su equipo:

- Durante el proceso de instalación puede necesitar conocer mejor su disco duro y el esquema de Linux para nombrar este tipo de dispositivos: The Linux System Administrators' Guide (version 0.6.2), Capítulo 4. Using Disks and Other Storage Media http://www.linuxdoc.org/LDP/sag/x754.html
- Para entender como está organizado lógicamente su disco, puede consultar The Linux System Administrators' Guide (version 0.6.2), capítulo 4. Using Disks and Other Storage Media http://www.linuxdoc.org/LDP/sag/x885.html
- Para determinar como asignar espacio en la partición `swap`: The Linux System Administrators' Guide (version 0.6.2), capítulo 5. Memory Management http://www.linuxdoc.org/LDP/sag/x1546.html

5.5.3. Ejercicios: Instalación de Debian

1. Verifique que puede teclear y ver los carácteres de español en las consolas virtuales y en X-Window, verifique que los mensajes se presenten en español.

2. Compruebe que todos los dispositivos de su computador funcionan. En particular las unidades de disquete y CD-ROM (recuerde cambiar el archivo `/etc/fstab` para que todos los usuarios puedan montar esos dispositivos, ver Disquetes, ver Unidades de CD-ROM y DVD).

3. Examine la memoria de su computador (comando **free**), verifique que el espacio swap esté siendo usado.

4. Revise los programas que están corriendo en su máquina. Revise la secuencia de inicialización. Elimine de esta secuencia programas que no vaya a utilizar (si los deja estará desperdiciando memoria).

5. Verifique que esté instalada la documentación y todos los programas recomendados en la plataforma de referencia (ver Plataforma de referencia).

6. En `/usr/share` busque los archivos de configuración de emacs, vea los directorios de las fuentes y de ser posible instale paquetes que puedan ser útiles para los usuarios finales (e.g w3, gnus).

7. Opcional: ¿Es posible generar su propia distribución o elegir por lo menos paquetes para hacer la instalación de una red con software homogéneo? ¿Qué distribución le permite hacer esto con facilidad?

5.6. Localización y configuración de algunos programas

Indicadores de Logros:

- Configura dispositivos y programas para usarlos en Colombia.

5.6.1. Lectura: Localización y configuración de algunos programas

Por su caracter abierto y por la diversidad de quienes lo desarrollan, Linux puede configurarse para ser usado en varios países, con alfabetos diferentes, dispostivos diferentes y gustos diferentes. En esta sección se presentan ideas para configurar mejor Debian y usarlo en Colombia, así como detalles sobre la configuración de diversos programas. Sin embargo una buena fuente de consulta puede ser la documentación que acompaña a cada program y que normalmente quedará en un subdirectorio de `/usr/share/doc`. Puede examinar exactamente que documentación y archivos son instalados por un paquete con **dpkg -L linux-util** cambiando `linux-util` por el nombre del paquete (ver Paquetes en Debian)

5.6.1.1. Configuración del teclado

Si la distribución es latinoamericana o española basta que configure el mapa del teclado (ver Dispositivos y Linux) y el teclado en X-Window (la configuración general ver Dispositivos y Linux o para usuarios ver Configuración de una sesión).

Si tiene un teclado US recomendamos que emplee el mapa de teclado `/usr/share/keymaps/i386/qwerty/us-latin1.kmap.gz` que permite generar ¿ con **AltGr+/**, ! con **AltGr+1**, ñ y Ñ con **AltGr+n** y **AltGr+N**, y vocales tildadas con **AltGr+*vocal***. Sugerimos que edite ese mapa (descomprimalo, editelo y pruebe los cambios con `loadkeys`), modificando:

```
keycode  40 = dead_acute dead_diaeresis dead_acute dead_acute Control_g
```

así cómo:

```
keycode  41 = grave            dead_tilde
```

y agregando (por ejemplo al final):

```
compose '~' 'n' to 'ñ'
compose '~' 'N' to 'Ñ'
compose '"' 'u' to 'ü'
compose '"' 'U' to 'Ü'
```

Para configurar como tecla muerta **'** para la tilde, **~** para la virgulilla y **"** para la dieresis (para producir los caracteres de español al hacer la edición puede usar digrafos de vim ver Lecturas Editores de texto).

Puede dejar su cambio de forma permanente comprimiendolo y copiandolo en `/etc/console-tools/default.kmap.gz` (ver Secuencia de inicio).

5.6.1.2. *Locale* para Colombia y mensajes en español

Para poder ingresar y ver caracteres del español desde programas que empleen la librería readline (por ejemplo bash) edite el archivo `/etc/inputrc` y agregue las siguientes líneas :

```
set input-meta on
set output-meta on
set convert-meta off
```

Para que los cambios surgan efecto debe cerrar y abrir una nueva sesión del programa. Alternativamente cada usuario podría modificar el archivo `~/.inputrc` (ver Configuración de una sesión).

Para que otros programas tipo texto que no usan `readline` reciban y presenten caracteres de español, debe configurar un locale apropiado y asegurarse de que la terminal permita caracteres de 8 bits.

Programas como vi, emplean funciones de la libreria de C para presentar y recibir información de acuerdo al locale. La librería de C emplea algunas variables de ambiente (LANG, LANGUAGE, LC_CTYPE, LC_NUMERIC, LC_TIME, LC_COLLATE, LC_MONETARY, LC_MESSAGES y LC_ALL) y la información del directorio `/usr/share/locale` cuando "responde" a programas que emplean funciones de locale. En particular los caracteres que el locale identifica como imprimibles se configuran con la variable de ambiente LC_CTYPE. El valor que está variable (al igual que las otras mencionadas) debería tener es "es_CO" que corresponde a español de Colombia.

Sin embargo Debian 2.2 no soporta este locale por defecto (la próxima versión Debian 3.0 si lo soporta), para suplir la carencia instale el paquete `i18ndata` y después ejecute:

```
cd /usr/share/i18n/locales
localedef -i es_CO -f ISO-8859-1 es_CO
```

Con lo cual se crearán los directorios y archivos apropiados en `/usr/share/locale/es_CO` (ISO-8859-1 o Latin1 se refieren a una codificación de caracteres apropiada para idiomas de europa occidental —incluyendo español. ISO-8859-1 es la referencia del documento ISO donde tal codificación se estandariza).

Puede comprobar que los cambios surgen efecto poniendo la variable LC_CTYPE en es_CO (i.e. **export LC_CTYPE=es_CO**), entrando por ejemplo a vi e ingresando un caracter tildado o propio del español (sólo cuando el locale y la terminal están bien configurados verá la tilde, cuando no verá códigos que corresponden a los caracteres tildados). Ponga el resto de variables LC_* también en es_CO con **export LC_ALL=es_CO**.

Para comprobar que los mensajes se presentan en español ponga las variables LANG y LANGUAGE en `es_CO` y entre a algun programa que emplee mensajes traducidos a español como dselect.

Cada usuario podrá entonces configurar de manera permanente el *locale* de Colombia (ver Configuración de una sesión) o usted puede establecer las variables de localización en `/etc/security/pam_env.conf` (ver Administración de usuarios).

Además del locale, recomendamos instalar el paquete `task-spanish`.

5.6.1.3. Tiempo y zona geográfica

En sistemas Linux hay dos horas: hora del reloj del hardware y hora del sistema. La primera se basa en un reloj físico que está en la placa base del computador y que opera incluso cuando este está apagado, la segunda es mantenida por el kernel [84] como tiempo UTC [85]. Los programas que requieren información sobre la hora y fecha, emplean funciones de la biblioteca de C; esta biblioteca, a su vez emplea la hora del sistema y la información sobre la zona geográfica en la que está el computador para calcular e informar la hora local.

Durante el arranque del sistema la hora del sistema es inicializada empleando el reloj del hardware con el programa hwclock, después el reloj del hardware no vuelve a ser consultada (excepto si se configuran ajustes de tiempo o si se emplea hwclock). En el reloj del hardware el administrador puede decidir dejar bien la hora local o bien la hora en el meridiano 0, (puede modificar esa hora con hwclock o seguramente desde el programa de configuración de la BIOS de su computador). La decisión que tome debe indicarla al sistema para que haga el ajuste apropiado durante el arranque, esto lo hace asignado "yes" o "no" a la variable UTC del archivo `/etc/defaults/rcS` –si emplea otros sistemas operativos que dependan sólo de la hora del sistema mejor mantenga la hora local en el reloj del hardware.

El formato de los archivos de `/usr/share/zoneinfo` está especificado en el archivo `/etc/localtime`, el cual normalmente debe ser un enlace a uno de los archivos del directorio `/usr/share/zoneinfo`, los programas **tzselect** y **tzconfig** le presentan un menú que le permite establecer este enlace.

5.6.1.4. vim

Vim es una versión mejorada de vi. Instalelo con el paquete vim. Para que todos los usuarios empleen este editor por defecto al ejecutar el comando **vi**, puede crear un alias en `/etc/profile`:

```
alias vi="vim"
```

5.6.1.5. LaTeX

Para emplear TeX y LaTeX (versection 3.1.1.2 LaTeX), instale la distribución Tetex, puede emplear el paquete (tarea) `task-tex` que instalará Tetex y otras aplicaciones útiles (por ejemplo gv ver Impresión y formatos para impresión).

Para convertir de LaTeX a HTML recomendamos las herramienta HeVea y latex2html. Las fuentes de la primera escritas en Ocaml están sponibles en: http://para.inria.fr/~maranget/hevea/

5.6.1.6. Gnome

Si durante la instalación no eligió el escritorio Gnome, puede hacerlo posteriormente instalando task-gnome.

5.6.1.7. Ocaml

Puede instalar los paquetes ocaml, ocamltk y ocaml-doc, aunque la versión disponible en estos paquetes es la 2.04. Si desea la versión más reciente puede compilarla directamente de las fuentes. El sitio de

84. La hora del sistema corresponde al tiempo en segundos transcurrido desde Enero 1 de 1970 a las 00:00 medido sobre el meridiano 0.
85. UTC - *Universal Time Coordination*, es la hora medida en el meridiano 0, también se llama GMT Greenwich Meridian Time.

distribución es http://caml.inria.fr/ocaml/distrib.html. Antes de compilar, si desea soporte para Labltk (Tcl/Tk desde Ocaml), instale los paquetes tcl8.2-dev, tk8.2-dev.

5.6.1.8. DocBook

Hay dos tipos de DocBook: XML y SGML, de cada uno hay diversas versiones. En el CD de Potato está DocBook SGML v3.1 (paquete docbook) y la versión 3.1.3 de DocBook XML (paquete docbook-xml).

Para realizar la conversión de DocBook SGML a PostScript, HTML y otros formatos puede emplear:

- Las hojas de estilo DSSL (paquete docbook-stylesheets) y el procesador jade (paquete jade). Por ejemplo para genera un HTML a partir del archivo `ej.db`:

  ```
  dsl=/usr/lib/sgml/stylesheet/dsssl/dcbook/nwalsh/html/docbook.dsl
  jade -t sgml -ihtml -d$dsl ej.db
  ```

- Las herramientas del paquete sgmltools-2 (ver DocBook). Con estas puede generar un HTML con:

  ```
  sgmltools -b html ej.db
  ```

  Si requiere una versión más reciente de DocBook SGML (4.1) puede emplear el repositorio no oficial (ver Paquetes en Debian):

  ```
  deb http://dtype.org/debian/potato ./
  ```

  Instale de allí los paquetes docbook y sgmltools-lite (el segundo remplaza a sgmltools-2). Podrá generar a partir de DocBook en otros formatos tal como con sgmltools.

  Si desea la versión más reciente de DocBook (SGML o XML) puede consultar http://www.oasis-open.org/docbook/. La versión más reciente de las hojas de estilo DSSSL, así como hojas de estilo XSL están disponibles en: http://docbook.sourceforge.net/projects/index.html que pueden emplearse con un procesaro XSL como **xsltproc** disponible en el paquete `libxslt` o la versión más reciente en: http://www.xmlsoft.org/XSLT/.

5.6.1.9. Mozilla

La versión incluida en los paquetes oficiales no es tan reciente. Puede instalar una reciente agregando la línea:

```
deb http://dtype.org/debian/potato ./
```

a '/etc/apt/sources.list' y ejecutando

```
apt-get update
apt-get install mozilla
```

5.6.1.10. Diccionarios

Para que puedan funcionar los diccionarios (ver Edición de texto), instale los paquetes dict (cliente) y dictd (servidor). Los diccionarios que hay disponibles (todos en inglés) son:

- dict-web1913 Diccionario Webster de 1913 (de dominio público)
- dict-wn Diccionario WordNet 1.6
- dict-jargon Diccionario con jerga de hackers.
- dict-easton Diccionario de términos bíblicos de 1897
- dict-elements Datos sobre los Elementos
- dict-gazetteer U.S. Gazeteer
- dict-hitchcock Diccionario de nombres bíblicos de Hitchcock

5.6.2. Lecturas recomendadas: Localización y configuración de algunos programas

- Puede consultar más sobre el locale en la sección "Locales" del manual info de la libreria de C (**info libc**) o en línea en http://www.gnu.org/manual/glibc-2.2.3/html_chapter/libc_7.html#SEC108

  Puede consultar la especificación que define como escribir su propio locale: http://www.opengroup.org/onlinepubs/7908799/xbd/locale.html

- Puede ver más sobre el tiempo y Linux en el *Mini-HOWTO sobre el reloj* http://www.linuxdoc.org/HOWTO/mini/Clock.html

5.6.3. Ejercicios: Localización y configuración de algunos programas

1. En la sugerencia hecha en esta guía para modificar un mapa del teclado US, se presupone que ya pueden generarse los caracteres especiales de español, sin emplear digrafos de vim o de otro editor de texto. Cómo puede generarlos?

Capítulo 6. Administración de una red con Linux

Público y logros

Administrador de Red del colegio, quien solucionará problemas con una red LAN de la institución o la instalará de acuerdo a la plataforma de referencia.

- Conoce algunas de las posibilidades que puede aprovechar al tener Linux en una red LAN.
- Entiende conceptos básicos de redes TCP/IP así como del esquema cliente-servidor.
- Puede planear, instalar el hardware y configurar una red Ethernet de PCs con Linux, de acuerdo a la plataforma de referencia S-Helio 1.1 (ver Plataforma de referencia).
- Conoce y puede instalar y configurar clientes y servidores para: ftp, telnet, ssh, finger, talk, smtp, http, NIS y NFS de acuerdo a la plataforma de referencia.

6.1. Redes, protocolos e Internet

Indicadores de Logros

- Distingue las capas de protocolos de Internet (físico, enlace, internet, transporte, aplicación) y sus generalidades.

6.1.1. Lectura: Redes, protocolos e Internet

Las redes de computadores constan de medios de transmisión (e.g cables), dispositivos (e.g tarjetas de red) y programas (e.g módulos para TCP/IP del kernel) que permiten transmitir información entre computadores. Pueden clasificarse de acuerdo al tamaño (WAN de cobertura amplia como un país, MAN de cobertura metropolitana o LAN para edificios o salones) o de acuerdo a la estructura de sus protocolos (e.g el módelo OSI que consta de 7 capas de protocolos: física, enlace, red, transporte, sesión, presentación y aplicación), o de acuerdo a la distribución física de los medios de transmisión y dispositivos (e.g estrella, bus).

LAN es acrónimo de *Local Area Network*, con este término se hace referencia a redes pequeñas (comúnmente menos de 100 computadores) cuyos computadores están generalmente en un mismo espacio físico. Los principales usos de una red LAN son:

- Compartir información distribuida en los discos duros o medios de almacenamiento de cada computador.
- Emplear recursos de un computador desde otro, o recursos conectados en red (por ejemplo una impresora).

- Compartir una conexión a Internet.
- Conformar una Intranet con servicios como correo, *web*, ftp, *news*.

Un protocolo [1] es una serie de reglas que indican como debe ocurrir una comunicación entre dos computadores de una red; las reglas de un protocolo son seguidas por programas que se ejecutan en los computadores interconectados. Su computador puede tener procesos activos que esperan conexiones de otros computadores o procesos para iniciar protocolos (por ejemplo con **ps -ax** es posible que vea: `portmap`, `inetd`, `lpd`, `xfs`, `postgres`). A estos programas, que esperan conexiónes, se les llama *daemons* o servidores. Usted puede iniciar un protocolo con otro computador empleando el programa cliente adecuado, por ejemplo el cliente **ping** con:

```
ping 127.0.0.1
```

iniciará el porotocolo de ping (`ICMP`, `ECHO_REQUEST`) con el computador con dirección IP 127.0.0.1 (su propio computador), un programa de su computador estará pendiente de conexiones de este tipo y seguira el protocolo[2].

Internet es una red mundial de redes que están interconectadas vía satelite, por cables submarinos, fibra óptica y muchos otros medios de transmisión financiados por estados y organizaciones. Las heterogeneidad de las redes que interconecta (e.g una LAN de un colegio con Linux, una LAN de una empresa con Windows, una WAN de un banco con X.25) es superada con los protocolos TCP/IP. Linux soporta bien estos protocolos y la red que sugerimos se basa en este protocolo. Por esto y para facilitar la conexión de una LAN a Internet, en esta sección se presenta una introducción a los aspectos de redes de Internet; los temas introducidos serán tratados en detalle en otras secciones aplicandolos especialmente al caso de redes Ethernet (ver Dispositivos para interconexión y Linux, Direcciones, enrutamiento y transporte y Protocolos de soporte y de usuario).

6.1.1.1. Internet

Hay muchas personas que aportan su trabajo (muchos de forma voluntaria) para mantener Internet en funcionamiento, para solucionar problemas que ocurren y para planear su futuro. Quienes dirigen el rumbo de Internet están organizados en comités encargados de diversas actividades:

IETF Internet Engineering Task Force

: Compuesta enteramente de voluntarios autoorganizados que colaboran en la ingeniería requerida para la evolución de Internet, es decir identifican problemas y proponen soluciones. Quien lo desee puede participar, asisitir a las reuniones, ayudar a dar forma o proponer estándares. La IETF recibe borradores (*Internet Draft*) de autores que renuncian al control del protocolo que proponen. Tras refinamiento estos borradores pueden llegar a ser RFC (*Request for Comment*) de uno de estos tipos: estándar propuesto, protocolo experimental, documento informativo o estándar histórico. Un estándar propuesto puede después convertirse en borrador de estándar y en casos muy contados en estándar de Internet.

1. Para ampliar el significado de los términos técnicos introducidos en estas guías (como *protocol* o *daemon*), se sugiere consultar el diccionario FOLDOC en su sistema (ver Edición de texto) o en Internet http://foldoc.doc.ic.ac.uk/foldoc
2. El programa que sigue este protocolo hace parte de la implementación de IPv4 en el kernel (archivo `net/ipv4/icmp.c`).

ISOC Internet Society

Organización sin ánimo de lucro, da soporte legal y financiero a otros grupos.

IESG Internet Engineering Steering Group

Ratifica o corrige estándares propuestos por la IETF.

IAB Internet Architecture Board

Planeación a largo plazo y coordinación entre las diversas áreas.

IANA Internet Assigned Numbers Authority

Mantienen registros de diversos nombres y números asignados a organizaciones. Es financiado por ICANN (*Internet Corporation for Assigned Names and Numbers*).

La interconexión de las redes conectadas a Internet y su mantenimiento se basa en el protocolo IP, que asigna a cada red y a cada computador un número único (dirección IP) que permite identificarlo y enviarle información.

La asignación de direcciones IP[3] es manejada por IANA, que a su vez la ha delegado a registros en diversas regiones. Las regiones de América y Africa debajo del Sahara son manejadas por el registro ARIN (*American Registry for Internet Numbers*), que recibe solicitudes para asignar bloques a provedores de servicio a Internet (ISP), estos proveedores deben contar con la infraestructura para conectar sus computadores a Internet (e.g con cables submarinos, vía satélite) y a su vez ofrecen bloques de direcciones y conexión a otros proveedores o a usuarios.

Dada que la comunicación de dos computadores es un proceso complejo (aún más dada la variedad de redes conectadas a Internet), además de una basta infraestructura física, Internet se basa en varios protocolos que siguen todos los computadores y enrutadores conectados. Algunos de estos protocolos dependen de otros, dando lugar a varias capas de protocolos: aplicación, transporte, internet, enlace[4]. El siguiente diagrama presenta algunos protocolos en las diversas capas de una red TCP/IP sobre una red física Ethernet o sobre una conexión con modem:

```
Capa de aplicación:
     Protocolos de usuario             FTP Telnet ssh   http   SMTP ...
                                        |     |     |      |    |
     Protocolos de soporte    DNS       |     |     |      |    |
                             |   \      |     |     |      |    |
Capa de transporte           UDP   TCP -----------------
                              \   /
Capa de internet (red):      IP -  ICMP
                             ___|___
                            /       \
Capa de enlace:           ARP        PPP
                           |          |
Capa física:        Ethernet     Modem
                        |              |
                     Par trenzado    Línea telef.
```

3. Puede verse más sobre asignación de IPs en el RFC 2050 y sobre asignación de direcciones in redes IP privadas en RFC 1918.
4. El RFC 1122 presenta las capas de una red TCP/IP

6.1.1.2. Capas de una red TCP/IP sobre algunos medios físicos

6.1.1.2.1. Capa física

Se refiere a las conexiones eléctricas y mecánicas de la red. Ejemplos de protocolos a este nivel son: Ethernet, Inalámbrico IEEE 802.11, Modem y fibra óptica. La información por transmitir se codifica en últimas como una señal electríca que debe transmitirse por cables o como una señal electromagnética (luz, ondas). En medios de transmisión compartidos por más computadores (Ethernet, fibra óptica, aire), el protocolo de este nivel debe tener en cuenta: codificación/decodificación de información del bus al medio de transmisión y arbitrar en caso de colisión de datos. Este tipo de protocolos es implementado por hardware, comúnmente por una tarjeta o dispositivo dedicado que se debe conectar al bus de cada computador. Las tarjetas de los computadores que se comunican se conectan empleando el medio de transmisión. Linux, a nivel físico soporta dispositivos ethernet, fddi, token ring, ax25 y cuenta con controladores para diversos modems. En el caso particular de Ethernet, existen en el momento de este escrito estándares para 10Mbps, 100Mbps, 1Gbps y 10Gbps, la versión de 10Mbps puede emplearse en estrella (i.e. todos los computadores conectados por cables de pares trenzados a un concentrador) o en bus (i.e. un sólo cable coaxial al que se conectan las tarjetas de todos los computadores), puede configurarse como *Half-duplex* o como *Full-Duplex*. Describiremos el uso del esquema más popular y económico en este momento: 10Mbps, Half-duplex con cable de pares trenzados [5]. Cada tarjeta Ethernet tiene una dirección única (llamada dirección MAC), que en una transmisión permite indicar tarjeta fuente y tarjeta destino (la dirección MAC de ambos es transmitida también).

6.1.1.2.2. Capa de enlace

Los protocolos de esta capa permiten interconectar la capa de internet (IP) con la red que se use. En el caso de una red Ethernet se trata del protocolo ARP[6] que en Linux está implementado como parte de IP, y que se encarga de traducir direcciones IP a direcciones MAC [7].

Cada vez que un computador de una LAN identifica una dirección IP de otro computador conectado a la LAN y su correspondiente dirección ARP, almacena la información en una tabla (se borra automáticamente después de 15 minutos o manualmente si se instala el paquete `iproute`con algo como **ip neigh delete 192.168.1.32 dev eth0**). Esta tabla puede consultarse en `/proc/net/arp` o con **ip neigh show**.

Para monitorear una red Ethernet y detectar nuevos computadores que se conecten, puede emplearse el programa arpwatch (paquete `arpwatch`) que cada vez que detecta cambios envía un correo a la cuenta root.

En el caso de una conexión por modem el protocolo es PPP[8], que se encarga de establecer, terminar y verificar la conexión. Durante la conexión autentica el computador que se conecta ante el servidor (bien con el protocolo PAP o con CHAP) y durante la operación preparar los paquetes enviados por otros protocolos (como IP) para transmitirlos por modem.

5. El estándar que define este esquema es IEEE 802.3, que se basa en el uso de un sólo medio de transmisión compartido por todos los dispositivos en el que sólo trasmite un sólo dispositivo durante un tiempo para evitar colisiones, se elige el siguiente dispositivo por transmitir con el algoritmo CSMA/CD –cuando un dispositivo detecta que la línea está libre transmite parte de la información si detecta colisión da oportunidad de transmisión a otros un intervalo aleatorio de tiempo.
6. ARP se define en el RFC 826
7. Un computador envía un mensaje a todos los demás de la red (*broadcast* que es posible en Ethernet), solicitando la dirección MAC de una dirección IP, el dispositivo con esa MAC responde enviando su dirección MAC al dispositivo que hizo la solicitud.
8. PPP se describe en el RFC 1661.

6.1.1.2.3. Capa de internet

En esta capa la información es fragmentada y envíada empleando el protocolo de la capa física. Los protocolos de esta capa deben tener en cuenta la retransmisión de la información en caso de error al enviar y el verificar información recibida. En esta capa está el *Internet Protocol* (IP) que es un protocolo diseñado para Internet y del cual hay dos versiones: IPv4 e IPv6. Describiremos la versión 4 [9] (la versión 6 fue diseñada para soportar más computadores conectados a Internet aunque en el momento de este escrito no es ampliamente usada [10]).

Este protocolo permite la transmisión de paquetes (llamados datagramas) en redes donde cada computador interconectado se identifica con una dirección única. Tales direcciones están diseñadas para interconectar varias redes, identificar los computadores que pertenecen a una red (empleando una máscara de red que indica que parte de la dirección del computador corresponde a la dirección de la red) y facilitar el enrutamiento. Si el medio de transmisión lo requiere el protocolo IP se encarga de la división de los datagramas en paquetes más pequeños para su tranmisión y de la posterior reagrupación (fragmentación), el tamaño máximo que un paquete puede tener para un protocolo de nivel físico se llama MTU (*Maximal Transfer Unit*), en el caso de Ethernet es 1500 bytes.

Cada datagrama por transmitir es pasado a la capa de IP por otro protocolo de una capa superior (e.g TCP) junto con dirección destino, IP mantiene una tabla de enrutamiento que asocia direcciones destino con compuertas (computadores intermediarios en inglés *gateways*). Así que envía el datagrama empleando el nível físico a la dirección de la compuerta que mantenga en su tabla de enrutamiento. La tabla de enrutamiento puede ser modificada manualmente (con el programa **route**) o puede ser modificada automáticamente cuando una compuerta envía un mensaje indicando la dirección de otra compuerta más apropiada para llegar a un dirección. Hay siempre una compuerta por defecto a la que se envían paquetes que IP no sepa como enrutar.

Este protocolo no es fiable, no asegura que un paquete llegue a su destino y no realiza retransmisiones. Para informar algunas situaciones anomalas emplea el protocolo ICMP [11].

Los kernels 2.2 y 2.4 cuentan con buenas implementaciones de IPv4 con multicasting nivel 2 (RFC 1112), enmascaramiento [12], con posibilidad de enrutar paquetes y de filtrar paquetes. Entre las posibilidades no estándar que ofrece están: balanceo de carga si hay múltiples provedores de acceso a Internet, políticas avanzadas de enrutamiento (por ejemplo con base en usuarios), túneles que permiten interconectar dos redes con la misma dirección de red pero con una red entre ellas, IPv6, y otras que se describen en: http://www.tldp.org/HOWTO/Adv-Routing-HOWTO/lartc.iproute2.explore.html

6.1.1.2.4. Capa de enlace

Los protocolos de esta capa asegura una conexión continua y posiblemente libre de errores entre emisor y receptor. Un protocolo de esta capa debe tener en cuenta mantener los paquetes en orden y asegurar que están completos. Esto se requiere porque el nivel de red puede enrutar diversos paquetes de forma diferente. En esta capa hay dos protocolos: TCP[13] y UDP[14]. El primero es fiable, asegura que la transmisión enviada sea recibida, UDP por su parte no busca confirmar que la información llegue a su

9. El protocolo IPv4 está descrito en el RFC 791, aunque puede verse una descripción en conjunto con otros protocolos en el RFC 1122
10. El sitio de desarrollo de IPv6 para Linux es http://www.linux-ipv6.org
11. ICMP Internet Control Message Protocol, se describe en el RFC792, permite enviar mensajes de error (e.g dirección inalcanzable, tiempo excedido) y algunos mensajes para hacer consultas (e.g eco, estampilla de tiempo).
12. Enmascaramiento del inglés *masquerading*
13. TCP se describe en el RFC 793 y se complementa y corrige en el RFC 1122
14. UDP está descrito en el RFC 768 y corregido en el RFC 1122

destino (aunque permite hacer broadcast y multicast mientras TCP no). Ambos protocolos emplean puertos para permitir más de una conexión con uno o más computadores, un puerto se identifica con un número entre 0 y 65536, los primeros 1024 números sólo pueden ser usados por servidores iniciados desde la cuenta root, los demás pueden ser usados por todos los usuarios.

Puede examinar información sobre conexiones a puertos TCP en `/proc/net/tcp` y a puertos UDP en `/proc/net/udp`.

6.1.1.2.5. Capa de aplicación

Consta de protocolos empleados por aplicaciones con propósitos específicos. Emplean los protocolos de la capa de transporte para establecer comunicación. Los protocolos de esta capa pueden dividirse a su vez en protocolos de usuario y protocolos de soporte. Los primeros son para programas empleados directamente por usuarios y los segundos son empleados por el sistema operativo o por otros protocolos.

6.1.1.2.5.1. Protocolos de usuario

Telnet

: Telnet está definido en el RFC 854 y complementado en el RFC 1123 Permite la operación remota de otro computador (ver telnet y ssh y Configuración de servicios básicos), además puede usarse para interactuar con otros protocolos (por ejemplo puede interactuar con el protocolo de correo SMTP de su propio computador —puerto 25— con **telnet localhost 25**)

(*File Transfer Protocol*) FTP

: FTP está definido en el RFC 959 y complementado en el RFC 1123, permite transmisión de archivos (ver FTP y Servicio FTP)

TFPT

: Está definido en el RFC 1350 (*Trivial File Transfer Protocol*), es análogo a FTP pues permite transmitir archivos, aunque es mucho más simple, es apropiado para transmitir el sistema operativo a computadores sin disco duro que arrancan por red

SMTP

: SMTP se definió en el RFC 822 y se complementó en el RFC 1123 (*Simple Mail Transfer Protocol*) que especifica como se realiza la transmisión de correo electrónico (ver Correo electrónico y Servicio de correo).

6.1.1.2.5.2. Protocolos de soporte

DNS

: Definido en los RFC 1034 y 1035 y complementado en el RFC 1123. Es empleado para dar nombres descriptivos a direcciones IP (e.g structio.sourceforge.net es el nombre DNS de la dirección 66.35.250.209)

BOOTP

Empleado por algunas estaciones sin disco para solicitar su dirección IP

SMMP

Empleado para monitorear uso de una red.

6.1.2. Lecturas recomendadas: Redes, protocolos e Internet

- Puede consultarse más sobre el IETF y otras organizaciones que mantienen Internet en "*The Tao of IETF: A Novice"s Guide to the Internet Engineering Task Force*" http://www.ietf.org/tao.html (http://www.ietf.org/tao.html). También puede consultar un escrito muy interesante sobre dicha organización en http://www.oreilly.com/catalog/opensources/book/ietf.html
- Puede consultar los RFC en http://www.rfc-editor.org (http://www.rfc-editor.org). También puede consultar versiones en castellano en http://lucas.hispalinux.es/htmls/estandares.html
- *Networking-Overview-HOWTO*. Networking Protocols. http://www.linuxdoc.org/HOWTO/Networking-Overview-HOWTO-3.html
- *Networking-Overview-HOWTO*. File Sharing and Printing. http://www.linuxdoc.org/HOWTO/Networking-Overview-HOWTO-5.html
- *Networking-Overview-HOWTO*. Internet/Intranet. http://www.linuxdoc.org/HOWTO/Networking-Overview-HOWTO-6.html
- Chris Campbell, *Introduction to Networking, Part 1: Physical Media* http://linux.com/sysadmin/newsitem.phtml?sid=1&aid=11135

6.1.3. Ejercicios: Redes, protocolos e Internet

1. Discuta en clase las posibilidades que ofrece Linux en una red LAN y estrategias para aprovecharlas en su institución.

2. Cotice con uno o más proveedores de acceso a Internet posibilidades y costos para dar conexión a Internet esporádica o permanente a su colegio.

3. Consulte la posibilidad de emplear fibra óptica como medio de transmisión con Linux.

6.2. Dispositivos para interconexión y Linux

Indicadores de Logro:

- Puede configurar y comprar dispositivos para instalar una red Ethernet como la de la plataforma de referencia.

- Puede configurar el *loopback* de un computador.
- Puede instalar y configurar una tarjeta de red Ethernet junto con una interfaz de red.
- Puede instalar cableado de una red Ethernet como la especificada en la plataforma de referencia.

6.2.1. Lectura: Dispositivos para interconexión y Linux

De acuerdo al uso hay diversos dispositivos que permiten conectar computadores. Para el caso de redes pequeñas algunos ejemplos son:

Modems

Permiten conectar dos computadores empleando una línea telefónica.

Cable serial

Permite conectar dos computadores por sus puertos seriales.

Cable Base10T

Consta de 8 cables internos trenzados en parejas (azul con azul/blanco, verde con verde/blanco, naranja con naranja/blanco, cafe con cafe/blanco), en sus extremos tiene conectores RJ45 [15]. Permite conectar un computador a un concentrador o dos computadores directamente haciendo un cruce de los cables internos (a un cable con tal cruce se le dice cruzado –*cross-over*– o invertido –*reverse*)[16] .

Concentrador [17]

Por medio de clabes Base10T (no cruzados) permite conectar varios computadores que tengan tarjetas Ethernet de 10MB, cada uno en un puerto del concentrador. Normalmente los concentradores son repetidores que retransmiten por todos sus puertos la señal que llega por uno de ellos y detectan colisiones. Hay modelos por ejemplo de 4, 8, 10 y 16 puertos, algunos tienen un puerto más para conectar a otro concentrador.

Para operar dispositivos de red conectados a un computador con Linux, es necesario emplear el controlador apropiado y asociar el dispositivo a una interfaz de red, que a su vez tendrá una dirección IP. Así por ejemplo a un modem se le puede asociar la interfaz `ppp0` (o con otro número en lugar de 0) y a una tarjeta Ethernet se le puede asociar la interfaz `eth0` (o con otro número en lugar de 0). Hay además una interfaz que siempre debe estar presente: `lo` (loopback) que representa una conexión con el mismo computador. Los nombres de las interfaces activas en un sistema Linux pueden consultarse con: **/sbin/ifconfig** o con **netstat -i**.

En esta sección se presentan detalles de algunos dispositivos para interconexión y su configuración. Hemos procurado dar especial enfasis a los requeridos en una red local Ethernet.

15. Los conectores RJ45 son similares a los conectores de teléfono (RJ11), sólo que RJ45 tiene 8 pines mientras RJ11 tiene 4 o 6
16. Para Ethernet sólo se requieren los pines 1,2,3 y 6 del conector RJ45, en un cable cruzado deben unirse 1 con 3 (preferiblemente con cable verde/blanco), 2 con 6 (cable verde), 3 con 1 (cable rojo/blanco), 6 con 2 (cable rojo).
17. Concentrado del inglés *hub*. De acuerdo al RFC 1516 se define en la sección 9 "Repeater Unit for 10 Mb/s Baseband Networks" del estándar IEEE 802.3/ISO 8802-3 CSMA/CD

6.2.1.1. Modem

Un modem permite modular información digital para transmitirla por una línea telefónica. Infortunadamente la tecnología de modems está en gran parte patentada y muchas especificaciones son secretas, así que hay muchos modems que *NO* son soportados en Linux. Por esto, el paso más difícil es lograr que su modem sea reconocido y manejado por Linux.

No tendrá problemas con modems que puedan configurarse con jumpers y que no sean especifícos para Windows (*winmodems*), tal es el caso de modems externos que se conectan a los puertos seriales o modems que se conectan directamente al bus, pero que sean genéricos. Mientras que los primeros emplean la configuración estándar de los puertos seriales, los segundos normalmente pueden configurarse con jumpers o por software, para emplear una dirección base y un puerto libres.

Aún si tiene un winmodem es posible que exista un controlador para Linux, puede consultar una base de datos en: http://www.idir.net/~gromitkc/winmodem.html#drivers o puede buscar en el sitio web del fabricante.

Para emplear el modem desde Linux, debe haber un archivo para dispositivos en el directorio `dev`, que permita usarlo. Comunmente un modem quedará asociado a uno de los dispositivo seriales: `/dev/ttyS0`, `/dev/ttyS1`, `/dev/ttyS2` o `/dev/ttyS3`. Tal dispositivo serial empleará puertos y direcciones base, que tienen valores por defecto o que en caso de requerirse pueden modificarse con **setserial** (ver Puertos y dispositivos seriales).

Una vez haya configurado el hardware de forma que esté asociado a un dispositivo serial, puede probar que este funcionando. Una forma fácil de hacerlo es enviando la cadena `ATA` al dispositivo del modem Por ejemplo si configuró su modem como primer puerto serial, intente:

```
echo "ATA" > /dev/ttyS0
```

al hacerlo debe escuchar la línea telefónica en el parlante del computador o en el del modem (claro está debe tener conectada la línea telefónica al modem) [18]

También puede emplear un programa como minicom para enviar comandos a su modem y ver las respuesta que el mismo genera (por ejemplo al enviar la cadena `ATZ` debe responder `OK`).

Cuando haya determinado que su modem está operando correctamente en un dispositivo puede emplear minicom para conectarse a un BBS o al computador de algún amigo. Para conectarse a Internet debe configurar el protocolo PPP y dar información sobre su provedor de acceso a Internet. En Debian puede hacer estos pasos con los programas **pppconfig** o con **wvdial**. Con **pppconfig** podrá configurar más de un provedor y asignarle a cada uno un nombre, tal nombre puede usarlo posteriormente para iniciar una conexión con el programa **pon**.

Hay también algunos Modems USB soportados (que cumplen ACM –*Abstract Control Model*): 3Com OfficeConnect 56k, 3Com Voice FaxModem Pro, 3Com Sportster MultiTech MultiModem 56k, Zoom 2986L FaxModem, Compaq 56k FaxModem, ELSA Microlink 56k. Se maneja con un dispositivo `/dev/usb/acm/0` (166,0) que puede emplearse como otros tipos de modems.

18. El comando **ATA** es entendido por el modem como "contestar una llamada", por eso cuando lo da sin que haya una llamada entrante escucha la línea.

6.2.1.2. Tarjeta Ethernet

Los integrados (y eventualmente tarjetas) Ethernet de 10MB o 100MB soportadas por el kernel 2.2.19 son [19]:

- Tarjetas ISA: AMD Lance y PCnet AT1500, NE2100 (lance), SMC Ultra (smc-ultra), 3Com 3c501 (3c501), 3c503 (3c503), 3c505 (3c505), 3c507 (3c507), (3c523), 3c527 (3c527), 3c509 3c579 (3c509), 3c515 ISA (3c515), AT170/1720 (at1700), Cabletron E21xx(e2100), WD80x3 (wd), DEPCA/DE10x,DE200,DE201,DE202, DE422 (depca),Etherworks DE203,DE204,DE205 (ewrk3), EtherExpress 16 (eexpress), EtherExpressPro/100 (eepro), Fujitsu FMV-181/182/183/184 (fmv18x), HP PCLAN+ 27247B/27252A (hp-plus), HP PCLAN 27245/27xxx) (hp.o), HP 10/100VG PCLAN (hp100), ICL EtherTeam 16i/32 (eth16i), NE2000/NE2100 (ne) SMC9192/9194 (smc9194), NI5010 (ni5010), NI5210 (ni52), NI6510 (ni65)
- Tarjetas MCA: MCA NE2000 MCA para PS/2 (ne2), SKnet MCA (sk_mca), 3Com 3c523 (3c523) 3c529 (3c509), SMC Ultra MCA (smc-mca)
- Tarjetas EISA: EISA 3Com 3c579 (3c509), 3c592/3c597 (3c59x) Generic DECchip/Digital EtherWorks (de4x5), Racal-Interlan EISA ES3210 (es3210), SMC-Ultra 32 (smc-ultra32),
- Tarjetas VLB y PCI: PCI RealTek 8129/8139 (rtl8139 aunque para tarjetas 8139 es mejor rtl8139too) Fast Etherlink 3c590/3c595 o EhtereLink XL 3c900/3c905/3c905B/3c905C (3c59x) AMD PCnet32 (pcnet32), Adaptec Starfire/DuraLAN (starfire), Ansel Communications EISA 3200 (ac3200), Apricot Xen-II (apricot), LP486E (lp486e), CS89x0 (cs89x), DM9102 PCI Fast (dmfe), DECchip 2104/21041/21140 Tulip (tulip), SMC EtherPower DECchip 21040/21041/21140 (old_tulip), NetGear FA310TX (ng_tulip), Digi Intl. RightSwitch SE-X (dgrs), Mylex EISA LNE390A (lne390), Novell/Eagle/Microdyne NE3210 EISA (ne3210), PCI NE2000 (ne2k-pci), TI ThunderLAN (tlan), Via Rhine-1 y Rhine 2 (via-rhine), SiS 900/7016 PCI (sis900), SMC EtherPower II (epic100), tarjeta del Zenith Z-Note (znet),
- Puerto paralelo: AT-LAN-TEC/RealTek (atp), D-Link DE600 (de600), D-Link DE620 (de620)
- Tarjetas PCMCIA: Una gran variedad de tarjetas Ethernet PCMCIA (para portátiles) es soportada. Los controladores están incluidos en el paquete `pcmcia-cs`, una vez instalado puede consultarse en: `/usr/share/doc/pcmcia-cs/SUPPORTED.CARDS.gz`
- USB. Kawasaki USB-ethenet controller (kaweth), ADMtek AN986, Accton 10/100, Billingston USB-100, Corega FEter USB-TX, MELCO/BUFFALO LUA-TX, D-Link DSB-650TX/DSB-650TX-PN/DSB-650/DU-E10/DU–E100, Linksys USB100TX, USB10TX, LANEED Ethernet LD-USB/TX, SMC 202, SOHOware NUB Ethernet (pegasus).
- PLUSB Puede consultarse un listado por tarjetas y fabricantes, así como un descripción detallada del estado de cada una en *Ethernet-HOWTO*.
- Otros dispositivos de comunicaciones, Prolific PL-2302 conector USB-USB (plusb), red punto a punto con ancho de banda de 5 Mbits/s.

En el sitio del principal desarrollador de controladores para tarjetas de red encontrará programas e información útil: http://www.scyld.com/network/

19. Al lado de cada integrado o tarjeta soportada se presenta entre paréntesis el módulo del kernel que lo puede controlar.

Varios controladores compilados junto con el kernel pueden configurarse desde el cargador de arranque con el parámetro `ether=12,0x300,eth0` remplazando 12 por la IRQ, 0x300 por la dirección base y `eth0` por la interfaz. Entre la dirección base y la interfaz pueden pasarse uno o dos parámetros cuyo significado depende del controlador, el primero es usado por tarjetas WD80x3 para establecer comienzo de memoria compartida, otros lo usan para establecer nível de mensajes de depuración y el controldor `lance` lo usa para especificar canal DMA; el segundo es usado por el controlador 3c503 (0 transreceptor interno, 1 - AUI externo) y por e2100 para escoger medio de sálida.

Cómo módulos estos controladores reciben diversos parámetros los más comunes para tarjetas ISA son `irq=`*número* y `io=`*puerto base*. Una lista de parámetros de algunos módulo está en las fuentes del kernel en `Documention/networking/net-modles.txt`.

6.2.1.3. Tarjeta Ethernet e interfaz de red

Para emplear una tarjeta de red Ethernet en Linux para conectarse a una LAN debe:

1. Instalarla físicamente en el computador.
2. Lograr que el kernel pueda comunicarse con la tarjeta, empleando el controlador apropiado
3. Configurar una interfaz de red que servirá como abstracción de la tarjeta.

El primer paso no tendrá inconvenientes mientras manipule con cuidado el hardware y en el caso de tarjetas ISA si no emplea el mismos recurso para dos dispositivos (ver Hardware básico y Linux). Puede serle de útilidad la información sobre configuración con jumpers de tarjetas Ethernet disponible en: http://www.slug.org.au/NIC/

El segundo paso requiere cargar el controlador apropiado para su tarjeta, para esto puede consultar la sección anterior (ver Tarjeta Ethernet) o emplear **modconf** o examinar los controladores disponibles para tarjetas de red en `/usr/lib/<version_de_su_kernel>/modules/net`. Puede que tenga que probar con varios y con diversos parametros (no necesita reiniciar después de cada intento, sólo descargar el módulo que haya cargado). Para esto le serán de utilidad los programas para administrar módulos (ver Módulos). Si no encuentra un controlador apropiado para su tarjeta (o el chip que esta use), puede intentar con los controladores genéricos `ne` (si su tarjeta es ISA) o `ne2k-pci` (si su tarjeta es PCI).

Cuando haya configurado el controlador, podrá activar una interfaz de red. El kernel puede manejar varias interfaces simultaneamente, para ver la lista emplee el comando **/sbin/ifconfig** o **netstat -i**. Al emplearlo, debe ver al menos la interfaz más sencilla: `lo` que representa una conexión a su propio computador empleando como dirección IP: 127.0.0.1 y normalmente configurada en `/etc/hosts` con el nombre `localhost`.

Los nombres de las interfaces de red dependen del protocolo que soporte el dispositivo que empleará, para el caso de ethernet los nombres son `eth0` para la primera tarjeta, `eth1` para la segunda y así sucesivamente (un computador puede tener más de una tarjeta de red ethernet, aunque para nuestra plataforma cada computador requiere una sola).

Para activar o desactivar una interfaz de red en Debian puede emplear **ifconfig** con las opciones **up** y **down** respectivamente junto con parametros del protocolo (dirección IP del computador, mascara de red, compuerta). Otra forma de hacer esto en Debian (que además mantendrá la configuración después de reiniciar) es editando los parametros de las interfaces en el archivo `/etc/network/interfaces` y empleando los programas **/sbin/ifup** y **/sbin/ifdown** que reciben como parámetro el nombre de la interfaz por activar o desactivar, por ejemplo:

```
ifup eth0
```

Su archivo `/etc/network/interface` debe tener datos como los siguientes (aunque cambiando los valores de `address` y `gateway` de acuerdo al diseño de la red):

```
# Interfaz loopback
iface lo inet loopback

# Primera tarjeta de red
iface eth0 inet static
address 192.168.1.5
netmask 255.255.255.0
network 192.168.1.0
broadcast 192.168.1.255
gateway 192.168.1.1
```

La línea `iface eth0 inet static` indica que la interfaz `eth0` empleará el protocolo TCP/IP4 (`inet`) con una dirección IP estática [20].

Aunque comúnmente cada tarjeta de red tiene asociado una sóla interfaz de red y una sóla IP, en algunas circunstancias puede requerir más. Una segunda interfaz para eth0 se puede configurar con el nombre `eth0:0` que podrá configurar de forma análoga a la primera en `/etc/network/interface`.

6.2.1.4. Instalación de una red local Ethernet

6.2.1.4.1. Planeación de la red

De acuerdo a la plataforma de referencia S-Helio 1.1 necesitará un concentrador con suficientes puertos para todos los computadores que tenga, cable 10BaseT con conectores RJ-45 y en cada computador deberá tener una tarjeta Ethernet (de 10MB) como NE2000 o compatible con un puerto para un conector RJ-45 (ver tarjeta-ethernet) .

El espacio físico en el que esté la red será la primera restricción que debe tener en cuenta. Haga un plano de ese espacio con las distancias a escala, ubique los computadores y diseñe el recorrido de los cables al concentrador. Hay varias recomendaciones que puede tener en cuenta al diseñar el mapa:

- Es aconsejable por estética y seguridad que los cables vayan por canaletas (aunque no es indispensable).
- La longitud máxima de cada cable (para unir concentrador y computador) es de 100 m.
- Busque que los cables/canaletas vayan por las paredes del recinto y estén resguardadas (para evitar que alguien se tropiece).

En el mapa que haga también puede consignar las direcciones IP que planee usar en cada computador. Emplee direcciones asignadas para redes privadas, por ejemplo 192.168.1.1 al servidor y los clientes

20. Una alternativa difere es configurar direcciones dinámicas don el protoclo DHCP. Con este protocolo la dirección IP de cada computador es asignada por un servidor, cuando el computador entra a la red. Esto es útil para redes inalámbricas o cuando debe conectar y desconectar con frecuencia computadores de la red —supuestos que no forman parte de nuestra (ver Plataforma de referencia)

192.168.1.2, 192.168.1.3 y así sucesivamente. Cómo compuerta emplee en todos los clientes la dirección del servidor y como máscara de red emplee `255.255.255.0` (ver Direcciones, enrutamiento y transporte).

6.2.1.4.2. Adquisición de Hardware

En el mercado encontrará diversos precios y sugerencias distintas a las que damos. Por ejemplo es posible que pueda conformar una red con cable coaxial, sin necesidad de concentrador a menor precio. Ese esquema está cada vez en más desuso, su instalación es un poco más difícil (debe poner resistencias de 50ohmios a los extremos del cable) y no escalará tan bien como concentradores en cascada.

Para hacer la adquisición de Hardware tenga en cuenta:

- Los nombres de los componentes pueden variar de un almacén a otro, algunos sinónimos son:
    - tarjeta ethernet, tarjeta compatible NE2000 con conectores RJ45
    - concentrador o *hub* o *hub* repetidor
    - cable de pares trenzados, par trenzado, twisted pair, cable 10BaseT, UTP (*Unshielded Twisted Pair*).
- Use el plano de red para determinar la longitud de cada cable, recuerde que todo computador debe tener un cable que lo una con el concentrador (compre un poco más de la longitud que midió pues al intentar ensamblar los conectores podría perder algo de cable en cada intento).
- El concentrador debe tener suficientes puertos para todos los computadores (pueden ponerse varios concentradores en cascada).
- Cada computador debe tener una tarjeta de red Ethernet 10MB (e.g. NE2000) que pueda usar desde Linux (ver Tarjeta Ethernet).
- Cada cable debe tener dos conectores RJ-45. Uno para conectarlo al computador y el otro para conectarlo al concentrador. (Compre varios conectores RJ-45 adicionales pues al intentar ensamblar podría perder algunos).

6.2.1.4.3. Instalación

Una vez tenga instaladas las tarjetas de red debe conectar los cables a tales tarjetas y al concentrador. Como eventualmente usted mismo hará los cables, en esta sección damos instrucciones para que le resulte fácil el proceso. Requerirá unas pinzas especiales [21] para conectores RJ-45 y un probador (*tester*) para comprobar que fluye corriente en los cables que haga.

Ubique en el espacio para la red los computadores y los cables (verifique que las medidas de su plano hayan sido correctas).

Ponga en cada extremo de cada uno de los cables un conector RJ-45 empleando unas pinzas especiales. Como el cable 10BaseT se compone de 8 cablecitos de colores tenga en cuenta:

21. En Colombia a tal "pinza especial" se le conoce como "ponchadora".

1. Deje entre 8mm y 12 mm de los 8 cables al descubierto. [22]
2. Empareje los 8 cablecitos antes de intentar ponerlos en el conector RJ-45. Pongalos en este orden: 1 naranja/blanco, 2 naranja, 3 verde/blanco, 4 azul, 5 azul/blanco, 6 verde, 7 café/blanco, 8 café [23].
3. El conector RJ45 tiene varios canales, por cada uno de esos canales debe pasar un cablecito de color. Empuje bien los cablecitos hasta el fondo del conector RJ-45 y con las pinzas especiales baje los contactos del conector y asegure el cable.
4. Después de ensamblar el primer extremo verifique con un probador que todos los cablecitos hagan contacto. Después ensamble el segundo extremo empleando la misma secuencia de colores y después verifique que estén haciendo buen contacto con un probador.

Una vez tenga los cables verifique que la tarjeta de red de cada computador esté bien instalada (algunas tienen luces [24] que se encienden cuando transmite o recibe información por el cable), y conecte con cables todas las tarjetas al concentrador.

Verifique también que cada tarjeta de red sea reconocida por el kernel y asigne la IP que planeó para cada una en `/etc/network/interfaces` (ver Tarjeta Ethernet e interfaz de red).

Finalmente verifique la instalación trasnmitiendo paquetes de un computador a otro. Por ejemplo desde el servidor (tal vez con IP 192.168.1.1) intente conectarse a un cliente (tal vez IP 192.168.1.2) con **ping** :

```
ping 192.168.1.2
```

6.2.2. Lecturas recomendadas: Dispositivos para interconexión y Linux

- Recomendamos de forma especial las siguientes páginas del manual: **ifconfig**, **ifup**, **ifdown** y `interfaces` (que presenta la sintaxis del archivo de configuración `/etc/network/interfaces`.
- En Internet, puede consultar más sobre controladores para tarjetas Ethernet en: http://www.linuxdoc.org/LDP/nag2/x-087-2-hardware.drivers.ethernet.html
- Si desea conocer más sobre redes Ethernet puede consultar la siguiente referencia: Charles Spurgeon. *Quick Reference Guides to 10 Mbps Ethernet*. http://wwwhost.ots.utexas.edu/ethernet/descript-10quickref.html
- Puede consultar más sobre cables UTP en Wiring: Color Codes, Terms and Tools (http://www.johnscloset.net/wiring/index.html)

22. Pablo Chamorro nos indicó que "algunas pinzas tienen un tope, entonces al colocar las puntas de los cables junto al tope, el corte de la envoltura del cable siempre se realiza en el mismo punto para que ni sobre ni falte y así no hay que preocuparse por estimar el punto de corte."
23. Dado que sólo se requieren los pines 1 (recepción +), 2 (recepción -), 3 (transmisión +) y 6 (transmisión -), en realidad sólo se requiere conectar los pines de ambos conectores así: 1 - 1, 2 - 2, 3 - 3 y 6 - 6. Pero debe hacerse con una pareja de cables trenzado (para evitar deterioro de la señal).
24. Luces es traducción de *LED (Light emitting diode)*.

- Otra fuente de información sobre redes Ethernet y Linux: Guido Socher. Redes Caseras, glosario y resúmen. Linux Focus.

6.2.3. Ejercicios: Dispositivos para interconexión y Linux

1. Compruebe que el *loopback* esté operando bien empleando **ping** y revisando la configuración que recibe cada vez que inicia el sistema.

2. Desactive el loopback de su computador, examine el cambio con **ifconfig** y después reactivelo.

3. El programa **ifconfig** presenta información sobre el dispositivo y el protocolo que abstrae. Por ejemplo dirección en la red, cantidad de paquetes recibidos, máscara de red, MTU, etc. Que significa el dato MTU?

4. Instale una tarjeta de red o configúrela de forma que se cumplan todas las siguientes condiciones:

- Al arranque del sistema la tarjeta de red sea reconocida.
- La interfaz `eth0` abstraiga la tarjeta y tenga asignada la dirección IP apropiada de acuerdo a su planeación de la red, que la máscara sea 255.255.255.0 y que el nombre de la máquina sea el que planeó.
- Como compuerta debe estar la IP del servidor.

6.3. Direcciones, enrutamiento y transporte

Indicadores de Logros

- Emplea direcciones IP y configura direcciones privadas para una red local.
- Puede configurar enrutamiento en una red local.

6.3.1. Lectura: Direcciones, enrutamiento y transporte

En una red TCP/IP, cada computador tiene como identificación una dirección IP única. Esta dirección consta de 32 bits, y suele escribirse como 4 números/bytes separando unos de otros por punto (cada uno es un número entre 0 y 255), por ejemplo 66.35.250.209. Cómo TCP/IP se diseño para interconectar redes, una dirección IP consta de una parte que identifica de forma única la red y otra que identifica de forma única el computador dentro de la red. Una máscara de red determina que parte identifica la red y cuáles computadores en la red puede denotarse con el número de bits del comienzo de la dirección que identifican la red (e.g 16 si los primeros 16 bits identifican la red) o como otra dirección que al hacer la operación lógica y con la dirección IP dará la dirección de red (por ejemplo 255.255.0.0 es una máscara que indica que los primeros 16 bits de una dirección IP son la dirección de red).

Al diseñar una red debe escogerse una dirección de red junto con la máscara de acuerdo al número de computadores, algunas posibilidades son:

/8 o 255.0.0.0

16777216 computadores

/12 o 255.242.0.0

1048576 computadores

/16 o 255.255.0.0

65536 computadores

/24 o 255.255.255.0

255 computadores

Además la dirección de red que escoja debe ser única para no producir conflictos con otras redes en caso de conectarse a Internet y puede facilitar la interconexión de diversas redes y el enrutamiento al interior de una organización.

Para facilitar la adopción de redes TCP/IP en organizaciones, el RFC 1918 destinó algunas direcciones de red para usar al interior de organizaciones (no puede haber computadores en Internet con esas direcciones):

10.0.0.0 - 10.255.255.255

máscara /8

172.16.0.0 - 172.31.255.255

máscara /12

192.168.0.0 - 192.168.255.255

máscara /16

Para nuestra plataforma de referencia, recomendamos emplear direcciones entre 192.168.1.1 y 192.168.1.255 con máscara de red /24 o 255.255.255.0. O en caso de contar con más redes en el mismo colegio, la segunda con direcciones entre 192.168.2.1 y 192.168.2.255 y así sucesivamente. Además de usar direcciones privadas, se facilita el crecimiento de la infraestructura de redes y la configuración del enrutamiento entre unas y otras

Como se presentó en la descripción de las capas en redes TCP/IP (ver Redes, protocolos e Internet), el protocolo IP mantiene una tabla de enrutamiento que asocia direcciones de red con compuertas, es decir con computadores conectados a la misma red que pueden retransmitir información a la red destino.

Puede ver la tabla de enrutamiento estático en `/proc/net/route` o mejor con **route** o con **netstat -r**. Entre los campos de cada entrada de esta tabla están: red destino, mascara de la red destino, interfaz por la cual enviar/recibir paquetes con ese destino, compuerta a la cual enviar para retransmisión (o * si se envia a toda la red conectada a la interfaz), opciones entre las que puede aparecer: U enrutamiento funcionando, H destino es un computador, G destino es una compuerta, ! ruta rechazada.

Hay una compuerta por defecto (`default`) a la se envia todo paquete que no tiene un destino en la tabla de enrutamiento. La compuerta por defecto corresponde a la configurada como `gateway` de la interfaz de red en el archivo `/etc/network/interfaces`

De requerirse pueden agregarse compuertas con **route add** por ejemplo, para agregar una ruta a la red 192.168.2.0/24 usando como compuerta 192.168.1.60 que está en la misma red:

```
route add -net 192.168.2.0 netmask 255.255.255.0 gw 192.168.1.60
```

Y pueden elminarse de forma análoga con **route del**.

Para determinar problemas de enrutamiento o en general de la red, puede emplear algunas herramientas de diagnóstico por ejemplo:

traceroute

Disponible en el paquete `traceroute`, presenta las direcciones de los computadores y enrutadores que transmiten un paquete hasta llegar a su destino. Por ejemplo

```
traceroute 192.168.2.2
```

netstat -s

Éste presenta estadísticas sobre IP, ICMP y TCP

ip link list

Que presenta direcciones MAC (debe instalar primero el paquete `iproute`

tcpdump

Disponible en el paquete `tcpdump` permite analizar el tráfico de una red TCP/IP. Desde la cuenta root puede usarse este programa para examinar todo el tráfico que circule por una red.

Y para realizar configuraciones a los protocolos IP y TCP:

`/etc/hosts`

En este archivo se listan nombres asociados a la dirección IP de algunos computadores, por ejemplo:

```
127.0.0.1       localhost
192.168.1.1    servidor.micolegio.edu.co servidor
```

sysctl

Para configurar algunas variables del kernel relacionadas con redes (ver Configuración del kernel). Los cambios que pueden hacerse con **sysctl** también se pueden hacer escribiendo en los archivos del directorio `/proc/sys/net/ipv4`. Algunos son:

`net/ipv4/ip_local_port_range`

El cuál determina el rango de puertos que pueden usarse.

`net/ipv4/ip_forward`

Que habilita reenvio en puertos IP. Esto es indispensable para hacer enmascaramiento y para que el sistema actue como enrutador. También puede establecerse en el archivo `/etc/network/options` con

```
ip_forward=yes
```

ip_default_ttl

Establece el tiempo de vida por defecto de paquetes que salgan.

ipchains

Para configurar enmascaramiento o un contrafuegos[25].

6.3.1.1. Caso: Conexión de dos redes de área local

Suponga que hay dos redes LAN con direcciones privadas que desea interconectar, cada una tiene un concentrador, una tiene direcciones de la forma 192.168.1.x (i.e 192.168.1.0 con netmask 255.255.255.0 o más breve 192.168.1.0/24), y la otra 192.168.2.x.

Para que las dos redes mantengan cada una su concentrador, se requeriría un computador conectado a ambas redes que pueda retransmitir los paquetes a la red apropiada, y se tendría que configurar el enrutamiento en cada red (puede ser en cada computador o en la compuerta por defecto de cada uno), para que envie paquetes dirigidos a la otra red por el computador intermediario.

Para conectarse a ambas redes el computador intermediario debería tener dos tarjetas de red, una conectada a la primera red (digamos `eth0` con dirección 192.168.1.200), la otra a la segunda (digamos `eth1` con dirección 192.168.2.100) y debe estar configurado para retransmitir paquetes. La interfaz de red para la tarjeta conectada a la red 192.168.1.x, debe configurarse con una dirección en esa red. Por ejemplo la primera tarjeta conectada a la red 192.168.1.x puede asociarse con la interfaz `eth0` con dirección 192.168.1.200 con:

```
ifconfig eth0 192.168.1.200 netmask 255.255.255.0
```

o podría configurarse en `/etc/networks/interfaces` con:

```
iface eth0 inet static
        address 192.168.1.200
        netmask 255.255.255.0
        network 192.168.1.0
broadcast 192.168.1.255
```

25. Contrafuego del inglés *firewall*

Una vez configuradas las dos tarjetas, podría configurar el enrutamiento primero habilitando reenvio de paquetes IP:

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

o para que la configuración sea tomada durante el arranque en el archivo `/etc/network/options` (ver Inicialización del sistema):

```
ip_forward=yes
spoofprotect=yes
syncookies=no
```

después puede configurar desde la línea de comandos (o un script durante el arranque del sistema) el enrutamiento de acuerdo a las interfaces:

```
route add -net 192.168.1.0 netmask 255.255.255.0 dev eth0
route add -net 192.168.2.0 netmask 255.255.255.0 dev eth1
```

Para que otros computadores de la red 192.168.1.x se puedan comunicar con otros de 192.168.2.x y viceversa, debería configurar en los que se comunicaran o en las computertas de cada red (*EN AMBOS*) al intermediario. Por ejemplo si 192.168.1.2 se conectará con 192.168.2.2, en el primero debe usar:

```
route add -net 192.168.2.0 netmask 255.255.255.0 gw 192.168.1.200
```

y en el segundo

```
route add -net 192.168.1.0 netmask 255.255.255.0 gw 192.168.2.100
```

después de hacer esta configuración en los dos, puede probar por ejemplo desde 192.168.2.2

```
ping 192.168.1.2
```

Si tiene problemas en la conexión, puede verificar las tablas de enrutamiento (con **route** o **netstat -r**) o puede rastrear la ruta que toman los paquetes con **traceroute**, por ejemplo desde 192.168.2.2:

```
traceroute 192.168.1.2
```

6.3.1.2. Caso: Enmascaramienton

En caso de que cuente con una conexión a Internet en el servidor de su red, bien sea permanente o esporádica (por ejemplo con un modem), le será posible establecer conexiones desde cualquier computador de la red con computadores en Internet empleando enmascaramiento (note que todas las direcciones IP de la red local son privadas y no corresponden a direcciones IP reales, globlamente localizables, así que sin enmascaramiento NO podrían emplear Internet).

El computador que conecte a Internet debe tener al menos dos interfaces de red además de loopback: una tipo ethernet para comunicarse con la red interna y otra para comunicarse con Internet (por ejemplo una interfaz PPP). Cada una de las interfaces tendrá una dirección IP, la de la interfaz conectada a Internet será una dirección IP real, localizable en el mundo entero (digamos 1.2.3.4) y la otra será una dirección privada no utilizable en Internet (digamos 192.168.1.1).

Al emplear enmascaramiento las conexiones iniciadas en un computador de la red privada pasarán por el servidor, el cual cambiará la dirección del computador que inicia la conexión por la del mismo servidor y reenviará la información a Internet. De esta forma el computador que conteste en Internet responderá al servidor, el cual podrá ajustar la dirección de la información que reciba para que llegue al computador dentro de la Intranet.

Para configurar enmascaramiento [26] en Debian instale el paquete `ipmasq`, el cual instala el script **ipmasq**, lo ejecuta y prepara el proceso de inicialización para ejecutarlo siempre durante el arranque (vea Inicialización del sistema. **ipmasq** configura el computador donde se instala para hacer enmascaramiento y como contrafuegos básico (que tras la instalación permite iniciar todo tipo de conexión desde la red interna hacia la externa con un poco de seguridad [27]). El contrafuego puede restringirse agregando reglas para **ipchains** en archivos del directorio `/etc/ipmasq/rules`.

En el caso de una conexión a Internet usando **ppp**, puede crear el archivo `/etc/ipmasq/ppp` para que las reglas del contrafuegos sean configuradas cuando **pppd** inicia o termina una conexión.

6.3.2. Lecturas recomendadas: Direcciones, enrutamiento y transporte

- *Net-HOWTO*. 5.1.2. IP Addresses, an Explanation. http://linuxdoc.org/HOWTO/Net-HOWTO/c135.html#AEN138
- *Net-HOWTO*. 5.5.1. What"s in a name ? http://linuxdoc.org/HOWTO/Net-HOWTO/x398.html#AEN404
- Puede consultar más sobre enmascaramiento después de instalar el paquete `ipmasq` en el directorio `/usr/share/doc/ipmasq` y en las página **man** de `ipmasq-rule` que indica como agregar reglas al contrafuego.

6.3.3. Ejercicios: Direcciones, enrutamiento y transporte

1. Suponga que a una red como la de la plataforma de referencia, desea agreguarsele un computador cliente, ¿qué pasos deben seguirse y cómo se configura?

6.4. Protocolos de soporte y de usuario

Indicadores de Logros

26. Para emplear enmascaramiento el kernel debe ser compilado con las opciones `CONFIG_FIREWALL`, `CONFIG_IP_FIREWALL`, `CONFIG_IP_FORWARD`, y `CONFIG_IP_MASQUERADE` (configuraciones ya hechas en los kernels por defecto de Debian).
27. ipmasq evita algunos ataques que emplean *spoofing*, es decir suplantación de la IP de un computador.

- Puede configurar **talk**, **finger**, **ftp** y **telnet** en cada computador de la red.
- Puede configurar un servidor DNS y usarlo en los clientes.
- Puede distribuir utilización de discos duros usando NFS, de acuerdo a las sugerencias de la plataforma de referencia.
- Puede instalar y configurar el servicios NIS en la plataforma de referencia S-Helio 1.1.
- Configura el servicio de correo de acuerdo a la plataforma de referencia.
- Configura el servicio de ssh de acuerdo a la plataforma de referencia.
- Configura el servicio de CVS de acuerdo a la plataforma de referencia.
- Configura el servicio de ftp de acuerdo a la plataforma de referencia.
- Configura el servicio de web de acuerdo a la plataforma de referencia.
- Configura una impresora en red.
- Puede instalar y configurar el servidor Apache en la plataforma de referencia.

6.4.1. Lectura: Protocolos de soporte y de usuario

6.4.1.1. Configuración de servicios básicos

Cómo se presentó en la lectura de la sección Redes, protocolos e Internet cada servicio que su sistema ofrezca requiere un proceso *daemon* que esté pendiente de atender conexiones. Para disminuir el número de procesos que deberían estar activos esperando conexiones por cada servicio, el programa inetd que se inicia durante el arranque de su sistema (con `/etc/init.d/inetd`, ver Inicialización del sistema) espera conexiones en diversos puertos y cuando recibe una inicia el *daemon* apropiado.

inetd se configura en el archivo `/etc/inetd.conf` que puede referenciar servicios por nombres si estos están listados en `/etc/services` o en `/etc/rpc` y protocolos listados en `/etc/protocols`. Debian inlcuye versiones de estos archivos listas para servir `telnet`, `talk`, `finger`, hora local y correo (`smtp`).

Cada línea del archivo `/etc/inetd.conf` puede ser un comentario (si comienza con el caracter "#") o puede ser análoga a:

```
smtp            stream  tcp     nowait  mail    /usr/sbin/exim exim -bs
```

que indica que el servicio `smtp`, se maneja con conexiones[28] tipo `stream` del protocolo `tcp`, sin espera. Cada vez que haya una conexión para este servicio se iniciará el programa `/usr/sbin/exim` con argumentos[29] `exim -bs` desde la cuenta `mail`. Otras posibilidades para cada parte son:

servicio

: El servicio es uno de los especificados en `/etc/services` o puede ser una dirección precedida por un puerto, o uno de los servicios RPC especificados en `/etc/rpc`.

28. conexión del inglés *sockets*
29. El primer argumento corresponde al nombre del comando, es útil al emplear **tcpd**

Tipo de conexión

`stream`, `dgram` para datagramas, `raw` para conexiones puras, `rdm` para *reliable delivered messages* o `seqpacket` para paquetes secuenciados.

Protocolo

Debe ser uno de los disponibles en el archivo `/etc/protocols`, por ejemplo `tcp` o `udp`.

Espera

Puede ser `wait` o `nowait`, indica si **inetd** debe esperar a que el proceso termine para recibir nuevas conexiones en el mismo puerto o no. Esto depende del tipo de servicio, la mayoría no requiere espera (porque emplean varios hilos [30] que emplean otro puerto por cada conexión), aunque son excepciones **talkd**, **biff** y **tftp**. Después de `wait` o `nowait` puede venir separado por un punto el máximo número de conexiones que inetd debe aceptar para el servicio en 1 minuto (por defecto 256).

Usuario

Usuario a nombre del cual se iniciará el proceso.

> **Aviso**
>
> Por seguridad, es recomendable ejecutar *daemons* desde cuentas diferentes a root

Puede especificarse un grupo primario poniendolo a continuación del nombre separado por el caracter "."

Programa

Ruta del programa que atenderá la conexión. Si se trata de un servicio interno se usa `internal`. Los servicios internos son `echo` que responde con los mensajes que recibe (útil para medir y probar como se especifica en el RFC 862); `discard` que descarta todo dato que recibe (también útil para probar y medir como se describe en el RFC 863); `chargen` definido en el RFC 864 responde generando caracteres hasta que la conexión sea cerrada; `daytime` que de acuerdo al RFC 867 informa fecha y hora en formato entendible para humanos `time` hora actual en formato para máquinas de acuerdo al RFC 868.

El archivo `/etc/inetd.conf`, en una LAN como la especificada en nuestra plataforma de referencia (sin conexión a Internet) incluye:

```
telnet          stream  tcp     nowait  telnetd.telnetd /usr/sbin/tcpd  /usr/sbin/in.telnetd
talk            dgram   udp     wait    nobody.tty /usr/sbin/tcpd  /usr/sbin/in.talkd
finger          stream  tcp     nowait  nobody  /usr/sbin/tcpd /usr/sbin/in.fingerd
smtp            stream  tcp     nowait  mail    /usr/sbin/exim exim -bs
```

Después de modificar este archivo puede reiniciar **inetd**, tal como haría con otros servicios que se inician al arranque del sistema (ver Inicialización del sistema):

30. Hilos del inglés *threads*.

```
/etc/init.d/inetd restart
```

o envíando la señal SIGHUP al proceso. (ver Señales).

Note que la ejecución de **telnet**, **finger** y **talk** se hace con el programa **tcpd**. Este programa puede manejar conexiones de diversos servicios, registrar las conexiones en la bitácora y agregar un mínimo de seguridad incluyendo listas de control configurables en los archivos `hosts.allow` y `hosts.deny`. En estos archivos se configuran con patrones los servicios, computadores y/o usuarios pueden iniciar conexiones, y los que no. Cuando recibe una solicitud de conexión `tcpd` primero lee los patrones de `/etc/hosts.allow` si ninguno concuerda con la conexión intenta con los de `/etc/hosts.deny` y si ninguno concuerda permite la conexión. Para permitir acceso a estos servicios sólo a computadores en la red privada `/etc/hosts.deny` podría ser:

```
ALL: ALL
```

y `/etc/hosts.allow`:

```
ALL: 192.168.0.0/255.255.0.0
```

6.4.1.2. Servicio DNS

DNS permite asociar nombres a direcciones IP, esto es importante para facilitarnos la identificación de computadores en una red.

Como se describe en el RFC 1034, DNS puede verse desde 3 puntos de vista:

Usuario

Un usuario emplea nombres de dominios DNS al solicitar conexiones con máquinas cuyos nombres describen posiciones en un árbol de nombres. Por ejemplo `structio.sourceforge.net`.

Resolvedor de nombres

Los programas emplean un resolvedor de nombres. Este resolvedor recibe nombres con la estructura antes descrita, busca en un depósito [31] de nombres, de requerirlo trata de conectarse con uno o más servidores de nombres que puedan responder la consulta, envia la consulta y analiza la respuesta.

Servidor de nombres

Un servidor de nombres, es un programa que además de atender solicitudes de resolvedores, mantiene tablas donde asocia nombres con direcciones IP de algunas zonas del árbol de nombres. También mantiene direcciones de otros servidores de nombres que puede emplear para actualizar las tablas de las zonas que mantiene y para obtener información de otras zonas.

31. Depósito es la traducción que empleamos para *cache*

6.4.1.2.1. DNS a nivel de usuario

Los nombres son independientes de las IPs, estos conforman un árbol dividido en zonas manejadas por diversos servidores de nombres distribuidos en todo el mundo y conectados a Internet [32], la raíz de este árbol se denota con el carácter '.'. Un nombre se compone de partes separadas por el caracter '.', cada parte se compone de letras mayúsculas o minúsculas sin distinción (preferiblemente menos de 12), eventualmente números y eventualmente guiones. Las partes del nombre indican la ruta entre el nodo y la raíz, de forma análoga a la ruta de un archivo del sistema de archivos, sólo que un nombre DNS completo comienza por la parte que corresponde a la hoja y termina en la parte que corresponde a un hijo de la raíz. Por ejemplo en la dirección `structio.sourceforge.net` la parte `net` es un nodo descendiente de la raíz del árbol de nombres. A la parte descendiente de la raiz se le llama dominio de nivel superior (TLD - *Top Level Domain*) y puede ser dos letras que identifiquen uno de los 244 paises registrados de acuerdo a ISO 3166 (por ejemplo .co en el caso de Colombia) o por ejemplo una de las siguientes:

.aero

   Industria de transporte aéreo

.biz

   Miscelánea

.com

   Comercial

.coop

   Cooperativas

.edu

   Educativo

.gov

   Gubernamental de EUA

.info

   Información

.int

   Organizaciones internacionales

.mil

   Militar de EUA

.museum

   Museos

32. Hay 13 servidores principales que manejan la raíz de este árbol.

.name

Individuos

.net

Redes

.org

Organizaciones

6.4.1.2.2. DNS a nivel de resolvedor

El resolvedor de nombre es un conjunto de funciones, que los diferentes programas emplean cuando requieren determinar la IP asociado con un nombre (resolución de nombres) o el nombre asociado con una IP (resolución inversa).

Puede emplear el resolvedor de nombres desde la línea de comandos, por ejemplo con el programa **dig** (incluido en el paquete `dnsutils`). Algunos de sus usos se ejemplifican a continuación:

dig structio.sourceforge.net

Consulta todos los registros disponibles para el nombre structio.sourceforge.net usando el servidor configurado en `/etc/resolv.conf`

dig @ns1.valinux.com structio.sourceforge.net

La misma consulta anterior pero en el servidor de nombres `ns1.valinux.com` (usa el resolvedor local para determinar la IP de ese servidor)

dig structio.sourceforge.net txt

Busca sólo registros tipo T_TXT, es decir con texto arbitrario. Otras posibilidades son a (T_A) dirección; any (T_ANY) todo tipo de información; mx (T_MX) intercambiadores de correo; ns (T_NS) servidores de nombres; soa (T_SOA) zona de autoridad; hinfo (T_HINFO) información de la máquina; axfr (T_AXFR) zona de transferencia.

dig -x 192.25.206.10

Para especificar resolución inversa. También podría emplearse **dig 10.206.25.192.in-addr.arpa**

En Internet puede consultarse información sobre el registro de un dominio (incluyendo datos de quien lo registro) empleando **whois**, o información sobre el nombre de máquina y dominio del computador que usa con **hostname** (ver Comandos y programas útiles al hacer scripts).

El resolvedor de nombres incluido en Debian puede resolver nombres empleando el archivo `/etc/hosts`, servidores de nombres o NIS. Se configura en los archivos `/etc/resolv.conf`, `/etc/host.conf` y si se usa NIS también en `/etc/nsswitch.conf`. El primero (`/etc/resolv.conf`) contiene direcciones de servidores de nombres y configuración para buscar en ellos. Por ejemplo en nuestra plataforma de referencia si el servidor tiene IP 192.168.1.1 y en este se configura un servidor de nombres, en todos los computadores de la red el archivo `/etc/resolv.conf` debe incluir:

```
search micolegio.edu.co
nameserver 192.168.1.1
```

Si conectará su sistema a Internet podría incluir más servidores de nombres especificando cada uno con líneas iniciadas con la palabra `nameserver` (podría especificarlos a continuación de su propio servidor para que la información de su servidor tenga prelación dentro de la LAN). La línea `search` indica dominios con los cuales completar direcciones (por ejemplo `purpura` será completado como `purpura.micolegio.edu.co`). El archivo `/etc/host.conf` de todos los computadores puede ser:

```
order hosts,bind
multi on
```

que indica en la primera línea el orden para resolver nombres, en este caso como `hosts` está primero, resolverá nombres primero revisando el archivo `/etc/hosts` y en segundo lugar empleará los servidores de nombres configurados en `/etc/resolv.conf` (también podría especificarse `nis` en caso de que se resolvieran dominios usando NIS). La línea **multi on** indica que el resolvedor debe retornar todas las direcciones IP que pueda encontrar para un nombre en el archivo `/etc/hosts` y no sólo la primera (útil en una red *multihomed* con varios IPs para una misma máquina). Cómo en nuestra plataforma de referencia empleamos DNS para resolver nombres de dominios, el archivo de configuración de NIS (`/etc/nsswitch.conf`) debe incluir la línea (ver Servicio NIS):

```
hosts: files dns
```

que de forma análoga a las líneas en `/etc/hosts.conf` indica buscar primero en `/etc/hosts` y después usar los servidores de nombres DNS.

Los resultados de las consultas que un resolvedor de nombres hace a un servidor de nombres, son registros de recursos (RR del inglés *Resource Records*). Este mismo tipo de registros se usa para configurar un servidor de nombres. Un ejemplo es:

```
servidor.micolegio.edu.co.   IN  A  192.168.1.1
                            A  128.9.0.32
micolegio.edu.co.             MX  10 servidor.micolegio.edu.co
```

que describe información de nombres Internet (indicado por la sigla `IN` de la primera línea y asumida por defecto en las demas). Las dos primeras líneas son tipo dirección (el tipo `A` –de *Address*– lo indica), mientras que la tercera identifica un intercambiador de correo para el dominio `micolegio.edu.co` (tipo `MX` –de *Mail Exchange*). Los dos primeros registros presentan información sobre servidor.micolegio.edu.co, la ausencia de otro nombre al comienzo de la segunda línea indica que es una segunda dirección para servidor.micolegio.edu.co. Los registros tipo dirección permiten asociar una IP con un nombre, así en el ejemplo anterior el nombre servidor.micolegio.edu.co se referirá a dos direcciones IP. Los registros tipo `MX` identifican computadores que pueden resolver direcciones de correo para un dominio, cada uno con un valor de preferencia (empleando para determinar que servidor se emplea primero si hay varios intercambiadores de correo para un mismo dominio). Note que los nombres de dominios completos terminan con un punto, esto es un requerimiento propio de la configuración como se explica más adelante.

Otros tipos pueden ser: `CNAME` que identifica el nombre principal de un alias, `HINFO` que identifica CPU y sistema operativo de un computador, `NS` que marca un servidor de nombres autoritario para un dominio (autoritario quiere decir que su respuesta es la que se considera más acertada en caso de que otros servidores den otras respuestas); `PTR` para referenciar otra parte del espacio de nombres de dominio; `SOA` (*start of authority*) para marcar el comienzo de una zona de autoridad.

Además de asociar nombres a IPs, pueden asociarse IPs a nombres para permitir hacer resolución inversa (dada la IP encontrar el nombre). Para lograrlo se emplean registrros PTR, usando como nombre de máquina la IP tras invertir el orden de los 4 bytes que la conforman y seguida de IN-ADDR.ARPA. Por ejemplo:

```
1.2.168.192.IN-ADDR.ARPA.        IN       PTR      servidor.micolegio.edu.co
```

6.4.1.2.3. DNS a nivel de servidor de nombres

Las consultas formuladas por los resolvedores de nombres que no pueden ser resueltas por el depósito local del resolvedor ni con otros mecanismos (e.g. `/etc/hosts`), pueden ser contestadas por uno de los servidores siguiendo el algoritmo descrito en el RFC 1034:

1. El resolvedor contacta a uno de los servidores configurados.
2. El servidor busca la zona que sea ancestro más cercano de la información buscada.
3. Si la información buscada está entre las zonas manejadas o referenciadas por el servidor, busca entre ellas hasta encontrar la información o hasta encontrar un registro que referencie otro servidor de nombres autoritario o en último caso reporta que el dominio no fue encontrado.
4. Si la información no está en las zonas manejadas por el servidor (o está en una subzona), busca en un deposito local.
5. Si no hay información en el deposito local y está en modo recursivo, empleando su resolvedor de nombres consulta al servidor encontrado en el paso 3 y retorna la respuesta además de almacenarla en el deposito. En modo iterativo sólo responde el nombre del servidor encontrado en el paso 3.
6. Una vez el resolvedor consigue la información, la almacena temporalmente en un depósito local.

Para distribuir la información en los servidores de nombres, el árbol de nombres se divide en zonas, cada una de las cuales puede ser atendida con registros autoritarios por un servidor primario. Para respaldar la información de servidores primarios, hay servidores secundarios que copian periódicamente la información y también proveen información autoritaria de la zona.

Para hacer consultas a servidores de nombres o para recibir respuestas, se emplean los puertos UDP 53 y TCP 53 (como puertos de entrada y de salida, aunque para la comunicación con algunos programas cliente pueden requerirse puertos no privilegiados (TCP 1024 y UDP 1024).

Entre los servidores DNS disponibles en Internet están djbdns y bind. Aunque djbdns es más seguro, bind (paquete `bind`, documentación en paquete `bind-doc`) es más popular y será el servidor que describimos en esta sección. La instalación por defecto de este paquete dejará una configuración sencilla que consta de:

/etc/bind/named.conf

: Archivo de configuración principal que define zonas y opciones del servidor.

`/etc/bind/db.local` y `/etc/bind/db.127`

Que definen la zona localhost (interfaz loopback) y su zona inversa (i.e direcciones 127.*.*.*)

`/etc/bind/db.root`

Que referencia los servidores de la raíz del árbol de nombres.

`/etc/bind/db.0` y `/etc/bind/db.255`

Que incluyen registros para hacer resolución inversa de direcciones de la forma 0.*.*.* (también localhost) y 255.*.*.* (broadcast).

Las zonas instaladas por defecto son recomendadas en el RFC 1912, y deben estar configuradas en todo servidor de nombres.

Para configurar una zona para su red local, es preferible que emplee un nombre de dominio que no esté registrado en Internet (aún si no planea conectarse a Internet a corto plazo). Cree un archivo para su zona, digamos `/etc/bind/db.micolegio.edu.co` y otro para resolución inversa en su red, digamos `/etc/bind/db.192.168.1` (que resolverá direcciones de la forma 192.168.1.*).

En el archivo de configuración principal (`/etc/bind/named.conf`), debe agregar entradas para la zona. Debe tener líneas como las siguientes, aunque quitando o agregando comentarios de ser necesario (son comentarios los textos iniciadas con // hasta final de línea o porciones encerradas entre /* y */ o líneas iniciadas con el caracter #).

```
options {
  directory "/var/cache/bind";
  forwarders {
//Remplazar por dirección del servidor DNS del proveedor si lo hay
// o comentar esta sección.  Esta dirección será usada en caso
// de que este servidor no tenga un registro en una zona manejada
// por él o en el deposito.
    1.2.3.4;
    };
    // Quitar comentarios si se hacen conexiones esporadicas a Internet
    // con un modem para actualizar zonas
    // dialup yes
}
logging {
  category lame-servers { null; };
  category cname { null; };
  };
  // Las siguientes líneas se comentan si la red no se conecta a
  // Internet
zone "." {
  type hint;
  file "/etc/bind/db.root";
};
zone "localhost" {
  type master;
  file "/etc/bind/db.local";
};
zone "127.in-addr.arpa" {
  type master;
```

```
  file "/etc/bind/db.127";
};
zone "0.in-addr.arpa" {
  type master;
  file "/etc/bind/db.127";
};
zone "255.in-addr.arpa" {
  type master;
  file "/etc/bind/db.255";
};
zone "micolegio.edu.co" {
  type master;
  file "/etc/bind/db.micolegio.edu.co";
};
zone "1.168.192.in-addr.arpa" {
  type master;
  file "/etc/bind/db.1.168.192";
};
```

Note que se configuran las nuevas zonas como tipo `master` porque el servidor podrá dar respuestas autoritarias sobre estas.

Los archivos con registros de recursos de cada zona, pueden tener comentarios en líneas que comienzan con ';'. El archivo `/etc/bind/db.local` que define una zona para la interfaz loopback —que siempre tiene dirección 127.0.0.1 y nombre `localhost`— sería:

```
@ IN SOA localhost. root.localhost. (
      1 ; Serial
 604800 ; Refresco
  86400 ; Reintento
2419200 ; Expiración
 604800 ) ; TTL Mínimo

@ IN NS localhost.
@ IN A 127.0.0.1
```

Note que se define una zona sobre la que este servidor tiene autoridad usando un registro SOA; cómo el registro ocupa más de una línea se emplean paréntesis para delimitarlo. El signo '@' es una abreviación del nombre de dominio base, es decir el que proviene del archivo `/etc/bind/named.conf` (en este caso `localhost`).

El registro SOA tienen el nombre completo de la máquina que mantiene la información (terminada en punto), seguida de la dirección electrónica del administrador del servicio, pero remplazando el caracter '@' por '.' y terminando con '.' [33] Los registros tipo SOA incluyen: un número serial que los servidores secundarios usan para determinar cuando la información de la zona del primario a cambiado, el número de refresco indica cada cuantos segundos un servidor secundario revisará el primario para determinar si el número serial ha aumentado, un número de reintentos que un servidor secundario debería efectuar en caso de ausencia de respuesta. Expiración es el tiempo máximo en segundos que un servidor secundario debe esperar antes de declarar la información que tenga de una zona como válida en ausencia de respuesta del servidor primario. Mínimo TTL indica el tiempo mínimo que los registros pueden estar en

33. Estos archivos de registros son preprocesados, y a todo nombre que no termine con el caracter '.' se le agrega la zona (definida en `/etc/bind/named.conf`) para convertirlo en un nombre completo (FQDN - *Fully qualified domain name*).

el deposito de otros servidores de nombres. Es el dato más importante. Si no se planean hacer cambios puede dejarse en un valor grande (3 días), si se planea un cambio es mejor disminuir este valor con suficiente anticipación y volverlo a un valor grande después del cambio. Note que para este ejemplo como se trata de `localhost`, estos valores pueden ser arbitrarios (en este caso muy grandes).

La información de la zona raíz corresponde a las direcciones de los 13 servidores de la raíz, puede obtenerse con:

```
dig @a.root-servers.net. . ns
```

En el momento de este escrito es:

```
. 518400 IN NS K.ROOT-SERVERS.NET.
. 518400 IN NS L.ROOT-SERVERS.NET.
. 518400 IN NS M.ROOT-SERVERS.NET.
. 518400 IN NS I.ROOT-SERVERS.NET.
. 518400 IN NS E.ROOT-SERVERS.NET.
. 518400 IN NS D.ROOT-SERVERS.NET.
. 518400 IN NS A.ROOT-SERVERS.NET.
. 518400 IN NS H.ROOT-SERVERS.NET.
. 518400 IN NS C.ROOT-SERVERS.NET.
. 518400 IN NS G.ROOT-SERVERS.NET.
. 518400 IN NS F.ROOT-SERVERS.NET.
. 518400 IN NS B.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET. 3600000 IN A 193.0.14.129
L.ROOT-SERVERS.NET. 3600000 IN A 198.32.64.12
M.ROOT-SERVERS.NET. 3600000 IN A 202.12.27.33
I.ROOT-SERVERS.NET. 3600000 IN A 192.36.148.17
E.ROOT-SERVERS.NET. 3600000 IN A 192.203.230.10
D.ROOT-SERVERS.NET. 3600000 IN A 128.8.10.90
A.ROOT-SERVERS.NET. 3600000 IN A 198.41.0.4
H.ROOT-SERVERS.NET. 3600000 IN A 128.63.2.53
C.ROOT-SERVERS.NET. 3600000 IN A 192.33.4.12
G.ROOT-SERVERS.NET. 3600000 IN A 192.112.36.4
F.ROOT-SERVERS.NET. 3600000 IN A 192.5.5.241
B.ROOT-SERVERS.NET. 3600000 IN A 128.9.0.107
J.ROOT-SERVERS.NET. 3600000 IN A 192.58.128.30
```

Si en la zona micolegio.edu.co el servidor de nombres tiene IP 192.168.1.1 con nombre `servidor` y algunos clientes con otras direcciones de la forma 192.168.1.*, el archivo `/etc/bind/db.micolegio.edu.co` sería:

```
@       IN      SOA     servidor.micolegio.edu.co. root.localhost. (
                            1 ; Serial
                          10800 ; Refresco cada 3 horas
                           3600 ; Reintento cada hora
                        2419200 ; Expiracion después de un mes
                         259200 ) ; TTL 3 dias

@             NS        servidor
servidor.micolegio.edu.co          A 192.168.1.1
www                     A 192.168.1.1
mail                    A 192.168.1.1
```

```
bachillerato1           A 192.168.1.2
bachillerato2           A 192.168.1.3
bachillerato3           A 192.168.1.4
bachillerato4           A 192.168.1.5
secretaria              A 192.168.1.100
profesores              A 192.168.1.200
```

Note que además de indicar que el archivo tiene información autoritaria sobre la zona con un registro SOA (que tiene parámetros más apropiados para un servidor de nombres primario); este ejemplo tiene: un registro tipo NS que presenta a `servidor` como servidor de nombres. Los demás registros tipo A asocian nombres a direcciones en la red. Note que todos los nombres de máquinas excepto el del registro SOA como no terminan en punto serán completados con el nombre de la zona (micolegio.edu.co declarada en `/etc/bind/named.conf` y referenciable como la variable `$ORIGIN`).

En cuanto a las zonas inversas, `/etc/bind/db.0` es

```
@ IN SOA localhost. root.localhost. (
      1 ; Serial
 604800 ; Refresh
  86400 ; Retry
2419200 ; Expire
 604800 ) ; Negative Cache TTL
@ IN NS localhost.
```

La zona inversa para localhost (`/etc/bind/db.127`):

```
@ IN SOA localhost. root.localhost. (
      1 ; Serial
 604800 ; Refresh
  86400 ; Retry
2419200 ; Expire
 604800 ) ; Negative Cache TTL
@ IN NS localhost.
1.0.0 IN PTR localhost.
```

para la zona de broadcast `/etc/bind/db.255`:

```
@ IN SOA localhost. root.localhost. (
      1 ; Serial
 604800 ; Refresh
  86400 ; Retry
2419200 ; Expire
 604800 ) ; Negative Cache TTL
@ IN NS localhost.
```

y para la zona de su red (`/etc/bind/db.1.168.192`):

```
@ IN SOA servidor.micolegio.edu.co. root.localhost. (
      1 ; Serial
  10800 ; Refresco cada 3 horas
   3600 ; Reintento cada hora
2419200 ; Expiracion después de un mes
 259200 ) ; TTL 3 dias
```

```
@ IN NS servidor.micolegio.edu.co.
1 IN PTR servidor.micolegio.edu.co
1 IN PTR www.micolegio.edu.co.
1 IN PTR mail.micolegio.edu.co.
2 IN PTR bachillerato1.micolegio.edu.co.
3 IN PTR bachillerato2.micolegio.edu.co.
4 IN PTR bachillerato3.micolegio.edu.co.
5 IN PTR bachillerato4.micolegio.edu.co.
100 IN PTR secretaria.micolegio.edu.co.
200 IN PTR profesores.micolegio.edu.co.
```

Después de hacer sus cambios puede reiniciar bind (lo cual limpiará el depósito y leerá de nuevo los archivos de configuración) con: **/etc/init.d/bind restart** (ver Inicialización del sistema).

Al editar zonas tenga en cuenta estas recomendaciones (de los RFC 1912 y 1034):

- Todo registro de recursos tipo `A`, debería tener el correspondiente registro tipo `PTR` para resolución inversa.
- No emplee en registros `CNAME` el mismo nombre con otros propósitos, ni emplee nombres de registros `MX`.
- No referencie alias definidos con `CNAME` en registros tipo `SOA`, `MX` ni `CNAME`.
- Si mantiene un servidor primario, cada vez que haga modificaciones a información de la zona que maneja, recuerde aumentar el número serial del registro SOA para que el o los servidores secundarios realicen la actualización.

6.4.1.3. Servicio NFS

Como se describe en el RFC 1813, el protocolo NFS [34] permite acceder de forma transparente sistemas de archivos compartidos que están en máquinas remotas. Hay muchas posibilidades para usar este servicio, nuestra plataforma de referencia lo aprovecha para distribuir información de usuarios (directorio `/home` del servidor), las colas de correo (`/var/mail`) y los programas y documentos disponibles en el servidor (directorio `/usr`). Así mismo permite aprovechar el espacio de sobra de cada cliente (directorio `/aux`).

Al igual que otros servicios, NFS cuenta con un cliente y un servidor. El servidor NFS permite exportar directorios del computador en el que corre a computadores donde se ejecute el cliente, mientras estos últimos tengan permiso para importar tales directorios. Los directorios que se exportan, así como las restricciones sobre los clientes que pueden importarlos se especifican en el archivo `/etc/exports`. Por ejemplo el siguiente es el archivo `/etc/exports` del servidor de nuestra plataforma de referencia:

```
/usr  *.micolegio.edu.co(ro,no_root_squash)
/home  *.micolegio.edu.co(rw,no_root_squash)
/var/mail  *.micolegio.edu.co(rw,no_root_squash)
```

Este archivo especifica que pueden exportarse con permiso de lectura y escritura los directorios `/home`, `/var/mail`. Puede exportar con permiso de sólo lectura (`ro`) el directorio `/usr`. Todos estos directorios

34. NFS es sigla de *Network File System* que en castellano es sistema de archivos en red. Anque en el momento de este escrito hay versión 4, en Debian está implementada la 3.

pueden ser importados por máquinas con nombres de la forma `x.micolegio.edu.co`. La opción `no_root_squash` indica que los archivos de usuario y grupo `root` exportados del servidor sean tratados como si fueran del usuario y grupo `root` en los clientes.

En nuestra plataforma de referencia tanto cliente NFS como servidor NFS deben instalarse en todos los computadores (porque los computadores clientes exportarán el espacio que resta de su partición `aux` al servidor). El archivo `/etc/exports` de cada cliente debe ser algo como:

```
/aux *.micolegio.edu.co(rw,no_root_squash)
```

Para instalar el servidor y el cliente NFS en Debian 2.2 basta que instale los paquetes `nfs-common` y `nfs-server`, siguiendo el procedimiento usual (ver Administración de programas). Como NFS depende de RPC, asegúrese también de dar acceso a las máquinas de su dominio con `portmap` y que esté operando. Dado que `portmap` es manejado con **tcpd** (ver Configuración de servicios básicos) este acceso se da o restringe modificando los archivos `/etc/hosts.allow` y `/etc/hosts.deny`. Por ejemplo el archivo `/etc/hosts.allow` debe tener una línea como:

```
portmap: .micolegio.edu.co
```

Puede comprobar que portmap está corriendo buscándolo entre los procesos (i.e **ps ax | grep "[p]ortmap"**) o revisando los programas que están registrados para usar RPC con `pmap_dump`.

Una vez esté corriendo el servidor y el cliente NFS en todas las máquinas, puede montar (ver Montaje y desmontaje de sistemas de archivos) los directorios exportados por el servidor en cada cliente, por ejemplo con algo como:

```
mount -t nfs servidor.micolegio.edu.co:/usr /opt
```

para montar el directorio `/usr` del servidor como el directorio `/opt` de cada cliente. Mejor aún, puede editar el archivo `/etc/fstab` para que cada vez que cada máquina inicie monte automáticamente ese directorio. Por ejemplo podría agregar la siguiente línea al archivo `/etc/fstab` de un cliente [35]:

```
servidor.micolegio.edu.co:/usr /opt nfs ro 0 0
```

En el servidor puede agregar al archivo `/etc/fstab`, líneas de la forma "`clienten:/aux /mnt/auxn nfs rw 0 0`" para montar en `/mnt/auxn` el directorio `/aux` de cada cada cliente. Para comprobar los directorios que ha montado con NIS puede emplear **mount**.

Mientras no configure el servicio NIS (ver Servicio NIS), recomendamos no montar `/home` ni `/var/mail` del servidor en los clientes.

6.4.1.4. Servicio NIS

NIS [36] es un servicio para centralizar nombres de usuarios, claves, e información de grupos en el servidor facilitando la administración de usuarios. Si además de NIS se usa NFS para montar el directorio `/home` del servidor en cada cliente, puede centralizarse también la información de todos los usuarios en el servidor.

35. Después de montar `/usr` del servidor como `/opt` de cada cliente, podrá ejecutar algunos programas que están en el servidor, desde el directorio `/opt` (aún para más comodidad puede agregar las rutas `/opt/bin` a la variable de entorno PATH).
36. NIS es sigla de *Network Information System* que en castellano es sistema de información de la red

NIS de forma análoga a DNS opera sobre un grupo de computadores (dominio) y mantiene bases de datos (mapas) centralizadas en un servidor maestro, que pueden ser consultadas por los clientes. Para disminuir carga podrían ponerse servidores esclavos que repliquen la información del servidor maestro.

Para usar NIS debe escoger un nombre de dominio NIS (puede ser diferente al dominio DNS) y usarlo en los computadores clientes y en el servidor.

Puede comenzar instalando el paquete `nis` tanto en clientes como servidor, al instalarlo podrá dar el dominio NIS (o puede editarlo en `/etc/defaultdomain`). En todos los computadores debe modificar el archivo `/etc/nsswitch` para cambiar el orden de búsqueda de usuarios, grupos y *shadow*. También debe agregar algunas líneas al final de los archivos `/etc/passwd`, `/etc/group` y `/etc/shadow`, tarea que puede hacer con los siguientes comandos:

```
echo "+::::::" » /etc/passwd
echo "+:::" » /etc/group
echo "+::::::::" » /etc/shadow
```

Que agregan líneas con un "+" y tantos ":" como separadores hay en los respectivos archivos. En el servidor los usuarios y grupos que estén después de estas marcas no serán compartidos por NIS.

En el servidor debe configurar NIS de la siguiente forma:

1. Edite `/etc/init.d/nis`, asegurándose de dejar `NISSERVER=master`.
2. Reinicie el servicio NIS con **/etc/init.d/nis restart**
3. Edite el archivo `/var/yp/Makefile` y cambie la regla `all:` para que incluya también `shadow`.
4. Ejecute **/usr/lib/yp/ypinit -m**.

Una vez NIS esté funcionando en clientes y servidor, puede agregar, eliminar o modificar usuarios y grupos con los comandos usuales (ver Administración de usuarios) y después de cada modificación, para que el cambio sea notado por NIS, debe pasar al directorio `/var/yp/` y ejecutar **make**.

Para centralizar la información de todos los usuarios en el servidor puede usar NFS una vez NIS funcione bien. Para lograrlo se debe hacer la administración de usuario siempre en el servidor (por ejemplo agregar nuevos usuarios sólo desde el servidor) para que los directorios queden allí; por otra parte debe montar el directorio `/home` del servidor en todos los clientes. Para centralizar la cola de correo en el servidor debe montar el directorio `/var/mail` del servidor en los clientes. De esta forma el archivo `/etc/fstab` de cada cliente debe incluir:

```
servidor.micolegio.edu.co:/home /home nfs rw 0 0
servidor.micolegio.edu.co:/var/mail /var/mail nfs rw 0 0
```

y el archivo `/etc/exports` del servidor debe tener las líneas apropiadas para exportar `/home` y `/var/mail`. Ver Servicio NFS.

6.4.1.5. Servicio `ssh`

El protocolo ssh cuenta con dos versiones, un borrador de la segunda de estas está en proceso de desarrollo. OpenSSH es la implementación de cliente y servidor para estos protocolos, la versión disponible para Debian permite usar tanto ssh 1 como ssh 2.

Tal como se describe en uno de los borradores de la especificación temporal "*SSH Protocol Architecture*" (http://www.ietf.org/internet-drafts/draft-ietf-secsh-architecture-13.txt) `ssh` es un protocolo para iniciar sesiones en máquinas remotas que ofrece autenticación, confidencialidad e integridad. Consta de tres componentes:

1. Protocolo de transporte. Que normalmente opera sobre TCP/IP dando autenticidad, confidencialidad e integridad.
2. Protocolo de autenticación de usuario. Que autentica al usuario ante el servidor.
3. Protocolo de conexión. Que multiplexa un canal encriptado en diversos canales lógicos.

Este protocolo requiere que los servidores tengan "llaves", las cuales son usadas por los clientes cada vez que se conectan a un servidor para verificar que no fue suplantado. Una llave es un número codificado y encriptado en un archivo. Para la encripción de llaves, OpenSSH ofrece los algoritmos RSA y DSA (de los cuales para la versión 2 recomendamos DSA).

Para instalar un servidor **OpenSSH**, que le permita conectarse a su sistema de forma segura, instale el paquete ssh preferiblemente tomando la versión más reciente del sitio de seguridad de Debian: http://security.debian.org o compile las fuentes más recientes que puede obtener en http://www.openssh.org. Cuando instale se generaran un par de "llaves" para su computador, una pública y una privada. Una vez instalado podrá afinar la configuración del servidor en el archivo `/etc/ssh/sshd.conf` que puede incluir líneas como las siguientes:

```
PermitRootLogin no
RSAAuthentication yes
PubkeyAuthentication yes
RhostsAuthentication no
hostsRSAAuthentication no
HostbasedAuthentication no
X11Forwarding yes
```

La última línea permitirá a los clientes que se conecten ejecutar aplicaciones de *X-Window* y transmitir la información gráfica sobre la conexión segura (ver Sección 2.3.1.1).

Un usuario también puede crear un par de llaves que le faciliten su autenticación al emplear **ssh** o **scp**. Estos programas por defecto piden clave al usuario que se conecte a un servidor ssh. Si un usuario genera sus llaves pública y privada, puede saltarse esta autenticación pues se hará de forma automática con las llaves. Para lograrlo su llave pública debe estar en el computador al cual se conecta (en `~/.ssh/authorized_keys`) y su llave privada en el computador desde el cual se conecta (normalmente en `~/.ssh/id_dsa`).

La generación de llaves puede hacerse con:

```
ssh-keygen -t dsa
```

que por defecto dejará su llave pública en `~/.ssh/id_dsa.pub` y su llave privada en `~/.ssh/id_dsa` (que además quedará protegida por una palabra clave que usted especifica). Como el nombre lo índica la llave privada no debe compartirla, por el contrario la llave pública puede transmitirla y puede ser vista por cualquiera.

En el computador en el que desee conectarse, agregue en el archivo `~/.ssh/authorized_keys` (o `~/.ssh/authorized_keys2` si usa DSA y una versión de OpenSSH anterior a la 3.1), su llave pública. Por ejemplo el usuario mario desde purpura.micolegio.edu.co puede configurar la entrada con autenticación automática a la cuenta pepe en amarillo.micolegio.edu.co con:

```
purpura> scp ~/.ssh/id_dsa.pub amarillo.micolegio.edu.co:/home/pepe/id_dsa_mario.pub
purpura> ssh -l pepe amarillo.micolegio.edu.co
...
amarillo> cat id_dsa_mario.pub » ~/.ssh/authorized_keys
```

Cuando mario se intente conectar desde purpura, a la cuenta pepe en amarillo ya no tendrá que dar la clave de pepe en ese computador sino la palabra clave con la que protegio su llave privada. Incluso esta palabra clave puede darse una sóla vez, aún cuando se realicen diversas conexiones con:

```
purpura> ssh-agent bash
purpura> ssh-add mario
```

Tras lo cual mario tecleará una vez la palabra clave de su llave privada, y después en esa sesión de **bash** todo ingreso que haga a la cuenta pepe en amarillo, no solicitará clave alguna.

6.4.1.6. Servicio CVS

En esta sección se presenta configuración de un servidor CVS y creación de repositorios (el uso de un repositorio y un servidor ya configurado puede verse en Sección 3.1.1.5).

Un servidor CVS puede emplearse sin conexión a una red, o bien en red con alguno de los protocolos rsh, ssh (ver Servicio ssh) o pserver. Los dos primeros requieren que el usuario que se conecta tenga cuenta (o una llave pública ssh entre las llaves autorizadas de una cuenta), mientras que pserver puede manejar su propio archivo de usuarios y claves. En todos los casos requiere instalar CVS y configurar un depósito. Si usa ssh debe instalar también el servidor **sshd** (ver Servicio ssh), si usa pserver debe agregar a su archivo `/etc/inetd.conf` (ver Configuración de servicios básicos) la línea:

```
cvspserver      stream  tcp     nowait.400      root    /usr/sbin/tcpd
/usr/bin/cvs -b /usr/bin --allow-root=/var/cvs pserver
```

(cambiando `/var/cvs` por el directorio donde está el depósito) y debe configurar los repositorios que desee poder acceder con pserver en `/etc/cvs-pserver.conf`.

Para instalar CVS instale el paquete cvs (ver Paquetes en Debian), que le permitirá activar o no el protocolo pserver y de requerirse hará los cambios apropiados en `/etc/inetd.conf`. El programa de instalación también podrá inicializar un repositorio por usted (puede ser `/var/cvs`, directorio que usted debe crear antes o durante la instalación de cvs).

Una vez CVS (y eventualmente ssh) esté instalado, todo usuario de la misma máquina puede crear uno o más depósitos. Tales depósitos podrán ser accedidos por otros usuarios del mismo sistema o de otro con ssh o con rsh o con el protocolo pserver (ver Uso de CVS)

Un usuario pepe emplearía una secuencia de comandos como la siguiente para crear un repositorio en `/home/pepe/cvs`:

```
cd ~
```

```
mkdir cvs
cd cvs
export CVS_RSH=""
export CVSROOT=/home/pepe/cvs
cvs init
```

Pepe mismo o algún otro usuario del sistema que tenga permiso de escritura en ese directorio podrá leer (*update*) y escribir (*commit*) en el nuevo repositorio. Para eso basta que emplee:

```
export CVS_RSH=""
export CVSROOT=/home/pepe/cvs
```

También puede usarlo vía ssh (por ejemplo desde otro computador de la misma red) con `CVS_RSH="ssh"`. O si durante la instalación o reconfiguración de cvs el administrador activa el protocolo pserver puede emplearse `CVSROOT=:pserver:pepe@purpura.micolegio.edu.co:/home/pepe/cvs`, siendo pepe un usuario de la máquina donde está el repositorio con permisos para leer (o escribir) o un usuario para ese repositorio agregado en el archivo `/home/pepe/cvs/CVSROOT/passwd`[37]

Todo usuario con permiso de escritura en ese depósito podrá ingresar un módulo nuevo. Por ejemplo un usuario con permiso de escritura y las variables CVS_RSH y CVSROOT con los valores recién presentados, puede ingresar todo el contenido del directorio `~/quimica`‘ como módulo `laquimica`:

```
cd ~/quimica
cvs import laquimica start vendor
```

Tras lo cual el mismo usuario que importó el módulo, o cualquier otro con permiso de lectura, puede sacar una copia local con:

```
cvs co laquimica
```

y hacer otras operaciones usuales de CVS (ver Uso de CVS).

6.4.1.6.1. Caso: Correo por cada actualización

Aprovechando la capacidad de CVS de ejecutar un script cuando actualiza la bitácora y el script `syncmail`, es posible configurar CVS para que por cada actualización (*commit*) envíe un correo a una dirección electrónica. Para esto haga *checkout* del directorio `CVSROOT` de su repositorio :

```
cvs co CVSROOT
```

En ese directorio copie el archivo syncmail que podrá descargar de: http://cvs-syncmail.sourceforge.net (http://cvs-syncmail.sourceforge.net) y después ejecute:

```
chmod +x syncmail
cvs add syncmail
```

37. El archivo de claves para CVS tiene 2 o 3 campos, el primero es el nombre con el que se conectará el usuario, el segundo la clave encriptada con DES (ver Administración de usuarios) y el tercer opcional es el nombre del usuario local a nombre del cual modificará el repositorio. Este archivo puede administrarse con htpasswd, para crear el archivo con un primer usuario: **htpasswd -c pepe** y posteriormente para agregar otros usuarios **htpasswd agustin**

Después agregue al archivo `checkoutlist` la línea:

```
syncmail
```

actualice:

```
cvs commit -m "Ahora usa syncmail"
```

y finalmente edite y actualice el archivo `loginfo`, agregando líneas como la siguiente:

```
^mimodulo $CVSROOT/CVSROOT/syncmail %{sVv} paz@micolegio.edu.co
```

que indican que toda actualización al módulo `mimodulo`, debe generar un correo indicando el cambio a paz@micolegio.edu.co. La primera parte de la línea (`^mimodulo`) es una expresión regular que se aplicará al módulo o puede ser `DEFAULT` para indicar cualquier módulo.

6.4.1.7. Servicio de correo

El servicio de correo empleado en Internet y en una red TCP/IP se basa en el protocolo SMTP (*Simple Mail Transfer Protocol*) descrito especialmente en los RFCs 821 (ftp://ftp.rfc-editor.org/in-notes/rfc821.txt) y 1123 (ftp://ftp.rfc-editor.org/in-notes/rfc1123.txt), funcionando sobre TCP/IP. En una situación típica en la que un usuario pepe@primaria2.micolegio.edu.co envía un mensaje al usuario paz@bachillerato3.micolegio.edu.co sin computadores intermediarios, se requiere:

- Que haya conexión física y a nivel de TCP/IP entre ambos computadores.
- Que ambos computadores tengan un programa que permita enviar y recibir correo usando el protocolo SMTP, como por ejemplo exim, sendmail o postfix (a tal programa se le llama MTA - *Mail Transport Agent*).
- Que ambos usuarios tengan un programa con el que puedan leer y redactar correos, como por ejemplo mail, mutt, elm, balsa, pine, evolution (a ese programa se le llamará MUA - *Mail User Agent*[38]).

Si tanto Pepe como Paz emplean como MUA **mail**, y ambos computadores tiene como MTA exim, el proceso sería:

1. Pepe emplea **mail** en su computador primaria2 para redactar el mensaje cuyo destinatario es paz.
2. En primaria2, el programa mail ejecuta exim para enviar el mensaje. exim deja el mensaje en una cola de mensajes por enviar. Esa cola de mensajes es actualizada por exim a medida que envía o intenta enviar mensajes (si un mensaje no puede ser enviado exim puede reintentar el envio cierto número de veces, haciendo pausas entre un intento y otro).

   Enviar un mensaje significa crear una conexión TCP con el MTA destino o con otro MTA que actúe de intermediario, típicamente en el puerto TCP 25, y transmitir el mensaje siguiendo las reglas del

38. De acuerdo al RFC 1123 los nombre MUA y MTA son propios del protocolo X.400.

protocolo SMTP [39]. Para establecer el computador con el cual conectarse exim revisa con el resolvedor DNS, registros MX asociados con el dominio de la dirección, si los hay intenta enviar a cada uno en orden de prioridad –los registros MX con menor número tienen mayor prioridad (ver Servicio DNS).

3. En bachillerato3 debe estar corriendo un proceso que acepte la conexión en el puerto 25. Puede ser exim mismo o puede ser inetd que una vez realizada la conexión ejecuta a exim (ver Configuración de servicios básicos). Después exim recibirá el mensaje siguiendo el protocolo SMTP.
4. exim agrega el mensaje que recibe en el archivo tipo texto `/var/mail/paz`.
5. Cuando Paz lo desee, podrá emplear **mail** para leer los correos que se hayan acumulado en `/var/mail/paz` —a medida que los lea saldrán de ese archivo para quedar en `~/mbox`.

Este es el esquema básico, aunque hay muchas otras situaciones en las que se emplean otras posibilidades de SMTP, protocolos auxiliares y programas[40]. Como parte de nuestra plataforma de referencia, sugerimos este esquema:

- Emplear un MUA que pueda leer directamente los correos que están en `/var/mail` o `/var/spool/mail` y configurarlo para esto, (en Debian el primero es el preferido, y `/var/spool/mail` es un enlace a `/var/mail`) y que emplee un MTA para enviar correos (ejemplos de MUAs con estás características son: mail, mutt, elm, balsa, evolution).
- Emplear exim como MTA en todos los clientes. Una ventaja de exim sobre otros MTA (como sendmail) es la facilidad con la que se configura en el archivo `/etc/exim/exim.conf` o aún más sencillo en Debian para muchas situaciones con el script `eximconfig`. Los computadores clientes se configuran como MTA satélites con el servidor como compuerta de correo (o *smart host*) —es decir se configura en los clientes sin capacidad de recibir correos sino sólo de enviar todo correo a un computador (al servidor). El MTA del servidor funciona como un MTA completo que pueda recibir y enviar correo usando SMTP.
- Emplear NFS para compartir el directorio `/var/mail` del servidor en todos los clientes. De forma que exista un único sitio físico donde estén todos los correos, pero que desde todos los computadores cada usuario pueda acceder a su archivo de correos.

Para configurar este esquema se requiere:

39. De acuerdo al protocolo SMTP, exim de primaria2 se conectaría por el puerto 25 a exim en bachillerato3 y enviaría el mensaje `MAIL FROM: pepe@primaria2.micolegio.edu.co`, después enviaría `RCPT TO: paz@bachillerato3.micolegio.edu.co`, después `DATA` y a continuación el cuerpo del correo comenzando con el encabezado de acuerdo al RFC 822, con un cuerpo de mensaje que emplee 7 bits y terminando con una línea que sólo tenga un punto. Por ejemplo

```
From: pepe@primaria2.micolegio.edu.co To: paz@bachillerato3.micolegio.edu.co Subject:
Saludo Un cortisimo saludo. .
```

Si lo desea puede experimentar con este protocolo, empleando telnet y el MTA de su computador: **telnet localhost 25**. Claro resulta más transparente empleando directamente exim (que tiene como alias el script `sendmail`):

```
sendmail -bm paz@bachillerato3.micolegio.edu.co -f pepe@primaria2.micolegio.edu.co
```

(para emplear `-f` con exim debe ser usuario autorizado).

40. Otro caso típico que evita tener MTA en el computador del usuario, requiere otro computador que almacene los correos y el protocolo POP3 (RFC 1939).

1. Instalar en todos los computadores los MUAs que habrá disponibles (mínimo mail disponible en el paquete `mailx`).
2. Instalar en todos los computadores el MTA exim (paquete `exim`). En los clientes configurarlo como sistema satélite. Como nombre de sistema emplee el dominio de su institución, como computador del cual leerán los correos y como *smart host* use el servidor. Para hacer un cambio en la configuración después de instalar puede emplear eximconfig, o con más práctica `/etc/exim.conf`. Algunas líneas que ese archivo debe tener en el caso de clientes, para enviar siempre por el servidor son:

```
qualify_domain = micolegio.edu.co
local_domains = micolegio.edu.co:localhost

smart:
  driver = smartuser
  new_address = ${local_part}@servidor.micolegio.edu.co
end

smarthost:
  driver = domainlist
  transport = remote_smtp
  route_list = "* servidor.micolegio.edu.co bydns_a"
end
```

También puede incluir reglas para reescribir direcciones de forma que parezcan provenir de servidor.micolegio.edu.co:

```
^(?i)(root|postmaster|mailer-daemon)@micolegio.edu.co ${1}@in.limbo Ffr
*@micolegio.edu.co ${1}@servidor.micolegio.edu.co Ffr
^(?i)(root|postmaster|mailer-daemon)@localhost ${1}@in.limbo Ffr
*@localhost ${1}@servidor.micolegio.edu.co Ffr
*@in.limbo root@servidor.micolegio.edu.co Ffr
```

y esta para agregar direcciones que deben cambiarse para ciertos usuarios en el archivo `/etc/email-adresses`[41]:

```
*@micolegio.edu.co    ${lookup{$1}lsearch{/etc/email-addresses}\
                                          {$value}fail} bcfrF
```

3. El servidor puede configurarlo como si fuera sitio de Internet usando **eximconfig** y agregando registros MX al servidor DNS. Configure como nombre del sistema el dominio de su institución, excluya de los controles de *relaying* su red interna (e.g `192.168.1.0/24`). De editar manualmente `/etc/exim/exim.conf`, entre las líneas que eximconfig incluye están las siguientes que configuran procmail (ver mutt y procmail):

```
procmail:
   driver = localuser
```

41. Cada línea de `/etc/email-adresses` será análoga a la siguiente, que indica reescribir la dirección del usuario pepe como pepe2@micolegio.edu.co: `pepe: pepe2@micolegio.edu.co`

```
   transport = procmail_pipe
   require_files =
   ${local_part}:+${home}:+${home}/.procmailrc:+/usr/bin/procmail
   no_verify
```

Con las siguientes permite redireccionamiento con el archivo `.forward`. Tenga en cuenta cambiar la configuración por defecto que deja eximconfig para esto (porque permite que `.forward` pueda ser escrito por el grupo):

```
userforward:
   driver = forwardfile
   file_transport = address_file
   pipe_transport = address_pipe
   reply_transport = address_reply
   no_verify
   check_ancestor
   file = .forward
   filter
```

Las siguientes indican emplear el sistema de correo local para usuarios locales, hacer consultas DNS antes de iniciar conexiones SMTP y emplear direcciones IP explícitas.

```
 localuser:
   driver = localuser
   transport = local_delivery

 lookuphost:
   driver = lookuphost
   transport = remote_smtp

 literal:
    driver = ipliteral
```

La configuración de DNS debería tener en cuenta emplear servidor.micolegio.edu.co como intercambiador de correo para el dominio micolegio.edu.co y también como intercambiador de correo para cada cliente (es decir que todo correo que vaya a uno de los clientes se redirija al servidor). Retomando el ejemplo de la zona de DNS (ver Servicio DNS), el archivo `/etc/bind/db.micolegio.edu.co` tendría dos nuevas líneas (con registros tipo MX):

```
@ IN SOA servidor.micolegio.edu.co. root.localhost. (
      1 ; Serial
  10800 ; Refresco cada 3 horas
   3600 ; Reintento cada hora
2419200 ; Expiracion después de un mes
 259200 ) ; TTL 3 dias

@ NS servidor
servidor.micolegio.edu.co A 192.168.1.1
www A 192.168.1.1
mail A 192.168.1.1
micolegio.edu.co. MX 5 mail
*.micolegio.edu.co. MX 5 mail
```

```
bachillerato1 A 192.168.1.2
bachillerato2 A 192.168.1.3
bachillerato3 A 192.168.1.4
bachillerato4 A 192.168.1.5
secretaria A 192.168.1.100
profesores A 192.168.1.200
```

Una vez haga cambios a la configuración de exim podrá realizar algunas pruebas para verificar la forma de enrutamiento:

```
exim -bV
exim -v -bt root@localhost
exim -v -bt root@servidor.micolegio.edu.co
exim -v -bt root@purpura.micolegio.edu.co
```

6.4.1.8. Servicio FTP

El protocolo FTP (que se describe en los RFC 1123 y en el RFC 959) permite transmitir archivos de un computador a otro de forma robusta y eficiente. Este protocolo requiere una conexión por la que se transmiten comandos y respuestas a comandos (normalmente por el puerto 21) y otra por la que se transmiten datos (el puerto es escogido por el cliente o el servidor). Pueden realizarse conexiones en modo activo o pasivo, en el primero el cliente establece la dirección y el puerto para el canal de datos, en el segundo es el servidor el que los establece. Al conectarse desde una red con direcciones privadas a un servidor FTP fuera de la red privada es necesario usar modo pasivo.[42]

Sugerimos emplear FTP para establecer un servidor FTP anónimo, pero no para brindar transmisión de archivos a usuarios del sistema (que pueden hacerlo con **scp** de ssh).

> **Aviso**
>
> Las claves transmitidas en sesiones de FTP viajan por la red sin encripción alguna.

42. Puede observarse con **telnet** el comportamiento de este protocolo por ejemplo:
`telnet ftp.ibiblio.org 21 user anonymous pass email@ pwd pasv retr index.html`
Como resultado del comando **pasv** el servidor FTP envía dirección y puerto al que debe conectarse el cliente para recibir la información. Si la respuesta es por ejemplo:
`227 Entering Passive Mode (152,2,210,81,230,214)`
el cliente debe abrir otra conexión para recibir los datos con la máquina con IP 152.2.210.81 (ftp.ibiblio.org) puerto 230*256+214 (59094). Al hacer:
`telnet 152.2.210.81 59094`
la conexión abierta recibirá el contenido del archivo `README`. Este ejemplo emplea FTP en modo pasivo, podría también indicarse modo activo empleando port en lugar de **pasv**. El comando **port** indica al servidor que inice una conexión con cierta dirección y puerto para realizar la transmisión, por ejemplo si en el cliente con dirección 1.2.3.4 se espera una conexión en el puerto 8000:
`port 1,2,3,4,31,64`
(note que 31*256+64 es 8000).

Como servidor FTP sugerimos proftp (paquete `proftpd`). La configuración por defecto queda en `/etc/proftpd.conf` que permite a todos los usuarios emplear ftp, permite ftp anónimo empleando los archivos del directorio `/var/ftp`. El directorio empleado puede tener una estructura arbitraria. proftpd deja errores y mensajes de error en `/var/log/syslog`.

Para limitar el servicio a ftp anónimo (los usuarios pueden emplear scp para realizar copias), puede agregar en la sección general del archivo de configuración limitación de acceso para todos los usuarios (con `LimitLOGIN`) y en la sección de ftp anónimo permitir acceso para todos los usuarios. El archivo `/etc/proftpd.conf` sería algo como:

```
ServerName "servidor"
ServerType standalone
DeferWelcome off

ShowSymlinks on
MultilineRFC2228 on
DefaultServer on
ShowSymlinks on
AllowOverwrite on

TimeoutNoTransfer 600
TimeoutStalled 600
TimeoutIdle 1200

DisplayLogin                    welcome.msg
DisplayFirstChdir               .message
LsDefaultOptions                "-l"

Port 21

Umask 022  022

MaxInstances 30

User nobody
Group nogroup

<Limit LOGIN>
    DenyAll
</Limit>

<Directory /*>
  AllowOverwrite on
</Directory>

<Anonymous ~ftp>
  User ftp
  Group nogroup
  UserAlias anonymous ftp

  RequireValidShell off
```

```
  MaxClients 10

  DisplayLogin welcome.msg
  DisplayFirstChdir .message

  <Limit LOGIN>
    AllowAll
  </Limit>

  <Directory *>
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>

</Anonymous>
```

6.4.1.9. Servicio Web

El servicio web se basa en el protocolo HTTP (*Hypertext Transfer Protocol*), cuya versión 1.1 está definida en el RFC 2616. Este protocolo permite:

- Transmitir hipertextos escritos en HTML con diversas codificaciones e idiomas, gráficas y otros tipos de información a partir de URLs solicitados por el navegador. Cada tipo de información solicitada por un cliente es transmitida por el servidor con una identificación MIME (*Multipurpose Internet Mail Extensions*) que permite al navegador desplegarlo apropiadamente (por ejemplo cargando otro programa o un plugin de ser necesario)[43]. La identificación MIME es de la forma tipo/subtipo, las identificaciones registradas pueden consultarse en: http://www.iana.org/assignments/media-types/index.html (http://www.iana.org/assignments/media-types/index.html)
- Sirve como protocolo genérico para transmisión de otros tipos de protocolos (e.g FTP, SMTP, Gopher, NNTP).
- Permite emplear intermediarios (proxys, compuertas y túneles), cada uno de los cuales puede tener un deposito. Normalmente HTTP emplea por defecto el puerto TCP 80 (si el servidor/proxy está en otro puerto, puede especificarse en un URL como `http://servidor.micolegio.edu.co:8080/index.html`).
- Ejecutar CGIs (*Common Gateway Interface*) que respondan dinámicamente a información transmitida desde el cliente. Por ejemplo el siguiente URL incluye las variables nombre y apellido que son pasadas al CGI `param.cgi` con valores `Pepe` y `Rojimes`: http://structio.sourceforge.net/cgi-bin/param.cgi?nombre=Pepe&apellido=Rojimes Los CGIs son

43. Puede verse en funcionamiento este protocolo empleando **telnet**, por ejemplo: `telnet structio.sourceforge.net 80 GET http://structio.sourceforge.net/index.html`

programas o scripts que reciben por entrada estándar y/o variables de ambiente, consultas hechas por un cliente y responden escribiendo en sálida estándar la respuesta que dará el servidor.

Como servidor web sugerimos Apache (paquete `apache`), que es el servidor más popular en Internet (puede consultar estadísticas en Netcraft (http://www.netcraft.com)) y para hacer más eficiente la conexión sugerimos emplear squid como proxy con depósito.

Apache se configura en el archivo `/etc/apache/httpd.conf`. La configuración por defecto supone que las páginas por servir están en `/var/www` y en los directorios `public_html` de cada usuario. Los errores y mensajes quedan en `/var/log/apache/error.log` y pueden dejarse CGIs en el directorio `/usr/lib/cgi-bin` (que es alias de `/cgi-bin`). Entre sus características están:

- Excelente implementación de HTTP 1.1 y CGI 1.1.
- Configuración sencilla (en `/etc/apache/httpd.conf`) y posibilidad de emplear archivos de confioguración por cada usuario `~/public_html/.htaccess`.
- Cuenta con módulos para soportar SSI (*Server Side Includes*) y PHP.
- Soporta SSL (*Secure Socket Layer*) para trasmisión encriptada y autenticación básica para tener zonas restringidas con claves.
- Soporta dominios virtuales (*Virtual Host*), para atender con un sólo servidor diversos dominios. Además de la configuración DNS que esto requiere (puede ser con registros CNAME), basta agregar una sección por cada dominio virtual, por ejemplo:

```
      <VirtualHost *>
ServerName estudiantes.micolegio.edu.co
DocumentRoot /var/www/estudiantes
</VirtualHost>
```

En cuanto a proxy, puede instalar squid (paquete `squid`). Este programa permite definir listas de acceso y emplearlas para dar servicio sólo a las direcciones con permiso. La configuración por defecto del paquete deja listo squid para servir sólo al computador donde se instala, para servir otros computadores de su red debe (1) agregar un ACL que especifique todos los computadores de su red y (2) indicar que el ACL definido puede emplear el proxy de HTTP, se hace con líneas como éstas en las secciones apropiadas del archivo de configuración `/etc/squid.conf`:

```
acl red src 192.168.0.0/255.255.0.0
```

y

```
http_access allow red
```

Tras hacer cambios en la configuración puede reiniciarlo con

```
/etc/init.d/squid restart
```

Puede comprobar que funciona o diagnosticar errores revisando las bitácoras del directorio `/var/log/squid`. Desde los clientes podrá configurar el navegador para que emplee este proxy que por defecto atiende el puerto 3128. Por ejemplo con **w3m** y **lynx** basta establecer el URL del proxy en la variable de ambiente `http_proxy`:

```
export http_proxy=http://servidor.micolegio.edu.co:3128
```

En el caso de mozilla, se configura en Advanced/Proxies después de elegir Edit/Preferences en el menú principal.

6.4.1.10. Impresora en red

El programa **lpd** es un servidor del protocolo LPD, mientras que **lpr** es un cliente. En esta sección se dan detalles mínimos para configurar una impresora en red, la configuración de una impresora local a un computador y la configuración básica de `/etc/printcap` puede consultarse en la sección Sección 5.4.1.5 y el uso de la impresora una vez configurada en Sección 3.1.1.1.

La impresora de la red debe conectarse a un computador y configurarse como local en ese computador. En el resto de computadores debe configurarse como impresora remota. Por ejemplo si el comptuador al que está conectado la impresora es `rojo.micolegio.edu.co` y allí se configura una impresora con nombre `psprn`, en los demás computadores en el archivo `/etc/printcap` debe haber líneas como:

```
psprn|rp|remote line printer:\
  :lp=:rm=rojo.micolegio.edu.co:rp=psprn:sd=/var/spool/lpd/psprn: \
  :lf=/var/log/lp-errs:
```

El servidor **lpd** que debe ser iniciado durante el arranque de `rojo`, recibe conexiones sólo de las máquinas especificadas en `/etc/hosts.equiv` o `/etc/hosts.lpd` y podrá entonces imprimir trabajos iniciados por usuarios en tales máquinas (agregue a uno de esos archivos los computadores clientes que podrán imprimir). Para restringir acceso a una impresora de otra forma, en `/etc/printcap` puede emplearse `rs` en lugar de `rm` para indicar que tal impreso sólo imprimirá trabajos de usuarios que tengan una cuenta en el mismo computador.

6.4.2. Lecturas recomendadas: Protocolos de soporte y de usuario

- *Net-HOWTO. 5.8*. Configuring your network servers and services. http://linuxdoc.org/HOWTO/Net-HOWTO/x619.html
- El tema seguridad no es tratado en profundidad (es indispensable estudiarlo si se planea conectar a Internet). *NET-3-HOWTO* Sección 5.10. Network Security and access control.

Configuración de servicios básicos

- Para conocer más sobre la configuración de `inetd` y `tcpd`, puede consultar las páginas man de inetd, tcpd y hosts_access.

- Puede consultar sobre el protocolo Telnet en los RFC 854 (ftp://ftp.rfc-editor.org/in-notes/rfc854.txt), 855 (ftp://ftp.rfc-editor.org/in-notes/rfc855.txt) y 1123 (ftp://ftp.rfc-editor.org/in-notes/rfc1123.txt).
- Sobre el protocolo de finger en el RFC 1288 (ftp://ftp.rfc-editor.org/in-notes/rfc1288.txt).

NFS

- La versión 3 de este protocolo está descrita en el RFC 1813 (ftp://ftp.rfc-editor.org/in-notes/rfc1813.txt).
- Páginas del manual de `exports` (**man exports**), de NFS y de portmap.
- De Olaf Kirch y Terry Dawson: *Linux Network Administrators Guide*. The Network File System. http://www.linuxdoc.org/LDP/nag2/x-087-2-nfs.html

DNS

- Puede consultar una exposición no técnica sobre DNS en http://www.internic.net/faqs/authoritative-dns.html
- Puede consultar sobre DNS en el RFC 1034 (ftp://ftp.rfc-editor.org/in-notes/rfc1034.txt) y recomendaciones en el RFC 1912 (ftp://ftp.rfc-editor.org/in-notes/rfc1912.txt).
- djbdns es otro servidor de nombre (no tan popular), en su página web puede consultarse información concisa sobre DNS http://cr.yp.to/djbdns.html
- Una explicación corta para la configuración de DNS en una red LAN en este artículo de Linux Gazette: http://www.linuxgazette.com/issue44/pollman/dns.html

NIS

- Una vez instale `nis` en Debian, en el directorio `/usr/doc/nis` encontrará un documento muy práctico y corto: `nis.debian.howto`. En Internet lo encuentra en: http://www.linux-nis.org/doc/nis.debian.howto

  NIS puede configurarse de forma más segura de la presentada en esta guía, puede manejar sus propios archivos de claves y grupos, permite tener servidores maestros y esclavos y muchas otras posibilidades que se describen brevemente en ese archivo.

- Una explicación sobre generalidades de NIS en: http://publib16.boulder.ibm.com/pseries/en_US/aixbman/nisplus/nis_intro.htm#a2b6a622144endr
- Puede averiguar más sobre NIS y sus variantes en el *NIS-HOWTO* http://www.linuxdoc.org/HOWTO/NIS-HOWTO/index.html o en el capítulo "The Network Information System" de "*Linux Network Administrators Guide*" http://www.linuxdoc.org/LDP/nag2/x-087-2-nis.html.

CVS

- Puede consultarse la arquitectura general de SSH 2 en http://www.ietf.org/internet-drafts/draft-ietf-secsh-architecture-13.txt

CVS

- Para saber más sobre CVS puede consultar la página man. En Internet puede consultar "*CVS–Concurrent Versions System*" http://www.cvshome.org/docs/manual/cvs.html (http://www.cvshome.org/docs/manual/cvs.html) o Open Source Development With CVS (http://cvsbook.red-bean.com/cvsbook.html). En particular sobre la configuración de syncmail y CVS: http://sourceforge.net/docman/display_doc.php?docid=772&group_id=1

Correo

- Especificación de exim 3.3: http://www.exim.org/exim-html-3.30/doc/html/spec.html

Web

- Puede consultar más sobre CGIs en la especificación: http://cgi-spec.golux.com/ y más sobre la configuración de Apache para usar CGIs en este tutorial: Dynamic Content with CGI (http://httpd.apache.org/docs/howto/cgi.html.html) y en la especificación propu.

6.4.3. Ejercicios: Protocolos de soporte y de usuario

1. Servicios

1.1. Active el servicio interno `daytime` de inetd, después revise la respuesta que este da a una conexión empleando **telnet**. Describa los pasos que siguió.

2. DNS

2.1. Desde el intérprete de comandos ¿Cómo puede determinar la IP de structio.sourceforge.net, los nombres DNS asociados a la IP 204.152.184.85 y la dirección de correo del contacto técnico de sourceforge.net? ¿Cuáles son esos datos?

3. NFS

3.1. En su máquina cree los directorios `/aux` y `/mnt/tmp`, ponga en el primero algún archivo. Después monte vía NFS el directorio `/aux` y móntelo en el directorio `/mnt/tmp`. Después de que funcione, desmóntelo.

3.2. Monte el directorio `/usr` del servidor principal como `/opt` en otro computador de la red. ¿Qué debe hacer para que en ese computador se emplee la ruta `/opt/bin` para buscar archivos y `/opt/lib` para buscar bibliotecas compartidas?

4. Correo

4.1. Configure exim en un computador cliente para no recibir correos y enviar empleando el MTA del servidor. Compruebe que funciona enviando desde el cliente un correo a una cuenta en el servidor.

4.2. Después de configurar NIS, empleando NFS comparta el directorio `/var/mail` del servidor en un cliente con la configuración del ejercicio anterior. Tras esto compruebe que cada usuario puede revisar su correo con el programa **mail** desde cualquier computador.

4.3. En caso de emplear un cliente de correo diferente a **mail** configúrelo de forma que lea correo de la cola `/var/mail` (o `/var/spool/mail`) y que transmita usando **exim**.

5. Web

5.1. Instale el servidor Apache en el servidor, compruebe que funciona desde un *browser* en otro computador.

5.2. Configure Apache para que puedan accederse los directorios de los usuarios con el URL `http://servidor/~login`.

Apéndice A. Plataforma de referencia S-Helio 1.1

A.1. Plataforma de referencia S-Helio 1.1

La plataforma de referencia S-Helio 1.1 es un diseño de una red de área local para un colegio, que se basa en Linux Debian y aplicaciones de libre redistribución. En esta red hay por lo menos un computador de más capacidad al que llamamos *servidor* y otros computadores tal vez menos poderosos a los que llamamos *clientes*. Al diseñarla hemos considerado:

- Emplear únicamente software que pueda usarse sin costo alguno y que resulte ser semilla y fruto de colaboración. Por esto sólo sugerimos software de libre redistribución de muy buena calidad. Ver Lectura Software de libre redistribución.
- Que la red se constituya en una intranet del colegio para ser usada por usuarios de uno de los siguientes grupos: estudiantes, profesores, personal administrativo y eventualmente padres y exalumnos.
- Que cada usuario cuente con su espacio propio, que disponga de herramientas básicas, que pueda comunicarse con otros usuarios de la red (`finger`, `talk`, correo electrónico) y que pueda consultar la documentación y los recursos que el colegio brinde en la intranet (NFS, `ftp`, `web`, `telnet`, `ssh`).
- Que el transito de otro sistema operativo a Linux sea lo menos abrupto posible. Por esto sugerimos un ambiente de escritorio sencillo (GNOME), escribimos documentación inicial de dominio público para usuar y administrar el sistema y ofrecemos soporte gratuito. Como las fuentes de esta documentación son de dominio público, pueden ser completadas, modificadas y usada en su colegio de acuerdo a sus políticas. Puede recibir soporte de voluntarios en la lista `<structio-sopcol@lists.sourceforge.net>`, a la que puede suscribirse en http://lists.sourceforge.net/lists/listinfo/structio-sopcol (http://lists.sourceforge.net/lists/listinfo/structio-sopcol), o si desea colaborar a desarrollar guías puede suscribirse a la lista `<structio-guias@lists.sourceforge.net>` desde http://lists.sourceforge.net/lists/listinfo/structio-guias (http://lists.sourceforge.net/lists/listinfo/structio-guias)
- Que el software y la documentación para el usuario final sean vistos en español y que el teclado pueda generar todos los caracteres de nuestro idioma.
- Suponemos que el colegio cuenta con PCs no tan modernos y que no podrá invertir grandes recursos económicos para actualizarlos (aun cuando esta plataforma de referencia S-Helio 1.1 también funciona con computadores nuevos).
- Suponemos que cada PC de la red es medianamente autónomo, es decir, que puede funcionar aislado de red con un sistema Linux mínimo[1].
- Buscamos distribuir al máximo el espacio de los discos de cada computador y la carga de los procesadores aprovechando para esto la red.
- Nos hemos basado en la experiencia en un colegio concreto[2].

1. Otro enfoque que permite usar computadores con aún menor capacidad puede consultarse en http://www.ltsp.org/
2. El Gimnasio Fidel Cano en Bogotá, Colombia. http://www.geocities.com/gimnasio_fidel_cano/proyectos.html

- Hemos tenido en cuenta otros diseños de red para colegios (RedEscolar[3].), software sugerido por organizaciones (OFSET[4]) y distribuciones enfocadas en educación (Debian-Jr[5]).
- Que la interconexión de diversas redes que sigan estas recomendaciones sea posible y fácil, así como una eventual conexión a Internet. Aunque en estas guías no se describe en detalle como lograr una conexión segura a Internet.

A continuación presentamos la plataforma de referencia S-Helio 1.1 en varias etapas: Red, Computadores y Software. Además al final de esta sección presentamos software de libre redistribución adicional que podría usarse en un colegio y que funciona bien sobre la plataforma de referencia.

Note que esta sección es de carácter técnico, precisamente estas guías le permitirán esclarecer los términos que aquí encuentre y pueden guiarlo en la instalación y uso de la plataforma de referencia S-Helio 1.1.

A.1.1. Red

Hardware:

Tarjetas

Ethernet 10Mb (e.g. Compatibles NE-2000).

Medio físico

Cable 10Base-T o de pares trenzados (en inglés *twisted pair*).

Concentrador

Software para distribución de información

NIS

Usado para distribuir claves que residirán en el servidor.

NFS

Usado para distribuir los directorios `/usr`, `/home`, `/var/mail` [6] y `/usr/share` del servidor en todos los clientes [7]

3. Distribución mejicana orientada a colegios http://redesc.linux.org.mx/es/
4. Organización francesa que promueve el uso de software libre en la educación http://www.ofset.org/
5. Puede verse el proyecto Debian-Jr en http://www.debian.org/devel/debian-jr/index
6. En Debian 2.2 `/var/spool/mail` es un enlace a `/var/mail`.
7. De acuerdo a http://www.pathname.com/fhs/2.0/fhs-2.html el estándar para jerarquías de Linux, esos directorios puede ser compartidos en red

`/usr`

se monta en los clientes como `/opt`, y por medio de la variables `PATH`, `LD_LIBRARY_PATH` (o el archivo `/etc/ld.so.conf`) y eventualmente enlaces, los usuarios pueden disponer de todos varios de los programas que hay en el servidor.

`/home`

se monta en todos los clientes como `/home`, para que los directorios de los usuarios estén disponibles en todas las estaciones.

`/var/mail`

se monta en todos los clientes como `/var/mail` para permitir que desde todos el programa `mail` funcione, mientras se centraliza la cola de correo en el servidor.

`/usr/share`

se monta en todos los clientes como `/usr/share` para dejar una sóla copia de la documentación, que pueda consultarse desde todos los computadores.

También se compartirá el directorio `/aux` de cada cliente en el servidor como `/mnt/aux-`*n*, de forma que en el servidor se pueda contar con todo el espacio que sobre en la red para redistribuirlo.

El servidor estará configurado como compuerta en todos los clientes (para una conexión sencilla a Internet de ser posible) y actuará como servidor de diversos protocolos (como se presenta a continuación).

A.1.2. Computadores

Servidor

- Procesador >= Pentium o AMD 686, Memoria: >=64MB.
- Tarjeta de video >=VGA, Monitor capaz de presentar gráficas 640x480x16.
- Disco Duro >=2GB dividido en particiones:
  - Partición para swap con la misma cantidad de memoria RAM o una cantidad cercana.
  - `/boot` con mínimo 10MB y preferiblemente ubicada entre los primeros 1024 cilindros del disco (la ubicación en los primeros 1024 cilindros es indispensable si su BIOS es de antes de 1998 o no soporta LBA32).
  - `/home` para cuentas de usuarios (se sugiere un espacio mínimo de 2MB por cada usuario del sistema).
  - `/var` por lo menos 100MB más al menos 500K por cada usuario (en esta partición se mantendrá la cola de correo electrónico).

- / con el resto del espacio del disco duro.

Computadores clientes

- Procesador >=486, Memoria: >=32MB.
- Tarjeta de video >=VGA, Monitor capaz de presentar gráficas 640x480x16.

- Disco Duro >=300MB por equipo dividido en particiones: @itemize @item `/boot` con mínimo 10MB y ubicado en los primeros 1024 cilindros del disco (la ubicación en los primeros 1024 cilindros es indispensable si su BIOS es de antes de 1998 o no soporta LBA32). el computador tiene una BIOS posterior a 1998 no es indispensable que `/boot` esté en los primeros 1024 cilindros).
  - `/` con mínimo 250MB, aun cuando de ser posible se sugiere 1.5GB.
  - `/swap` con la misma cantidad de memoria RAM o un poco más.
  - `/aux` con el espacio restante del disco duro.

- Teclado, ratón, unidad para disquete 3 1/2 (se sugiere unidad de CD-ROM al menos durante la instalación).

A.1.3. Software

- En cada cliente:
  - La distribución Debian 2.2.
  - GNOME >= 1.2 (si es posible XimianGNOME >= 1.2).
  - Locales deben estar instalados por lo menos los siguientes programas: servidor de X-Window, cliente y servidor `telnet`, cliente y servidor `ssh`, cliente y servidor `ftp`, cliente y servidor `finger`, cliente y servidor `talk`, cliente y servidor de `NFS`, cliente de `NIS`, servidor de correo `exim`, `bash`.

    Los paquetes son: adduser, ae, alsa-base, apt, at, base-config, base-files, base-passwd, bash, bc, biff, bin86, binutils, bison, bsdmainutils, bsdutils, bug-buddy, console-data, console-tools, console-tools-libs, cpio, cpp, cron, dc, debconf-tiny, debian-policy, debianutils, dialog, diff, dnsutils, doc-debian, doc-linux-text, dpkg, dpkg-dev, dpkg-ftp, dpkg-perl, e2fsprogs, ed, eject, elvis-tiny, emacs20, emacsen-common, enscript, esound, esound-common, exim, fbset, fdflush, fdutils, file, fileutils, findutils, finger, fingerd, flex, freetype2, ftp, g++, gcc, gcompris-data, gdb, gdk-imlib1, gdm, gettext-base, gmc, gnome-applets, gnome-bin, gnome-control-center, gnome-core, gnome-help, gnome-help-data, gnome-libs-data, gnome-panel, gnome-panel-data, gnome-session, gnome-terminal, gpm, grdb, grep, groff, gs, gsfonts, gzip, helix-sweetpill, hostname, iamerican, ibritish, imlib-base, imlib1, info, isapnptools, ispell, ldso, less, libart2, libasound0.4, libaudiofile0, libc6, libc6-dev, libcapplet0, libdb2, libdb2-util, libesd-alsa0, libgdbmg1, libgdk-pixbuf-gnome2, libgdk-pixbuf2, libghttp1, libglade-gnome0, libglade0, libglib1.2, libgmp2, libgnome32, libgnomesupport0, libgnomeui32, libgnorba27, libgnorbagtk0, libgtk1.2, libgtkxmhtml1, libgtop1, libident, libjpeg62, liblockfile1, libncurses4, libncurses5, libnet-perl, libnet-telnet-perl, libnewt0, libopenldap-runtime, libopenldap1, liborbit0, libpam-modules, libpam-runtime, libpam0g, libpanel-applet0, libpaperg, libpcre2, libpng2, libpopt0, libreadline4, librep9, libssl09, libstdc++2.10, libstdc++2.10-dev, libtiff3g, libungif3g, libwrap0, libxml1, libzvt2, lilo, locales, lockfile-progs, login, logrotate, lpr, lsof-2.2, lynx, m4, magicfilter, mailx, make, makedev, man-db, manpages, manpages-dev, mawk, mbr, mc-common, mime-support, minicom, modconf, modutils, mount, mpack, mtools, mutt, ncurses-base, ncurses-bin, ncurses-term, netbase, nfs-common, nfs-server, nis, nvi, passwd, patch, pciutils, perl-5.005, perl-5.005-base, perl-5.005-doc, perl-5.005-suid, perl-base, perl-tk, pidentd, ppp, pppconfig, procmail, procps, psmisc, pump, rcs, rep, rep-gtk, sawfish, sawfish-gnome, sed, setserial, sharutils, shellutils, slang1, ssh, ssh-askpass, ssh-askpass-gnome, ssh-askpass-ptk, strace, svgalibg1, sysklogd, syslinux, sysvinit, talk, talkd, tar, task-helix-core, tasksel, tcl8.2, tcpd, tcsh, tcsh-i18n, telnet, telnetd, tetex-base, tetex-bin, tetex-lib, texinfo, textutils, time, tk8.2, update, util-linux, vflib2, wenglish, whiptail, whois, xbase-clients, xf86setup, xfonts-100dpi, xfonts-base, xfree86-common, xlib6g, xpm4g, xserver-common, xserver-svga, xserver-vga16, xviddetect, zlib1g

  - Bien locales o compartidos de `/usr` del servidor, montado como `/opt` en los clientes: `gmc`,`man`, `gcalc`, `gnotepad`, `gnomecal`, `mtools`, `info`, `chfn`, `mesg`, `useradd`, `userdel`, `groupadd`, `groupdel`, `mail`, `ps`, `pstree`, `top`, `dmesg`, `last`, `w`, `who`, `grep`, `sed`, `awk`, `sort`, `cat`, `tac`, `head`, `tail`, `nl`, `wc`, `split`, `csplit`, `abiword`, `gimp`, `dia`, `emacs`, `gcc`, `make`, `gdb`, `xdvi`, `dvi2ps`, `gv`, `chown`, `chgrp`, `gnorpm`, `lynx`, `w3m`, `gtop` de ser posible un navegador gráfico y un cliente de correo gráfico.

- En el servidor deben estar los mismos programas de cada cliente y además:
  - El servidor Apache.
  - Toda la documentación posible en `/usr/share` (estas guías, Documentación de GNOME, HOWTOs, Network Administrator Guide, System Administrator Guide, archivos para `man` e `info`).
  - Servidor de `NIS`.
  - Herramientas para documentación con Texinfo: `makeinfo`, `texi2dvi`, `texi2html`.
  - Extensiones y programas para Emacs: `gnus` y opcionalmente `w3` que estarían compartidos en `/usr/share`.
  - Los paquetes incluyen: abiword, adduser, ae, alsa-base, amaya, anacron, apache, apache-common, apache-doc, apt, asigna, at, auctex, autoclass, autoconf, automake, balsa, base-config, base-files, base-passwd, bash, bc, bibindex, bibtool, biff, bin86, bind, bind-doc, binutils, bison, blt, blt-common, blt-dev, blt8.0, bootpc, bsdgames, bsdmainutils, bsdutils, bug-buddy, build-essential, bzip2, ccmalloc, codecommander, console-data, console-tools, console-tools-libs, cpio, cpp, cron, cvs, cvs-buildpackage, cygnus-stylesheets, dc, ddd, debconf, debhelper, debian-policy, debiandoc-sgml, debianutils, debmake, developers-reference, dh-make, dia, diald, dialdcost, dialog, diff, diffstat, diploma, dlint, dnsutils, doc-base, doc-debian, doc-linux-es, doc-linux-text, docbook, docbook-doc, docbook-stylesheets, docbook-stylesheets-doc, docbook-to-man, docbook-xml,

docbook2man, docbook2texi, dpkg, dpkg-dev, dpkg-ftp, dpkg-multicd, dpkg-perl, dupload, dvidvi, e2fsprogs, ecpg, ed, eeyes, eject, electric-fence, elvis-tiny, emacs20, emacsen-common, enscript, eog, esound-alsa, esound-common, exim, expect5.31, expectk5.31, f2c, fakeroot, fbset, fdflush, fdutils, fetchmail, fftw-dev, fftw2, file, fileutils, findutils, finger, fingerd, flex, fort77, fping, freetype2, ftnchek, ftp, g++, g77, gaim-gnome, gawk, gcc, gcc-doc, gcd, gdb, gdk-imlib1, gdm, gedit, gettext, gettext-base, gimp1.1, glade-gnome, glibc-doc, gmc, gnapster, gnome-applets, gnome-audio, gnome-bin, gnome-control-center, gnome-core, gnome-faq, gnome-gv, gnome-help, gnome-help-data, gnome-iconedit, gnome-libs-data, gnome-media, gnome-network, gnome-panel, gnome-panel-data, gnome-pim, gnome-pim-conduits, gnome-session, gnome-terminal, gnome-users-guide-en, gnome-utils, gnomeicu, gnosamba, gnotepad+, gnotepad+-help, gnumeric, gnuplot, gob, gobjc, gpm, gqview, grace, grdb, grep, groff, gs, gsfonts, gsfonts-x11, gtop, guile1.3, guitar, gv, gzip, helix-sweetpill, hostname, hyperlatex, iamerican, ibritish, icmpinfo, imlib-base, imlib1, indent, info, inn2, inn2-inews, ircd, ircd-dalnet, ircii, isapnptools, ispanish, ispell, jade, jadetex, junkbuster, kaffe, kernel-package, kernel-source-2.2.18pre21, lacheck, latex2html, ldso, less, lesstif1, lftp, libart2, libasound0.4, libaudiofile0, libbz2, libc6, libc6-dev, libcapplet0, libdb2, libdb2-util, libdigest-md5-perl, libesd0, libgal3, libgc5, libgc5-dev, libgdbmg1, libgdk-pixbuf-gnome2, libghttp1, libgimp1.1, libglade-gnome0, libglade0, libglib1.2, libglib1.2-dev, libgmp2, libgnome-pilot0, libgnome32, libgnomeprint-bin, libgnomeprint-data, libgnomeprint11, libgnomeprint6, libgnomesupport0, libgnomeui32, libgnorba27, libgnorbagtk0, libgpmg1, libgtk1, libgtk1.2, libgtk1.2-dev, libgtkxmhtml1, libgtop1, libguile6, libhtml-parser-perl, libi18n-langtags-perl, libident, libjpeg-progs, libjpeg62, liblockfile1, libltdl0, libltdl0-dev, libmime-base64-perl, libncurses4, libncurses5, libncurses5-dev, libnet-perl, libnet-telnet-perl, libnewt0, libnspr4, libobgnome0, libobgtk1, libogg0, libole2-0, libopenldap-runtime, libopenldap1, liborbit0, libpam-modules, libpam-runtime, libpam0g, libpanel-applet0, libpaperg, libpcap0, libpcre2, libpgsql2, libpgtcl, libpisock3, libpng2, libpopt0, libproplist0, libreadline4, librep9, librpm1, libruby, libsgmls-perl, libsp1, libssl09, libstdc++2.10, libstdc++2.10-dev, libtext-format-perl, libtiff3g, libtimedate-perl, libtool, libungif3g, libunicode0, liburi-perl, libvorbis0, libwrap0, libwraster1, libwww-perl, libxml1, libzvt2, lilo, lintian, locales, lockfile-progs, login, logrotate, lpr, lsof-2.2, lynx, m4, magicfilter, mailtools, mailx, maint-guide, make, makedev, makepasswd, man-db, manpages, manpages-dev, manpages-es, mawk, mbr, mc-common, memprof, menu, metrics, mime-support, minicom, modconf, modutils, mount, mozilla, mpack, mpage, mtools, ncurses-base, ncurses-bin, ncurses-term, netbase, netpbm, netpbm1, nfs-common, nfs-server, nis, nvi, ocaml-doc, octave, octave-ci, octave-doc, octave-htmldoc, packaging-manual, pan, passwd, patch, perl-5.005, perl-5.005-base, perl-5.005-doc, perl-5.005-suid, perl-base, pgaccess, pidentd, plotutils, pnmtopng, postgresql, postgresql-client, postgresql-contrib, postgresql-doc, postgresql-pl, postgresql-slink, ppp, pppconfig, procmail, procps, proftpd, psgml, psmisc, pstoedit, psutils, pump, python-base, python-doc, rcs, rdate, rdist, rep, rep-gtk, rpm, ruby, ruptime, rusers, rwall, rwho, rwhod, sawfish, sawfish-gnome, sed, setserial, sgml-base, sgml-data, sgmlspl, sgmltools-2, sharutils, shellutils, slang1, sp, ssh, stl-manual, strace, svgalibg1, swig, sysklogd, syslinux, sysvinit, t1lib1, talk, talkd, tar, task-c++-dev, task-c-dev, task-database-pg, task-devel-common, task-dns-server, task-fortran, task-gnome-apps, task-gnome-desktop, task-gnome-net, task-helix-core, task-objc-dev, task-python, task-science, task-sgml, task-spanish, task-tcltk, task-tcltk-dev, task-tex, task-x-window-system-core, tasksel, tcl8.0, tcl8.2, tcl8.2-dev, tclreadline, tcpd, tcpdump, telnet, telnetd, tetex-base, tetex-bin, tetex-doc, tetex-extra, tetex-lib, texinfo, textutils, tftp, time, tk8.0, tk8.2, tk8.2-dev, tktable, tktable-dev, traceroute, transfig, typist, ucblogo, untex, update, user-es, util-linux, vflib2, vile, vile-common, vile-filters, w3m, wenglish, wget, whiptail, whois, wmaker, wspanish, wvdial, wwwoffle, wzip, xaw3dg, xbase-clients, xchat-common, xchat-gnome, xcontrib, xf86setup, xfig, xfonts-100dpi, xfonts-75dpi, xfonts-base, xfonts-scalable, xfree86-common, xlib6g, xlib6g-dev, xpdf, xpm4g, xscreensaver, xserver-common, xserver-mono, xserver-svga, xviddetect, zip, zlib1g

A.1.4. Software Adicional

De acuerdo a las necesidades de su colegio, a los usuarios de la red y eventualmente al currículo, puede instalar software adicional. A continuación resumimos algunas de nuestras recomendaciones de acuerdo al tipo de usuario. La información más actualizada podrá consultarla en la versión más reciente de estas guías que está disponible en: http://structio.sourceforge.net/guias/AA_Linux_colegio

- Para Estudiantes:

Procesador de texto y hoja de cálculo livianos

: `abiword` y `gnumeric` o `pw` y `siag`. Los primeros hacen parte de Gnome Office: http://www.gnome.org/gnome-office/ (paquetes `gnumeric` y `abiword`) mientras los segundos de Siag Office: http://www.siag.nu/ (paquete `xsiag`). Si los computadores cliente tienen más de 64MB en RAM puede ser mejor opción OpenOffice que incluye filtros para leer y escribir en formatos de Microsoft Office: http://www.openoffice.org (no hay paquete para Debian 2.2, si para versiones posteriores, los binarios distribuidos por OpenOffice corren bien).

Graficadores

: Además de `gimp` (paquete `gimp`) se recomiendan `xfig` (paquete `xfig`) y `dia` (paquete `dia`). Tanto `gimp` como `dia` hacen parte de Gnome Office, el sitio de desarrollo de `xfig` es: http://www.xfig.org/

Mecanografía

: GTypist disponible en: http://www.gnu.org/software/gtypist

Lenguajes

- Logo: Una implementación es Berkeley Logo disponible en http://www.cs.berkeley.edu/~bh/logo.html
- C: `gcc`, `make` y `vim` o `emacs`.
- Java: paquete `jdk1.1`, hay versiones más recientes disponibles en el sitio de desarrollo: http://www.javasoft.com
- Tcl/Tk: paquetes `tcl8.2` y `tk8.2`. El sitio de desarrollo es http://www.scriptics.com
- Ocaml: El sitio de desarrollo es http://caml.inria.fr/ocaml

Bases de Datos

postgresql (paquete postgresql). Si prefiere una base de datos menos poderosa pero más liviana puede emplear MySQL (paquete mysql-server)

Matemáticas

Para estudiantes de Bachillerato Dr. Genius y Dr Geo. (paquete drgeo). Los sitios de desarrollo son: http://www.ofset.org/drgenius (http://www.ofset.org/drgenius) y http://www.ofset.org/drgeo (http://www.ofset.org/drgeo).

Para estudiantes de primaria que se acerquen a un computador

GCompris.http://www.ofset.org/gcompris (http://gcompris.sourceforge.net)

- Para Profesores

Herramientas de Productividad

Si tiene computadores modernos cada uno con más de 64MB en RAM se recomienda OpenOffice: http://www.openoffice.org En otro caso debe emplear herramientas más livianas como abiword, gnumeric o pw y siag.

Gradebook

para mantener planilla personal de calificaciones: http://www.gnu.org/software/ggradebook/

- Para Secretarias y personal administrativo

Horarios

Para ayudar a asignar el horario de un colegio puede emplear el programa asigna –mantenido y desarrollado por miembros de Structio–. http://structio.sourceforge.net/asigna

Herramientas de Productividad

Vea el comentario sobre herramientas de productividad para profesores.

A.1.5. Usuarios y Grupos

Grupos

estudiantes

Sugerido GID 1101

profesores

Sugerido GID 1102

administración

Sugerido GID 1103

Usuarios

Usuarios: Identificaciones superiores a 1000 (siguiendo política de Debian).

Apéndice B. Sobre estas guías

El desarrollo de las guías al igual que el de la plataforma de referencia S-Helio 1.1, comenzó identificando en un colegio concreto grupos de usuarios y sus necesidades en materia de software (Estudiantes, Profesores, Personal Administrativo y Administradores de red). Después se identificó software de libre redistribución para satisfacer sus necesidades y se establecieron las habilidades que los usuarios debían desarrollar para emplearlo. Tales habilidades se clasificaron en 6 grupos, cada uno de los cuales corresponde a un grupo de guías y a un capítulo de este documento.

Cada grupo de guías tiene unos logros por alcanzar y un público objetivo al interior del colegio. Hay grupos de guías para diversos niveles de usuarios, comenzando por usuarios que tendrán su primera experiencia con Linux y terminando con guías para administradores de la plataforma de referencia S-Helio 1.1. Por su parte cada guía se compone de:

- Indicadores de Logro y/o una breve introducción.
- Una lectura que busca dar información completa pero corta sobre el tema de la guía.
- Referencias a lecturas concisas sobre el tema de la guía, disponibles en su computador con Debian o en Internet.
- Ejercicios de diversos tipos (consulta, discusión, práctica).

El ambiente de aprendizaje en el cual se ha pensado al desarrollar las guías, es un colegio dotado de una red que cumpla con los lineamientos de nuestra plataforma de referencia S-Helio 1.1 [1]. Las guías para usuarios (capítulos 1, 2 y 3) presuponen que los lectores emplean la plataforma de referencia S-Helio 1.1 y que pueden leer español. Las guías para administradores (capítulos 4, 5 y 6) además presuponen que los administradores tienen clave de root de una máquina Linux, cuentan con acceso a Internet, pueden leer en inglés y pueden comunicarse en español.

En cuanto a la dinámica de una clase apoyada en estas guías hemos supuesto que las lecturas junto con algunos ejercicios se realizarán en casa mientras que los ejercicios prácticos se desarrollarán en el colegio con ayuda de un coordinador quien debe conocer previamente estas guías y la plataforma de referencia S-Helio 1.1. [2].

Finalmente queremos resaltar que estas guías sólo cubren el uso básico de la plataforma de referencia S-Helio 1.1. Cada colegio, tras instalar la plataforma de referencia debe instalar software de acuerdo a sus necesidades y escribir documentación apropiada para los usuarios. En el sitio web http://structio.sourceforge.net/guias podrá encontrar documentación para ciertos programas útiles en colegios que funcionan sobre la plataforma de referencia S-Helio 1.1; además encontrará instrucciones para aportar las guías que usted haga, las cuales pueden ser de utilidad para otras personas y colegios.

1. Si su institución no cuenta con la plataforma de referencia S-Helio 1.1, podrá encontrar instrucciones para instalarla en los capítulos 5 y 6. Así como soporte de voluntarios en la lista `<structio-sopcol@lists.sourceforge.net>`
2. Aún cuando hemos supuesto un ambiente de aprendizaje y una plataforma de referencia básica, partes de estas guías también podrían ser empleadas con precondiciones algo diferentes enmarcadas en el uso de Linux o Unix

Apéndice C. Actualización del *kernel* de Debian Linux

Pablo Chamorro

<pchamorro@ingeomin.gov.co>

El *kernel* o núcleo es una de las partes esenciales de un sistema operativo. El *kernel* proporciona todos los servicios básicos requeridos por otras partes del sistema operativo y aplicaciones, y en complemento, se encarga de la administración de la memoria, los procesos y los discos. El *kernel* es independiente de la distribución Linux utilizada, de tal forma que el mismo *kernel* debería servir en cualquier caso. La actualización del *kernel* se divide en 2 partes: compilación e instalación del nuevo *kernel*.

C.1. Introducción

C.1.1. Acerca de esta guía

En esta guía se desarrolla el ejemplo de la compilación del *kernel* versión 2.2.18 bajo Debian 2.1 Citius (*kernel* 2.0.36). Si desea compilar un *kernel* 2.4.x bajo un sistema con *kernel* 2.0.x o 2.2.x deberá consultar otros documentos que tratan específicamente sobre el tema.

De cualquier forma al final de este instructivo se presentan algunas referencias de documentos disponibles en Internet para mayor ilustración.

C.1.2. ¿Por qué compilar?

Hay diversas razones para querer reconstruir el *kernel* de Linux:

Es útil recompilar un *kernel* porque trabajando con *kernel*s nuevos generalmente se obtiene:

1. Un sistema más rápido, estable y robusto.
2. Un sistema con soporte a elementos de hardware no encontrado en *kernel*s viejos.
3. Un sistema con soporte a características especiales disponibles pero no habilitadas en *kernel*s viejos.

Una alternativa a recompilar el *kernel* es instalar una nueva versión de la distribución Linux que esté utilizando. Dependiendo de las circunstancias puede resultar conveniente o preferible lo uno o lo otro. Por ejemplo, si tiene un sistema Linux que le costó esfuerzo y tiempo para instalar y configurar, y no desea volver a realizar todo ese trabajo otra vez, es posible que recompilar el *kernel* sea lo más sencillo para dar solución a su necesidad particular.

Recompilar el *kernel* de Linux no es más que personalizar el *kernel* y como con cualquier aplicación, la personalización se hace para sacar un mayor provecho de las diferentes características que ofrece el software.

Si usted no tiene experiencia compilando el *kernel* es recomendable que utilice un equipo donde usted pueda practicar sin temor de que su sistema eventualmente vaya a quedar estropeado. No hay ningún riesgo al compilar el *kernel* (salvo que se quede sin espacio), en cambio, la instalación del mismo debe de realizarse con algo de cuidado.

C.1.3. Versiones de producción y de desarrollo.

Los *kernel*s pueden corresponder a versiones de "producción" o a versiones de "desarrollo". Las versiones 2.0.x, 2.2.x y 2.4.x son versiones de producción, por el contrario, las versiones 2.1.x y 2.3.x son versiones de desarrollo. En forma particular se tiene que, por ejemplo, la versión 2.2.17 o la versión 2.2.18 son versiones de producción, estables y para usuarios finales y las versiones 2.1.1 o 2.1.2 corresponden a versiones de desarrollo, inestables, utilizadas por desarrolladores o por ciertos usuarios que desean probar.

No utilice *kernel*s de desarrollo a menos que usted sepa lo que está haciendo.

C.1.4. Acerca de los módulos

Varias partes de código del *kernel* pueden compilarse por separado en forma de módulos, dando flexibilidad al sistema. Los módulos no están enlazados directamente en el *kernel*, siendo necesario insertarlos en él ya sea durante el proceso de arranque o si se quiere, posteriormente, de tal forma que sólo se usan cuando se necesitan, sin utilizar innecesariamente la memoria RAM del sistema.

Hay componentes que no deben compilarse como módulos porque son esenciales para hacer posible el arranque del sistema, por ejemplo el soporte para el sistema de archivos ext2, que Linux utiliza en forma predeterminada. Otros componentes si pueden hacerlo, por ejemplo, el soporte para una cierta tarjeta de red.

C.1.5. Software requerido

Los requisitos de software necesario para compilar el *kernel* varian considerablemente de una versión mayor del *kernel* a otra, por ejemplo para compilar un *kernel* 2.4.x bajo un sistema con *kernel* 2.2.x. Si es esa su necesidad, es mejor que consulte la nota respecto a la instalación de *kernel*s 2.4.x, en la sección de Introducción de este documento.

Para compilar un nuevo *kernel* correspondiente a una misma versión mayor del *kernel*, como es el caso del ejemplo escogido (instalar *kernel* 2.2.18 bajo Red Hat 6.1/*kernel* 2.2.12), seguramente no necesitará software adicional al incluido en la distribución Linux que esté empleando.

Para compilar un nuevo *kernel* generalmente no necesitará software adicional al incluido en la distribución Linux que esté empleando, a menos que se trate de un *kernel* 2.4.x, caso en el cual es posible que si se lo requiera.

En cualquier caso es recomendable consultar primero el archivo `README` principal incluido en los fuentes del *kernel* a instalar bajo el directorio base linux.

> En particular, para el ejercicio planteado en este documento basado en Debian 2.1 Citius, fue necesario instalar, adicionalmente al software base, el paquete bin86 para lograr el objetivo.

C.1.6. Tiempo de compilación

El tiempo necesario para compilar es muy variable, dependiendo del computador que se vaya a utilizar. En computadores recientes, por ejemplo en equipos Pentium Celeron o superiores, el proceso toma 10 minutos o menos. Tenga en cuenta que si utiliza computadores poco recientes, por ejemplo en equipos Pentium viejos, 486s y 386s el proceso puede tardar incluso varias horas.

> El tiempo de compilación puede reducirse a la mitad o menos, si primero se optimiza el disco duro donde se encuentra instalado Linux, utilizando el comando hdparm. Las mejoras en velocidad de lectura/escritura en discos será mayor cuanto más nuevos sean los discos. La optimización de un disco debe realizarse con cuidado ya que existe algún riesgo de malograr los archivos contenidos en el disco. Al final se presentan algunas referencias al respecto. Una vez se consigue optimizar un disco, se configura el sistema operativo para trabajo en modo optimizado en forma permanente.

C.1.7. Espacio en disco requerido

El espacio necesario para compilar el *kernel* varía de acuerdo a la versión del *kernel* a utilizar. Los fuentes del *kernel* 2.2.18 ocupan cerca de 92 MB, así que para este caso tomado de ejemplo sería bueno disponer de al menos unos 100 MB libres.

C.2. Compilación del *kernel*

Estrictamente hablando, para compilar el *kernel* no se necesita ser root ya que se puede realizar desde el directorio de un usuario, siempre y cuando se disponga de suficiente espacio en disco y de las herramientas de compilación instaladas. En este instructivo se utiliza la cuenta del root para efectos prácticos.

C.2.1. Obtención y descompresión de los fuentes

Obtenga una copia del código fuentes del *kernel* de http://www.kernel.org o de alguno de los servidores alternos listados en http://www.kernel.org/pub/mirrors/ . Para el ejemplo, para el *kernel* versión 2.2.18 el archivo a obtener es ya sea `linux-2.2.18.tar.bz2` (empaquetado con tar y comprimido con bzip2) o `linux-2.2.18.tgz` (empaquetado con tar y comprimido con gzip). Los archivos comprimidos con bzip2 normalmente son de menor tamaño.

C.2.2. Descompresión de los fuentes

Antes de proceder a su descompresión, verifique que disponga de 70 o más MBytes libres en el disco duro, por ejemplo con el comando df:

```
[root@linux]# df
```

Normalmente el directorio donde se descomprimen los fuentes del *kernel* es `/usr/src`. Los archivos fuentes del *kernel* crean una estructura de directorios a partir de `/usr/src/linux`, lo que puede resultar inconveniente si

1. ya existe un vínculo o un directorio con ese nombre
2. al desempaquetar se sobreescriben archivos.

Luego de seleccionar el directorio de trabajo proceda a descomprimir de la siguiente forma:

1. Si decidió trabajar con el archivo `linux-2.2.18.tar.bz2`, desempaquete y descomprima utilizando mediante:

   ```
   [root@linux]# bzcat linux-2.2.18.tar.bz2| tar -xvf -
   ```

   Alternativamente puede ejecutar:

   ```
   [root]# tar --use-compress-program bunzip2 -xvf linux-2.2.18.tar.bz2
   ```

2. Si optó por usar el archivo `linux-2.2.18.tgz`, el comando a utilizar es:

   ```
   [root]# tar -zxvf linux-2.2.18.tgz
   ```

C.2.3. Creación de un archivo de configuración

La definición de la configuración del nuevo *kernel* es el punto crucial de la compilación. Una buena configuración se alcanza sólamente:

1. Si tiene a la mano todos los datos del hardware que el *kernel* va a soportar: puertos, discos duros, tarjetas, etc.
2. Si define qué características ofrecidas van a ser soportadas por el nuevo *kernel*, ya sea directamente dentro del núcleo o como módulos.

Las alternativas para configurar el nuevo *kernel* corresponden a los comandos:

- **make config**,
- **make menuconfig**, o
- **make xconfig**

El comando **make config** ofrece un interfaz primitivo basado en modo texto, sin ayuda ni menús. Los comandos **make menuconfig** y **make xconfig** conducen a mejores interfaces de usuario, con menús y quizá lo más importante, con una ayuda contextual que facilita en gran medida la configuración.

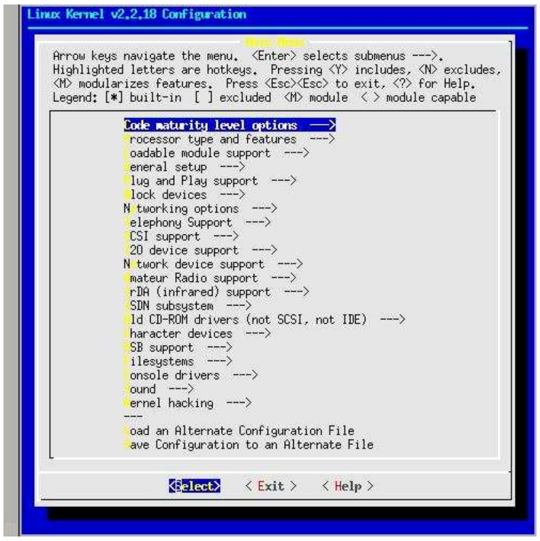
Para el ejemplo se utiliza **make menuconfig**, porque ofrece un interfaz amigable al usuario al igual que **make xconfig**, pero resulta más práctica si no se dispone o no se quiere utilizar X Windows. Antes de proceder a ejecutar este comando, cámbiese al directorio apropiado:

```
[root]# cd linux
```

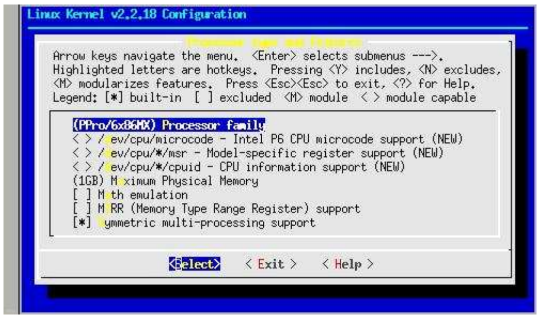
Ahora si ejecute:

```
[root]# make menuconfig
```

Las diferentes opciones disponibles a partir del menú inicial se presentan en forma ampliada en la siguiente figura:



Un ejemplo de uno de los submenús expandido (Processor type and features) se aprecia enseguida:



En esta pantalla se puede apreciar por ejemplo que en forma predeterminada se ha desactivado la emulación matemática (Math emulation) y que si se desea habilitar ésta quedaría incluida necesariamente dentro del *kernel* y no como un módulo (como lo indican los parentesis rectangulares de esa opción), contrario a las opciones que están acompañadas por paréntesis angulares (<>), que pueden compilarse o no como módulos si se desea.

Continuando con el ejemplo, si se ingresa al sub-menú Processor Family, se aprecian las siguientes opciones:

```
+----------------- Processor family -----------------+
|  Use the arrow keys to navigate this window or press the hotkey of |
|  the item you wish to select followed by the <SPACE BAR>. Press    |
|  <?> for additional information about this option.                 |
+--------------------------------------------+
| |                    ( ) 386                                   | |
| |                    ( ) 486/Cx486                             | |
| |                    ( ) 586/K5/5x86/6x86                      | |
| |                    ( ) Pentium/K6/TSC                        | |
| |                    (X) PPro/6x86MX                           | |
+--------------------------------------------+
|                    <Select>      < Help >                      |
+--------------------------------------------+
```

En definitiva el procedimiento a seguir es entonces:

1. Con la ayuda de las teclas de movimiento del cursor y la tecla [**Enter**] recorra todas y cada una de las opciones de los submenús disponibles a partir del menú principal,
2. Con la ayuda de la barra espaciadora defina si habilita o no el soporte para una u otra característica, y si lo hace en forma integrada al *kernel* o en forma modular si desea y es posible.

   Al oprimir la barra espaciadora una o varias veces según se necesite, es posible que visualice:

   - Un nuevo sub-menú.
   - Un asterisco (en ocasiones acompañado de sub-opciones), indicando que se va a compilar integrado al *kernel*.
   - La letra **M** (en ocasiones acompañada de sub-opciones), indicando que se va a compilar en forma modular.

- La letra **X**, similar al asterisco, pero sólo para opciones de selección múltiple con única alternativa.
- Un espacio en blanco, indicando que no se va a incluir dentro del *kernel* el soporte para un determinado item.

3. Cuando haya llevado a cabo por completo los puntos 1 y 2, desde el menú principal seleccione la opción exit y responda con **Y** (de *Yes*) para grabar la configuración resultante en el archivo `.config`, de lo contrario tendrá que volver a realizar todo el proceso otra vez.

C.2.4. Una guía de apoyo para definir un archivo de configuración

La ayuda contextual de disponible con **make menuconfig** o con **make xconfig** es quizás lo suficientemente apropiada para facilitar el proceso de definición del archivo de configuración, no obstante aquí se presenta alguna ilustración adicional.

Code Maturity Level Options

Esta opción viene desactivada en forma predeterminada, deshabilitando el uso de código experimental en el *kernel*.

Processor Type and Features

Bajo esta sección es posible definir entre otros:

- Tipo de procesador de su computador. Para optimizar el *kernel* para su procesador utilice la ayuda contextual para mayor ilustración.
- Emulación del procesador matemático (útil con equipos 386 y similares que vienen sin él). Deshabilitada en forma predeterminada.
- Soporte para sistemas con más de 1 procesador (SMP). Deshabilite este soporte si va a trabajar en un computador que solamente dispone de 1 procesador.

Es posible que el *kernel* no arranque si usted selecciona un procesador incorrecto, por ejemplo si optimiza el *kernel* para un procesador Pentium y usted tiene un procesador 486.

Para el ejemplo, el cambio principal realizado fue deshabilitar el soporte para SMP.

Loadable Module Support

Si por alguna razón desea generar un *kernel* monolítico, sin módulos, desactive esta opción. Lo normal es por el contrario, no realizar ningún cambio, aceptando la opción predeterminada.

General Setup

Esta sección incluye entre otros, soporte para:

- Redes.

- Bus PCI.
- Puerto paralelo
- Administración Avanzada de Energía (APM).

Para el ejemplo, el cambio principal realizado fue habilitar el soporte para el puerto paralelo y para APM, deshabilitados en forma predeterminada.

Plug and Play

Habilite esta opción si desea que el *kernel* automáticamente configure algunos dispositivos periféricos.

Block Devices

Bajo esta sección se presenta soporte para:

- Unidades de discos flexibles.
- Soporte mejorado para dispositivos de almacenamiento estándar IDE y
- Otros dispositivos de datos en modo bloques.

Si se dispone de un disco duro IDE, será necesario habilitar el soporte para el disco IDE apropiado.

Para el ejemplo, se aceptaron los valores predeterminados propuestos sin realizar ningún cambio.

Networking Options

Esta sección es de vital importancia para equipos que van a operar en red. Las opciones que en forma predeterminada vienen habilitadas son:

- Packet socket
- Unix domain socket
- TCP/IP networking, e
- IP: Allow large windows

Para el ejemplo, se habilitaron adicionalmente las opciones:

- Network firewalls, para permitir que Linux actúe como un equipo "corta fuegos" con posibilidad de filtrar paquetes y protegerse de accesos no autorizados de Internet, e
- IP firewalling, para poder acceder a equipos remotos (por ejemplo de Internet) utilizando enmascaramiento a través de un equipo "corta fuegos", de tal forma que los equipos de la red local no son visibles al mundo exterior, evitando la necesidad de direcciones IP válidas.

En la práctica, estas dos opciones permiten por ejemplo, poder utilizar en forma compartida para una red local una conexión a Internet a través de una sola línea telefónica, disponiendo de una seguridad básica.

Adicionalmente, entre otras están las opciones que permiten a Linux actuar como un enrutador o dar soporte a protocolos tales como IPX o *Reverse* ARP.

Telephony Support

Esta opción, que permite dar soporte a correo de voz, fax y otros dispositivos multimedia que inteoperan con modems, viene deshabilitada en forma predeterminada.

SCSI Support

En forma predeterminada esta opción viene habilitada, pero si usted no tiene o no desea usar un disco, unidad de cinta, unidad de CD-ROM o cualquier otra unidad tipo SCSI, desactívela.

I2O Device Support

Esta opción que hace referencia a la *Intelligent Input/Output (I20) architecture*, viene deshabilitada en forma predeterminada.

Network Device Support

Entre las diferentes alternativas disponibles en esta sección se destacan:

- Ethernet 10/100
- PPP (point-to-point) support

Respecto a la primera opción, al seleccionarla aparece un listado de tarjetas de red disponibles, incluyendo entre otros, tarjetas 3COM, tarjetas EISA, VESA, PCI y controladores integrados en las tarjetas principales, y otras tarjetas ISA. Por otro lado, el soporte para protocolo PPP es necesario para permitir conexión a Internet por vía telefónica.

Para el ejemplo, se habilitó soporte integrado al *kernel* para la tarjeta de red disponible y también pero en forma modular para otras tarjetas de red de las cuales eventualmente se pueda disponer. En complemento, se habilitó el soporte para protocolo PPP, desactivado en forma predeterminada.

Amateur Radio Support

Esta sección permite soporte para transmisión de datos para radio aficionados. En forma predeterminada esta opción está deshabilitada.

IrDA Support

Esta sección permite soporte para trabajar con protocolos IrDA™, relacionados con comunicaciones inalámbricas infrarojas, soportadas por la mayoría de computadores portátiles (*laptops*) y PDA’s.

ISDN Subsystem

ISDN o RDSI como se conoce en Colombia, es un tipo especial de servicio telefónico completamente digital, disponible únicamente en las principales ciudades. Una línea RDSI permite una mejor conexión a Internet que a través de una línea telefónica convencional. Habilite esta opción si tiene interés y en su ciudad cuenta con proveedores de enlaces RDSI.

Old CD-ROM Drivers (not SCSI, not IDE)

Habilite esta opción si usted tiene una unidad de CD-ROM que no sea ni IDE/ATAPI ni SCSI, es decir, si tiene una unidad de CD-ROM antigua que se conecta a la tarjeta de sonido.

Character Devices

Bajo esta sección se define el soporte para dispositivos de caracteres, entre otros:

- Terminales
- Puertos seriales
- Ratones
- Palancas de juegos (soporte deshabilitado por omisión)

Para el ejemplo, se aceptaron los valores predeterminados.

USB Support

Habilite esta opción si su computador tiene puerto USB.

File Systems

Esta sección determina el tipo de sistemas de archivos a utilizar. Entre las opciones habilitadas en forma predeterminada se destacan entre otros:

ISO 9960 CDROM filesystem support

Para poder acceder a CDROM estándares.

/proc filesystem

Sistema de archivos virtual que proporciona información sobre el sistema operativo.

Second extended fs support

El sistema de archivos estándar de Linux

NFS filesystem support

Para interactuar con otros equipos UNIX utilizando el protocolo NFS.

Adicionalmente a estos sistemas de archivos podría resultar interesante habilitar el soporte para:

Quota support

Para poder colocar límites al uso del disco.

DOS FAT fs support

Para poder trabajar con sistemas de archivos de MS-DOS (*MSDOS fs support*) y de Windows 95 (*VFAT fs support*)

Microsoft Joliet CDROM extensions

Para poder leer CDROM en ese formato

SMB filesystem support

para poder montar sistemas de archivos de Windows for Workgroups (WfW), Windows 95/98, Windows NT y OS/2 Lan Manager. Si lo que se necesita es solamente acceder a un servidor Linux utilizando Samba no se necesita habilitar esta opción.

Finalmente, parece interesante también habilitar soporte para el Español como idioma nativo preferido al momento de montar sistemas de archivos, definiendo como predeterminado a NSL ISO 8859-1 (Latin 1).

Console Drivers

Bajo esta sección se presentan dos opciones:

- VGA text mode
- Video mode seleccion support

La primera de ellas, habilitada en forma predeterminada, permite utilizar Linux en modo texto en una pantalla que cumpla con el estándar VGA genérico. La segunda opción, deshabilitada en forma predeterminada, permite aprovechar algunas altas resoluciones en modo texto que podría ofrecer el BIOS.

Sound

Habilite esta opción si su computador tiene una tarjeta de sonido. Consulte la ayuda disponible.

Kernel Hacking

Esta opción, principalmente útil para *hackers*, permite tener cierto control sobre el sistema aún si el sistema se cae haciendo uso de la tecla de Peticiones del Sistema (SysRQ).

C.2.5. Compilación

Como se mencionó, no se necesita ser *root* para compilar el *kernel*. En este ejemplo se muestran los comandos a nombre del *root*.

1. Para compilar el *kernel* ejecute primero:

   ```
   [root]# make dep
   ```

   O mejor, para poder observar de mejor forma la salida y eventuales errores, ejecute:

   ```
   [root]# make dep 1> dep-salida.txt 2>dep-errores.txt
   ```

2. Ahora compile el *kernel* mediante:

   ```
   [root]# make bzImage
   ```

   o mejor, por ejemplo mediante:

   ```
   [root]# make bzImage 1> bzImage-salida.txt 2>bzImage-errores.txt
   ```

   Es necesario mencionar que **make bzImage** no es la única opción. Consulte la documentación disponible bajo el directorio `linux/Documentation` para obtener alguna información al respecto.

Al ejecutar `bzImage` se generan entre otros, el *kernel* `linux/arch/i386/boot/bzImage` y el archivo `linux/System.map`, para instalar posteriormente bajo el directorio `/boot`.

3. Si se habilitaron los módulos, ejecute:

   ```
   [root]# make modules
   ```

   o si prefiere:

   ```
   [root]# make modules 1> modules-salida.txt 2>modules-errores.txt
   ```

C.3. Instalación del nuevo *kernel*

Por el contrario, la instalación del nuevo *kernel* es tarea que sólo el root puede realizar.

C.3.1. Instalación de los archivos del *kernel*

1. Para comenzar de un vistazo al contenido del directorio `/boot` y tome atenta nota de ello. Note que algunos archivos son vínculos simbólicos.

   ```
   [root]# ls -l /boot
   ```

   Algunos de los archivos allí contenidos son:

   ```
   -rw-r-r-  1 root    root        4540 Feb 2 1999 boot.b
   -rw-r-r-  1 root    root        4540 Feb 2 1999 boot.b.preserved
   -rw-r-r-  1 root    root         612 Feb 2 1999 chain.b
   -rw-r-r-  1 root    root         612 Feb 2 1999 chain.b.preserved
   -rw-----  1 root    root       17408 Apr 9 00:45 map
   -rw-r-r-  1 root    root         444 Dec 12 1997 mbr.b
   -rw-r-r-  1 root    root         620 Feb 2 1999 os2_d.b
   -rw-r-r-  1 root    root         620 Feb 2 1999 os2_d.b.preserved
   -rwxr-xr-x  1 root    root       715259 Apr 8 17:56 vmlinuz-2.0.36
   ```

2. Mire el contenido del archivo `/etc/lilo.conf`, dando como un hecho que se está utilizando LILO:

   ```
   [root #] cat /etc/lilo.conf
   ```

   Por ejemplo, ese archivo podría contener:

   ```
   boot=/dev/hda1
   root=/dev/hda1
   install=/boot/boot.b
   map=/boot/map
   vga=normal
   delay=20
   image=/vmlinuz
    label=Linux
    read-only
   ```

3. Desde el directorio linux utilizado, proceda a copiar el archivo `bzImage` generado durante la compilación dentro del directorio `/boot`. Por ejemplo, de esta forma:

   ```
   [root]# cp -i arch/i386/boot/bzImage/boot/vmlinuz-2.2.18
   ```

4. Desde el directorio linux utilizado, proceda a hacer lo propio con el archivo `System.map`. Por ejemplo, de esta forma:

   ```
   [root]# cp -i System.map /boot/System.map-2.2.18
   ```

5. Instalación de los módulos del *kernel*

   Ahora proceda a instalar los módulos. Si está instalando un *kernel* de la misma versión del que ya tiene en el sistema, ejecute primero, por ejemplo:

   ```
   [root]# mv /lib/modules/2.2.12-20 /lib/modules/2.2.12-20.old
   ```

   y luego si:

   ```
   [root]# make modules_install
   ```

6. Edición del archivo `/etc/lilo.conf` e instalación de un nuevo MBR

   Edite el archivo `/etc/lilo.conf` y adicione una nueva sección para el nuevo *kernel*, para el ejemplo, así:

   ```
   boot=/dev/hda1
   root=/dev/hda1
   install=/boot/boot.b
   map=/boot/map
   vga=normal
   default=nuevo
   delay=20
   image=/vmlinuz
    label=Linux
    read-only
   image=/boot/vmlinuz-2.2.18
        label=nuevo
        read-only
   ```

   Note el cambio realizado en la línea `default`.

7. Ubíquese dentro del directorio `/boot`:

   ```
   [root]# cd /boot
   ```

8. Pruebe sin llevar a cabo la instalación del nuevo MBR ejecutando:

   ```
   [root]# lilo -t -v
   ```

9. Si no observó nada anormal proceda a la instalación del nuevo MBR ejecutando:

   ```
   [root]# lilo -v
   ```

```
debian:/usr/src/linux# lilo -v
     LILO version 21, Copyright 1992-1998 Werner Almesberger
Reading boot sector from /dev/hda1
Merging with /boot/boot.b
Boot image: /vmlinuz
Added Linux
Boot image: /boot/vmlinuz-2.2.18
Added nuevo *
/boot/boot.0301 exists - no backup copy made.
Writing boot sector.
```

10. Reinicie el sistema

    Si en un dado caso tuviera problemas con el nuevo *kernel* puede arrancar utilizando el *kernel* anterior, escribiendo linux frente al indicador de lilo, cuando apenas se inicie el arranque.

    Al reiniciar, el comando **uname -a** mostrará algo similar a:

    ```
    Linux debian 2.2.18 #2 Tue Apr 9 00:32:31 COT 2002 i586 unknown
    ```

C.4. Referencias

- The Linux Kernel HOWTO: http://linuxdoc.org/HOWTO/Kernel-HOWTO.html
- New Linux Kernel Introduced: http://www.linux.com/live/newsitem.phtml?sid=1&aid=11762 (http://www.linux.com/live/newsitem.phtml?sid=1&aid=11762)
- Kernel Upgrade: Know Your Hardware: http://www.linux.com/hardware/newsitem.phtml?sid=26&aid=11532 (http://www.linux.com/hardware/newsitem.phtml?sid=26&aid=11532)
- Configuring The Kernel For Your Needs: http://www.byte.com/column/BYT20000829S0006
- Compiling the Linux kernel: http://freeos.com/articles/2589/2/1-3/
- Rebuilding the Linux Kernel for Complete Newbies:http://www.gnulinux.com/help/kernel_recompile.shtml
- Compiling the Kernel: Part 2: http://www.linux.com/newsitem.phtml?sid=60&aid=8931 (http://www.linux.com/newsitem.phtml?sid=60&aid=8931), http://www.linux.com/newsitem.phtml?sid=60&aid=8931
- Compiling Linux 2.2 Kernel: 5594 (http://www.zdnet.com/zdhelp/stories/main/0)5594 (http://www.zdnet.com/zdhelp/stories/main/0)

Apéndice D. Respuestas a los ejercicios

1.1

1. Depende la licencia, algunas licencias permiten la venta de copias otras no.
2. Permiten libre redistribución y libre uso: Dominio público, BSD, MIT License, GPL, LGPL.
3. Personal

1.2

1. Los *BSD por ser menos populares tiene menos aplicaciones y cuentan con una capa para emular Linux; los desarrolladores se concentrar en portar aplicaciones de Linux y de otras plataformas; el desarrollo es dirigido por un equipo empleando CVS.
2. Fuentes abiertas busca presentar los méritos técnicos de usar programas cuyas fuentes puedan ser vistas por el público. Es impulsado por Open Source Initiative. Software libre se refiere a una ideología y una clasificación de licencias acorde. Se trata de la ideología de la Free Software Foundation.

1.3

1. Una vez sabe login y clave prende un computador, espere hasta que se inicie el modo gráfico y un recuadro en el que diga Login y usted pueda teclear su login y **ENTER** después verá otro recuadro en el que se le pide la clave –no la verá al teclear cuando termine presione **ENTER**.
2. Ctrl-Alt-F1 ; Ctrl-Alt-F7
3. Ctrl-Alt-F1 ; login y clave ; passwd ; clave anterior ; clave nueva dos veces ; exit ; Ctrl-Alt-F7

1.4

1. Menu; gedit; escribir; marcar presionando botón izquierdo y arrastrando; pegar ubicando el cursor y presionando botón del centro.
2. Menu; centro de control Gnome; papel tapiz
3. Menú; centro de control Gnome; tema
4. Menú; centro de control Gnome; gestor de ventanas.
5. Menu. Abrir y explorar cada aplicación para después opinar
6. Algunas especialmente utiles son: calendario y time tracker tool.

1.5

1. desplazarse; copiar; renombrar; borrar; mover; buscar; ordenar
2. **mkdir graficas**
3. **cp /usr/share/pixmaps/una.jpg graficas**
4. Sobre el directorio graficas presionar **del**
5. Sobre el directorio `/usr/share/pixmaps` presionar **del**

1.6

1. Puede preguntar al administrador o eventualmente revisar `/etc/fstab`, revise el tipo de disquete (puede ser `auto`, `ext2` o `vfat` junto al dispositivo del disquete (normalmente `/dev/fd0`).
2.
   ```
   mount /floppy
   cp /usr/share/pixmaps/uno.jpg /floppy
   ```
3. umount /floppy
4. Para copiar el archivo `~/carta.txt` a la primera unidad de disquetel **mcopy ~/carta.txt a:**. Para eliminar `doc.txt` se usa **mdel a:doc.txt**, para crear un directorio **mmd a:midir**; y para formatear el disquete: **mformat a:**

1.7

1. whatis presenta una breve descripción del comando que recibe, extractada de la correspondiente página del manual de Unix.
2. Un applet es un programa que se mantiene activo durante una sesión Gnome, presentando información en un pequeño espacio del panel.
3. Espacio para avanzar una pantalla o de un nodo a otro; q para salir; h para ver un tutorial; n para avanzar al siguiente nodo; p para ir al nodo anterior; u para ir al nodo superior; TAB pasa de un enlace a otro; Enter va a un enlace; l retorna al nodo antes visitado.
4. Para mover un archivo de un directorio a otro o para renombrarlo. Por ejemplo para mover `tmp/p.txt` a `casa/p.txt` se usa:
   ```
   mv tmp/p.txt casa/p.txt
   ```
5. Por cada programa hay un directorio que contiene al menos la licencia. Puede haber otros archivos que documentan el programa o el paquete Debian y pueden haber manuales de usuario.
6. Entre los HOWTO para principiantes uno muy apropiado es Unix-and-Internet-Fundamentals-HOWTO.
7. Colecciona y organiza documentos relacionados con Linux.

2.1

1. `~/.bashrc`
2.
   ```
   cd /usr/share/pixmaps
   ls *.xmp
   cp arc.xmp ~
   ```
3. cd ~ ; chmod og-x .
4. cd /home ; ls -l ; chmod -w pepe

2.2

1. Basta teclear **chfn** y cambiar los datos.
2. Editarlos con algún editor, e.g **vi ~/.plan**
3. **finger *usuario*@localhost**
4. **finger root@localhost**
5. **finger @localhost**
6. **finger quake@geophys.washington.edu** produce información sobre terremotos en Washington y Oregon.
7. Abrir las dos terminales, revisar TTY empleada por cada una, después desde la termina 1 usar **talk *usuario*@localhost *tty-terminal-2***
8. **mesg n** después repetir ejercicio anterior y después **mesg y**
9. **finger purpura.micolegio.edu.co talk davpin@purpura.micolegio.edu.co**
10. **mail *usuario*@localhost**

2.3

1. **ssh -l vlatam 192.168.2.32**
2. **ftp servidor.micolegio.edu.co**; como usuario emplear anonymous; como clave emplear su correo electrónico.
3. `ftp.ibiblio.org`
4. Abrir navegador, y después emplear el URL servidor.micolegio.edu.co
5. http://www.promo.net/pg
6. **mkdir ~/public_html; cd ~/public_html** En el archivo `~/public_html/index.html` transcribir el HTML presentado.
7. Abrir navegador; conseguir URLs.

2.4

1. **cd ~ ; cp /usr/doc/nvi/vi.beginner.gz . ; gzip -d vi.beginner.gz ; vi vi.beginner**
2. emacs ; C-h t
3. **ispell texto.txt**. Desde emacs usar el menu Tools+Spell Checking+Spell-Check Buffer. Desde **vim** instalar el modo sugerido y **F9**.

3.1

1. **cd ~ ; cp $(INSDOC)/AA_Linux_colegio.ps ~; gv AA_Linux_colegio.ps & ; ps2pdf AA_Linux_colegio.ps AA_Linux_colegio.pdf**
2. Las instrucciones están explicitas en la guía.
3. Cada modo provee teclas que abrevian comandos comunes y eventualmente acciones como procesar, desplegar DVI, etc.
4. Obtener las fuentes, leer el archivo `Leame.txt`, compilarlas con

   ```
   ./conf.sh
   make
   ```

   y experimentar.
5. Obtener las fuentes, compilarlas con make y experimentar.

3.2

1. Pega textos previamente cortados y almacenados en el anillo de textos cortados
2. Copia `datos2.txt` en `datos3.txt`.
3. Muestra la cantidad total de archivos en todos los subdirectorios del usuario.
4. Presenta el manifesto de Debian con las líneas numeradas por intermedio del programa **less**.
5. El resultado de **wc -w resumen.txt** debería ser menor que 5000.
6. El resultado de **wc -L correo.txt** debería ser menor que 75.
7. **head -n 40 pord.txt > uno.txt** y después **tail -n 60 pord.txt > dos.txt** (hay más formas).
8. `sort pers.txt | head -n 10`
9. `sort -u palabras.txt | wc -l` (también podría usarse `uniq`)
10. `ls -la ~/.* | wc -l`
11. **ls -l | mail usuario@micolegio.edu.co**

3.3

1. Basta teclearlos.
2. Convierte el archivo **entrada.dvi** a PostScript, dejando el resultado en **salida.ps**. La operación se realiza en segundo plano, la salida estándar es enviada al archivo `log` y el error estándar al archivo `err`
3.

   ```
   yes &
   ps -e
   kill -SIGTSTP n_p
   ps -e
   kill -SIGCONT n_p
   ps -e
   kill -SIGTERM n_p
   ```

   Tenga en cuenta que *n_p* es el número de proceso que tenga el proceso (que verá con el primer **ps -e**).

4. Una vez esté ejecutando **vi**, presione **Ctrl**-**Z** para interrumpir sesión e ir a línea de comandos. Desde la línea de comandos vea los procesos con **ps**, regrese entonces al proceso interrumpido con **fg %-**. Vuelva a interrumpir con **Ctrl**-**Z** y termine el proceso con **kill %-**.
5. **time ls -R /usr/doc**
6.

   ```
   at 10:00 4.06.1977 « FIN1
   mail $(USERNAME)@localhost « FIN2
   Acuerdese
   FIN2
   FIN1
   ```

7. Crear archivo con comandos para bc (digamos `factores.bc`) con la definición de la función `factores` presentada seguida de `factores(129098564527119574834)` y de `quit`. Ejecutar:

   `nohup /usr/bin/time -o restime.txt bc < factores.bc > resfactor.txt &`

   Que requiere 12 minutos en un Pentium MMX de 266Mhz para dar 2 12323 25849 202642820971

3.4

1.

   ```
   #!/bin/bash
   cd $HOME
   ls -R | sort -u | mail $USERNAME@$HOSTNAME
   ```

2. **echo $HOME; cd $HOME** El comodin ~ es remplazado por el valor de la variable $HOME.

3.

```
export PAGER=more
man man
```

4. Ejecuta el comando **ls** y el resultado del mismo como cadena lo presenta con el comando **echo**

5. Presenta `1+$N = 7`.

6. El archivo puede ser `recuerda.sh`, su contenido:

```
#!/bin/bash
sleep 90m
echo "Hacer tareas"
```

   después ejecutarlo.

7.

```
#!/bin/bash
echo 'PATH=$PATH:/opt/bin' > script2
echo 'alias sp="echo $PATH"' » script2
chmod +x script2
```

8.

```
#!/bin/bash
read -p "Numeros: " A B
echo $((A+B)) $((A-B)) $((A*B)) $((A/B))
```

3.5

1. Primero edite, por ejemplo con **vi ~/.bash_profile**, agregue por ejemplo al final:

```
if (test -f ~/.bashrc) ; then
    source ~/bashrc ;
fi;
```

2. Editar por ejemplo con **vi ~/.bashrc** y agregar:

```
PATH=$PATH:/opt/bin
export LD_LIBRARY_PATH=/opt/lib
```

3. Editar la variable PATH. Aunque en la documentación de `readline` dice que **Control-x Control-r** lee de nuevo el archivo de configuración de readline, pero el archivo que lee es `/etc/inputrc` y no `~/.inputrc`. Para que lea `~/.inputrc` debe reiniciarse bash.

4.

```
export LANG=en;
export LANGUAGE=en;
export LC_ALL=en;
man man
```

5. **PS1="\a\u \t>"**

6. Para que reinicie la aplicación Gnome basta salir de Gnome dejandola abierta, para que reinicie Emacs emplear panel de control Gnome.
7. Editar `~/.Xresource` y agregar:

```
emacs*Background: white emacs*Foreground: Wheat
emacs*pointerColor: Orchid emacs*cursorColor: Orchid
```

4.1

1. Algunas ídeas son: (1) un horario en el que puedan acudir los usuarios y disposición para atenderlos, correo electrónico, buzón de sugerencias (2) inventarios, mantenimiento preventivo, actualización continua de software y buena educación a los usuarios para que se conviertan en amigos y no enemigos (3) buscar por ejemplo en Internet y publicar en la Intranet los documentos que se consideren apropiados así como guías de uso —pueden ser estas o extensiones a estas.
2. Para medir aceptación pueden realizarse encuestas, tener un buzón de sugerencias, entrevistar usuarios. Para mejorar la aceptación puede hacerse publicidad (presentaciones en las que se destaquen las ventajas sobre otros sistemas a estudiantes, profesores y padres). Nos parece muy importante estar dispuesto a escuchar críticas y responder con argumentos o aceptar errores.
3. Creemos que la mejor estrategía es promover el autoaprendizaje, aunque en ocasiones es bueno no presentar los detalles del sistema, es importante poder tener acceso a los mismos para poder resolver problemas y aprovecharlo al máximo. Esto es posible con Linux y puede aprovecharse sólo aprendiendo (leyendo y practicando).

4.2

1. En nuestro parecer la mejor política es buena instrucción, es enseñar a los usuarios a apagar correctamente el sistema buscando que ellos quieran ayudar a mantener la red.
2. Se ven los procesos con **ps -ax**, sólo están procesos indispensables para el kernel y para mantener una sesión de **bash** (**kswapd**, **init**, **kflushd**, **kupdate**, **kpiod**, **bash**). Se llama *single* porque es para un sólo usuario y sin red, sirve para efectuar labores de mantenimiento como corrección de discos.
3. Se consigna arranques del sistema, reinicios y entradas a sesiones por parte de usuarios. Se consigna la fecha y hora de cada evento.
4. Editar por ejemplo con **vi /etc/logrotate** buscar la sección `/var/log/wtmp` y cambiar `rotate 1` por `rotate 2`. Además de rotar las bitácoras, eliminando la más antigua después de cierto tiempo, puede mantenerlas comprimidos y enviar correos a usuarios. Por defecto en Debian 2.2 este programa es ejecutado a diario.
5. Crear esta bitácora en blanco por ejemplo con **touch /var/log/btmp**. Intentar entrar con un usuario que no exista o una clave errada, ejecutar **lastb** y comprobar que se agregó una entrada.
6. **lastcomm** presenta los programas iniciados, cada uno junto con el usuario que lo inicio, el tiempo que duró la ejecución, la fecha cuando inició y flags que indica como inició y terminó; la información la extrae de `/var/account/pacct`. **ac** presenta tiempo total de conexión de cada usuario de acuerdo al archivo `/var/log/wtmp`.

7. Editar `/etc/lilo.conf` agregar las líneas `password=`*`clave`* y `restricted`, después ejecutar `/sbin/lilo`.

4.3

1. UID=0 y GID=0. Son menores a 100 lo que indica que deben ser fijos en cualquier computador con Debian.

2.

```
chgrp estudiantes tarea1;
chmod o+t tarea1
```

   Para la segunda parte se requiere que cada archivo no conceda permiso de lectura ni escritura al grupo ni a otros.

3. "enc.pl vida" (con MD5) da por ejemplo "1wv$mOwf63L.QvbJ.f7U362Os1" que puede ponerse bien en el campo para la clave de `/etc/passwd` o bien si las claves shadow están activas, dejar 'x' en `/etc/passwd` y remplazar la clave en `/etc/shadow`.

4.

```
adduser --ingroup users gloria
adduser gloria lpr
adduser gloria audio
adduser gloria floppy
su - gloria
mdir
exit
deluser --remove-home gloria
```

5. Los grupos básicos no tendrian clave: profesores, estudiantes y administación (podrían crearse grupos para otros miembros de la comunidad como exalumnos o padres). Los estudiantes pueden estar sólo en el grupo de estudiantes, los profesores pueden tener como segundo grupo estudiantes, los usuarios de la parte administrativa podrían tener como grupo principal administracion y como grupos secundarios profesores y estudiantes. En principio la información que cree un usuario no debe ser visible a su grupo. Podrían haber grupos con clave para grupos liderados por un profesor, administrador o estudiante que requieran manejar información separada (con un poco de instrucción los lideres podrían ser los administradores de grupo).

6.

```
addgroup --gid 1100 estudiantes;
addgroup --gid 1101 profesores;
addgroup --gid 1102 admin;
adduser --ingroup estudiantes esperanza;
adduser --ingroup profesores johannes;
adduser --ingroup tomas;
adduser estudiantes johanes;
adduser estudiantes tomas;
adduser profesores tomas
```

7. Suponiendo que los grupos son estudiantes (GID=1100), profesores (GID=1101), cuerpo administrativo (GID=1102) `adduser.conf` puede ser como el que viene por defecto con los siguientes cambios:

   ```
   USERGROUPS=no
   USERS_GID=1100
   ```

   Así por defecto todo usuario nuevo sería estudiante (para agregar profesores y administradores se usaría la opción `--ingroup` de **adduser**). Entre las opciones por defecto de ese archivo que puede ser mejor conservar están: bash como intérprete de comandos por defecto (que ha sido documentado en estas guías), como directorio para usuarios `/home` (que se ha supuesto en los primeros capítulos de estas guías), los números de usuarios que siguen las políticas de Debian.

8. Como se desea que por defecto los usuarios de unos grupos no puedan ver información de usuarios del mismo grupo: **umask u=rw,g=,o=**. Debe agregarse tal orden en `/etc/profile` y en los archivos de sesiones de **gdm** en el directorio `/etc/gdm/Session`.

9. Crear el archivo en `etc/skel` (como parte del contenido puede recordarse a los usuarios actualizar la información personal con **chfn**). En /etc/skel/.bashrc puede agregarse algo como:

   ```
   echo "Hay ayuda que puede completar en el archivo ayuda.txt"
   ```

10. Evitar entrada a la cuenta root desde el prompt de arranque. Evitar en lo posible emplear los bits SUID y GUID con usuario o grupo root. Emplear claves shadow con MD5 (en Debian se usa por defecto PAM que es muy apropiado). Educar a los usuarios en este tema, buscando que eligan buenas claves y ayuden a cuidar y construir la red (emplear para la cuenta root una muy buena clave conocida sólo por el administrador). Emplear cracklib para evitar uso de claves simples.

11. Editar en `/etc/gdm/gdm.conf` la línea:

    ```
    logo=/usr/share/pixmaps/gnome-logo-large.png
    ```

4.4

1. **ae**, **ed**, **vi**, **echo**
2.

   ```
   mkdir /root/nj;
   mkdir /root/nj/lib;
   cp /bin/cat /root/nj;
   cp /lib/libc.so.6 /root/nj/lib;
   cp /lib/ld-linux.so.2 /root/nj/lib
   ```

3. **cat /dev/vcs1**
4. El archivo es `/etc/motd`, su primera línea que contiene el nombre de la máquina y la versión del sistema es generada en cada arranque por el script `/etc/init.d/bootmisc` —que es arrancado desde el nivel S—, puede evitarse la modificación poniendo en "no" la variable EDITMOD en `/etc/default/rcS`.

5. **cat /proc/cpu/info**

6.

```
ln -s /dev/ttyS1 /dev/mouse
ln -s /dev/hdc /dev/cdrom
```

4.5

1.

```
PATH=$PATH:/home/pedro/bin;
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/home/pedro/lib
```

2.

```
chmod a+rx
/home/pedro/bin
chmod a+rx /home/pedro/lib
echo "/home/pedro/lib" » /etc/ld.so.conf
ldconfig;
```

además agregar `/home/pedro/bin` a la variable `PATH` en `/etc/profile`

3. Otro puede ser http://www.linux-directory.com/ (http://www.linux-directory.com) Freshmeat es el más tradicional, allí a diario se anuncian nuevas versiones de programas (quien lo desee puede anunciar sus programas).

4. Una vez obtenidas las fuentes se consulta la documentación y se siguen.

5. **ldd /usr/bin/gcalc**

6.

```
echo /opt/lib » /etc/ld.conf
ldconfig
```

7.

```
wget URL/paquete.deb
dpkg -i paquete.deb
```

puede revisarse el nombre de lo que se instala (para usarlo) con: **dpkg -L paquete** y para desinstalar:

```
dpkg -r paquete
```

8. **tar cvfz ej.tar.gz ej/**

4.6

1. Hay una lista en la sección de enlaces de SLEC (http://ar.groups.yahoo.com/group/slec)

2. Puede ser revisando en http://www.sourceforge.net

5.1

1. `/sbin/init`
2. `/etc/rcS.d/S10checkroot.sh`
3. Se inicia **syslogd** que maneja bitácoras, se inicia **kerneld** que puede cargar módulos por demanda, si el sistema usa PPP y existe el archivo `/etc/ppp/ppp_on_boot` con información apropiada se conecta a un ISP, inicia servidor de nombres, inicia programas usados por NFS, inicia servidor de correo, ratón y servicios en red configurados en `/etc/inetd`, se inicia **logoutd** para evitar ingreso a ciertos usuarios a ciertas horas, se inicia cola de impresión, se crea el enlace `/dev/MAKEDEV` de `/sbin/MAKEDEV`, se inicia el servidor de SSH, inicia servidor de fuentes para X11, inicia el servidor de NFS, inicia **atd** que maneja tareas iniciadas con **at**, inicia **cron**, elimina `/etc/nologin` –que había sido creado en el nivel S para no permitir accesos durante el arranque, inicia el administrador de vistas.
4. Crear un script (digamos `mailboot`) en `/etc/init.d` con:

   ```
   date | mail usuario@localhost
   ```

   Después ejecutar **update-rc.d mailboot start 99 3**
5. Dos posibilidades: init 3 o cambiar en `/etc/inittab` el nivel por defecto a 3 —después restablecer el 2.

5.2

1. Ejecutar **kernelversion** o **uname**
2. Ejecutar **lsmod**, consultar información de cada uno por ejemplo con **modconf**.
3. Para examinarlo: **less /etc/X11/XF86Config**.
4. Son nombres de funciones o variables de las fuentes en C del módulo y del kernel, para que los módulos puedan usar las funciones y viceversa la ubicación en memoria de estos símbolos debe resolverse dinámicamente. Los símbolos del kernel puede examinarse en `/boot/System.map-`*`version`* o en `/proc/ksyms`.

5.3

1.

   ```
   idcomp=`uname -a`
   echo "Configuración de: $idcomp";
   echo;
   echo "+
   Procesador=====================================================";
   echo;
   cat /proc/cpuinfo
   echo;
   ```

```
echo "+
Memoria=======================================================";
echo;
free
echo;
echo "+ IRQ, DMA, Puertos ===========================================";
echo
cat /proc/interrupts
cat /proc/dma
cat /proc/ioports
echo;
echo "+ PCI=======================================================";
echo;
cat /proc/pci
```

2. Seguir instrucciones

5.4

1.

```
echo "+ IDE ========================================================";
for i in /proc/ide/hd* ; do \
cat $i/driver; \
if (test -f $i/geometry) then { \
  cat $i/geometry; \
  d=`echo $i | sed -e "s|/proc/ide/|/dev/|g"`; \
  echo $d; \
  fdisk  -l -u $d; \
} fi;\
done
echo "+ X-Window ========================================================";
r=`ps ax | grep "[X]11/X"`
if (test "$r" != "") then {
f=/etc/X11/XF86Config
if (test -f "/etc/X11/XF86Config-4") then {
echo "Servidor 4.x";
f="/etc/X11/XF86Config-4";
} fi;

grep "^[ ]*XkbLayout" $f
grep "^[ ]*Protocol" $f
grep "^[ ]*Device[ ]*\"/" $f
grep "^[ ]*HorizSync" $f
grep "^[ ]*VertRefresh" $f
grep "^[ ]*Chipset" $f

cat /var/log/XFree86.0.log
}
else {
```

```
echo "No se ha configurado X-Window" > /dev/stderr
} fi;
echo;
echo "+ Módulos=====================================================";
echo;
lsmod
echo;
if (test "`lsmod | grep "soundcore"`" = "") then {
echo "No hay dispositivo de sonido" > /dev/stderr;
} fi;

echo "+ /etc/modules.conf============================================";
echo;
ls /etc/modutils
cat /etc/modules.conf
echo;
echo "+ /etc/hostname================================================";
echo;
cat /etc/hostname
echo;
echo "+ /etc/network/interfaces======================================";
echo;
cat /etc/network/interfaces
echo;
```

2. Base de datos de las distribuciones Mandrake (http://www.linux-mandrake.com/en/hardware.php3) o Suse (http://hardwaredb.suse.de/index.php?LANG=en_UK).
3. Por ejemplo hay soporte de Intel para algunas tarjetas gráficadoras http://support.intel.com/support/graphics/linux/graphics.htm.

5.5

1. Para verificar puede emplearse **vi**.
2. Puede intentarse la compilación de un programa grande para tratar de verificar que procesador y memoria funcionan.
3. **free**
4. Revisar procesos con **ps**. Buscar en la secuencia de arranque del nivel en el que está los que quiere eliminar y retirar el enlace apropiado de `/etc/rcx.d`
5. Puede por ejemplo revisar la lista de los programas que están instalados en su computador con el siguiente script awk (digamos que lo llama `s21.awk`):

```
/^Package:/ {
  pack=substr($0,9,length($0)-8);
}

/^Status: install ok installed/ {
  print pack;
}
```

```
/.*/ {
}
```

usando **awk -f s2l.awk /var/lib/dpkg/status**

6. Pude consultarse `/usr/share/emacs/site-lisp/debian-startup.el` que indica que los archivos que se cargan al arranque están en: `/etc/emacs/site-start.d`.
7. Si es posible. http://www.rocklinux.de/about.html

5.6

1. Se carga el archivo original con `loadkeys` así ya se pueden generar todas excepto la ü y Ü. Para generar estas dos últimas puede navegarse sobre las guías y copiar y pegar, puede usarse `+udiaeresis` y `+Udiaeresis` o puede emplearse un editor de texto que permita ingresar un código ASCII o Unicode.

6.1

1. Cómo se documenta en estas guías puede usarse en todos los clientes como sistema de escritorios además un servidor puede configurarse para: dar conexión a Internet a todos los computadores, servidor de nombres, servidor de correo, servidor web, servidor ftp, compartir archivos con NFS, centralizar manejo de usuarios con NIS, compartir impresora, dar acceso remoto de forma segura con ssh, servidor cvs. Además puede servir por ejemplo como: servidor de nntp, servidor de ntp, servidor de bases de datos relaciones, servidor con clientes delgados (sin disco duro).
2. La propuesta de un colegio colombiano en 2002, incluyendo costos está disponible en: http://www.gfc.edu.co/proyectos/conexioninternet/conectainternet.html
3. Para transmisión por fibra óptica se emplea el protocolo ANSI FDDI (Fiber Distributed Data Interface), que puede transmitir información hasta 100Mb por segundo (datos en http://www.iol.unh.edu/training/fddi/htmls/ y http://www2.rad.com/networks/1995/fddi/fddi.htm). Puede emplearse para LANs aunque soporta una distancia máxima de 100Km. De acuerdo al Net-HOWTO se usan tarjetas que se conectan al computador, de forma análoga al caso de Ethernet. De acuerdo a Hardware-HOWTO son soportadas las tarjetas DEC DEFA (EISA) y DEFPA (PCI) en kernels posteriores al 2.0.24.

6.2

1. Comprobar con **ping 127.0.0.1** Puede verse la configuración en `/etc/network/interfaces`:

   ```
   iface lo inet loopback
   ```

2. **ifdown lo; ifconfig; ifup lo**

3. MTU Maximum Transfer Unit, es la máxima longitud de un paquete soportada por el protocolo y el dispositivo, en el caso de Ethernet es 1518 en el caso de FDDI es 4905.
4. modconf hasta que el módulo carge ; configurar eth0 en /etc/networks/interfaces; ifup eth0 para iniciar interfaz; route para ver que la compuerta sea el servidor; ping nombre para comprobar que está en /etc/hosts;

6.3

1. Se planea la ubicación y la nueva dirección IP que tendrá. Se asegura que el hub tenga un puerto más o se cambia por uno con más puertos. Se hace un cable nuevo de acuerdo a ubicación que se designe para el nuevo computador. Se instala la tarjeta de red en el nuevo computador y se conecta al hub usando el cable. Se instala/configura el sistema operativo dle nuevo computador, asegurando que la tarjeta quede asociada a una interfaz (eth0) y que tal interfaz quede con la dirección IP planeada usando como compuerta el servidor, después se prueba.

6.4

1. A `/etc/inetd.conf` debe agregarse la línea:

   ```
   daytime stream tcp nowait root internal
   ```

   Después de reiniciar: **/etc/init.d/inetd restart** Y finalmente se examina la respuesta de este protocolo:

   ```
   $ telnet localhost 13
   Trying 127.0.0.1...
   Connected to localhost.
   Escape character is '^]'.
   Wed Jan 1 19:26:31 2003
   Connection closed by foreign host.
   ```

2.
   ```
   ping structio.sourceforge.net
   whois sourceforge.net
   dig -x 204.152.184.85
   1. 66.35.250.209
   2. dns-tech@valinux.com
   3. pubweb.isc.org, ns-ext.vix.com, ns1.gnac.com
   ```

3. Instalar cliente y servidor NFS. **mkdir /aux; mkdir /mnt/tmp; cp /etc/hosts /aux** Agregar a `/etc/exports` la línea:

   ```
   /aux localhost(rw)
   ```

   y después **mount -t nfs localhost:/aux /mnt/tmp**

4. En algún archivo de inicialización (preferiblemente `/etc/security/pam_env.conf`) agregar:

   ```
   export PATH=$PATH:/opt/bin
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/lib
   ```

5. Ejecutar **eximconfig** y elegir configuración como sistema satélite como smart host usar servidor. Después **mail -v root@servidor**
6. Editar el archivo `/etc/exports` del servidor, agregar:

   ```
   /var/mail  *.micolegio.edu.co(rw,no_root_squash)
   ```

   Reiniciar NFS. En el cliente agregar en `/etc/fstab`:

   ```
   servidor.micolegio.edu.co:/var/mail /var/mail nfs rw 0 0
   ```

   y después **mount -t nfs servidor.micolegio.edu.co:/var/mail /var/mail**
7. En el caso de **mutt** esa es la configuración por defecto.
8. **apt-get install apache**
9. Entre los módulos que carga apache, configurados en `/etc/apache/httpd.conf` debe estar

   ```
   LoadModule userdir_module /usr/lib/apache/1.3/mod_userdir.so
   ```

Índice