

CAPÍTULO 1

Introducción

Como su propio nombre indica, éste es un curso avanzado de Linux. Esto quiere decir que se supone que estás familiarizado con la interfaz del sistema, que conoces el manejo del ratón y que tienes cierta soltura con algunas aplicaciones, como el navegador de Internet o el editor de texto *gedit*, ya que aunque iremos guiándote en todo momento, no se va a entrar en cómo utilizarlas. Dicho de otra forma: se supone que has realizado un curso básico de Linux.

Lo que obligatoriamente debes conocer es la contraseña de administrador; en caso contrario no podrás realizar la mayoría de las acciones que se te proponen. Si no la conoces ponte en contacto con la persona encargada del sistema y coméntale tu problema. El saber esta contraseña es fundamental para ejecutar algunas aplicaciones y realizar ciertos cambios.

Ten en cuenta que la presente documentación se ha realizado directamente (sin realizar ninguna actualización) a partir de la distribución GuadaLinux v1.0 (también llamada versión ciudadano rc5) distribuida por la Junta de Andalucía en la Conferencia Internacional de Software Libre, celebrada en Málaga del 18 al 20 de Febrero de 2004. GuadaLinux está basada en la distribución Debian. Esto lo hacemos para tener una referencia de partida, dada la evolución y variedad que sufre todo sistema una vez instalado. Por ello debes tener en cuenta que puede haber algún cambio sobre la versión que tienes instalada, ya que se demostró que dicha distribución tenía algunos fallos, que se solucionan simplemente realizando una actualización; salvo en un par de casos puntuales estos fallos no afectan a esta documentación. También es posible que se produzca algún pequeño cambio respecto al nombre o ubicación de las aplicaciones, así que si no encuentras un programa o un archivo justo donde te decimos es probable que no ande muy lejos (utiliza las herramientas de búsqueda para localizarlo).

Intentaremos en la medida de lo posible realizar todas las acciones desde el entorno gráfico, pero cuando uno debe hacer cambios importantes en el sistema tarde o temprano debe enfrentarse con el terminal de comandos. No le tengas

miedo: la consola permite realizar todo lo que se puede hacer de forma gráfica, y muchos más. La clave para dominar las tareas avanzadas reside en controlar la consola de comandos; no te preocupes, con el tiempo verás que no es tan complicado y esta complicación se ve recompensada con creces por la potencia y versatilidad que tiene.

Por otro lado, al final de cada tema se te proponen unos ejercicios, de manera que tengas otra referencia (además de los ejemplos desarrollados) a la hora de realizar modificaciones.

Una última advertencia: a veces se van a hacer cambios que afectan inmediatamente al sistema (algunos irreversibles, los cuales te avisaremos de antemano), y es posible que pierdas la conexión a Internet o dejes al sistema inservible; esto no ocurrirá si sigues al pie de la letra las instrucciones, pero la forma de aprender Linux es probando y experimentando cosas nuevas. Si tienes datos importantes, realiza una copia de seguridad de ellos; además es muy recomendable que realices una instalación nueva a parte de la que utilices; de esta forma no modificaras el sistema que utilizas normalmente y partirás del “estado conocido” del que hablábamos antes. En cualquier caso es buena idea realizar copias de seguridad con cierta frecuencia.

CAPÍTULO 2

Usuarios y grupos

Gestión básica de usuarios

Linux es un sistema multiusuario; eso quiere decir que puede ser utilizado por varias personas, pudiendo cada una de ellas particularizar el sistema a su gusto. Más aún, cada una tendrá su propio espacio de almacenamiento para guardar sus archivos privados.

Al mismo tiempo los usuarios se distribuyen en grupos, de forma que aquellos que comparten alguna actividad se encuentran perteneciendo a uno de estos grupos. De esta forma es fácil permitir o prohibir a determinados usuarios que utilicen algún recurso (una impresora, Internet, la grabadora de CD-ROM, ...).

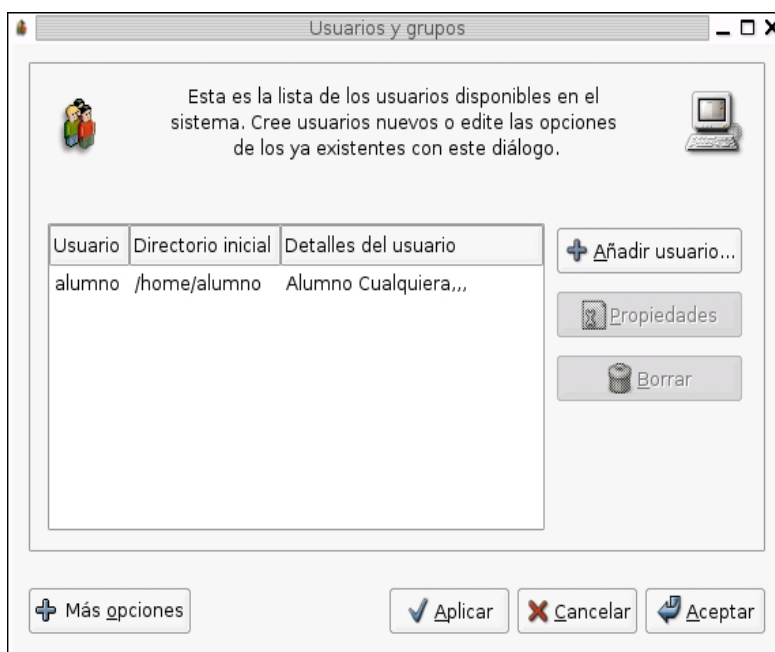
Cada usuario se identifica ante el sistema con un nombre de usuario, también llamado “login”, y una contraseña o “password”. De esta manera para que el usuario pueda acceder al sistema debe introducir correctamente ambos elementos; en caso contrario el sistema no permitirá la entrada. Al conjunto de cada una de estas parejas login-password se le denomina “cuenta”.

En todo sistema Linux existe siempre un usuario especial cuyo nombre es “root”, conocido también como “superusuario” o “administrador”. Éste es el responsable del buen funcionamiento del sistema; ya veremos que cada vez que tengamos que realizar alguna tarea importante para el sistema deberemos introducir su contraseña.

Pero éste no debe ser el único usuario que exista, ya que su utilización frecuente puede originar fallos de seguridad o pérdida de datos (incluso la pérdida total del sistema). Para solucionar esto la persona encargada de la administración utilizará la cuenta de root sólo cuando sea estrictamente necesario (para efectuar cambios en el sistema por ejemplo), mientras que para el resto de acciones que

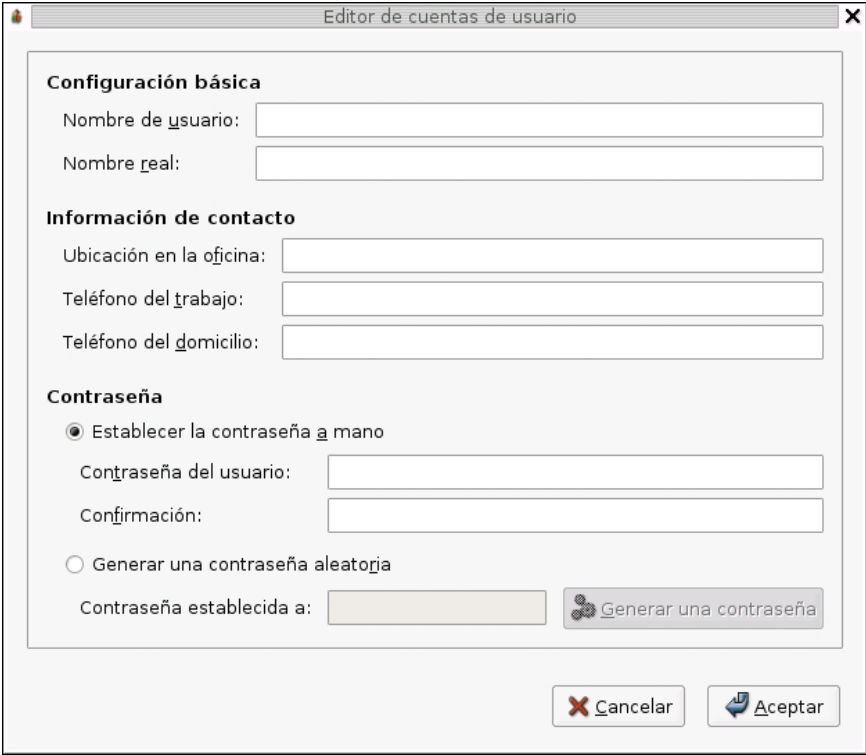
necesite (escribir documentos, conectarse a Internet, ...) utilizará una cuenta con menores privilegios (y por lo tanto menos peligrosa)

Para tratar usuarios y grupos se utiliza una herramienta llamada “Usuarios y grupos”, localizada en el menú: Aplicaciones -> Herramientas del Sistema -> Panel de Control. Tras pulsar sobre ella verás una ventana como ésta (aunque antes te va a pedir la contraseña de root).



Podemos ver que consta de una parte central, con la lista de los usuarios que existen en ese momento, y a su derecha se muestran las opciones que se pueden realizar. Sólo las iluminadas están disponibles (las sombreadas no lo están por ahora). Según la imagen sólo existe un usuario, llamado “alumno”, cuyo directorio inicial es “/home/alumno” y cuyo nombre real es “Alumno Cualquiera”.

Para crear un nuevo usuario hay que pulsar en “Añadir usuario...”; veremos la ventana con el formulario de la imagen anterior; no todos los campos son necesarios, pero suele ser buena idea poner la mayor información posible.



The screenshot shows a window titled "Editor de cuentas de usuario" with a close button (X) in the top right corner. The window contains three sections: "Configuración básica", "Información de contacto", and "Contraseña".

Configuración básica

Nombre de usuario:

Nombre réal:

Información de contacto

Ubicación en la oficina:

Teléfono del trabajo:

Teléfono del domilio:

Contraseña

☒ Establecer la contraseña a mano

Contraseña del usuario:

Confirmación:

☐ Generar una contraseña aleatoria

Contraseña establecida a:

At the bottom right, there are two buttons: "Cancelar" (with a red X icon) and "Aceptar" (with a blue checkmark icon).

En primer lugar se necesita el nombre de usuario; éste será su “login” para acceder al sistema; es muy recomendable que tenga entre 4 y 8 caracteres, comience por una letra, esté en minúsculas y no contenga ni espacios ni signos de puntuación (ni la letra “ñ”); nombres incorrectos serían:

peperodriguezvazquez 1pepe PepeRodríguez pepe.rodri

Aunque es posible poner algunos de estos nombres, a la larga sólo te causarán problemas (te lo aseguro). Nombres válidos serían:

peperv pepe1 peperodri

En el apartado “Nombre real” se debe poner el nombre y apellidos de la persona; aquí sí que se pueden poner mayúsculas, espacios, tildes ...

Los tres siguientes apartados se mantienen por compatibilidad con los antiguos sistemas Unix; puedes obviarlas si quieres.

El campo “Contraseñas” también es obligatorio; tenemos dos opciones: establecerla a mano (debemos escribirla dos veces, para evitar equivocaciones) o que el sistema la genere por nosotros (aleatoriamente). Una buena contraseña

debería tener entre 6 y 8 caracteres, y contener letras (algunas en mayúsculas), números, símbolos de puntuación, ... Pero recuerda que debes memorizarla ya que hay que introducirla cada vez que quieras entrar en el sistema. Para ver algunas contraseñas “buenas”, selecciona la opción aleatoria y pulsa “Generar una contraseña” varias veces. Y por supuesto, nunca se te ocurra poner algo que tenga que ver con tu nombre, tu edad o tu mascota (una de las principales formas de entrar en un sistema vienen por aquí). Observa que por seguridad aparecen “*” en lugar de los caracteres.

Así que rellena los campos necesarios con los datos adecuados y pulsa “Aceptar”; volveremos a la pantalla principal, donde ahora aparece el usuario que acabas de crear.

La opción de “Propiedades” muestra la misma pantalla que la de “Añadir usuario...” pero con los datos que se introdujeron para él anteriormente, pudiendo modificarse ahora según se quiera. Para que esté disponible esta opción hay que seleccionar previamente un nombre de la lista.

Por último, el botón “Borrar” elimina el usuario; como medida de precaución pide confirmación ya que la eliminación es irreversible. Al eliminar un usuario no se elimina su directorio personal ni su contenido, por lo que habrá que realizarlo a mano si es necesario.

Gestión avanzada de usuarios

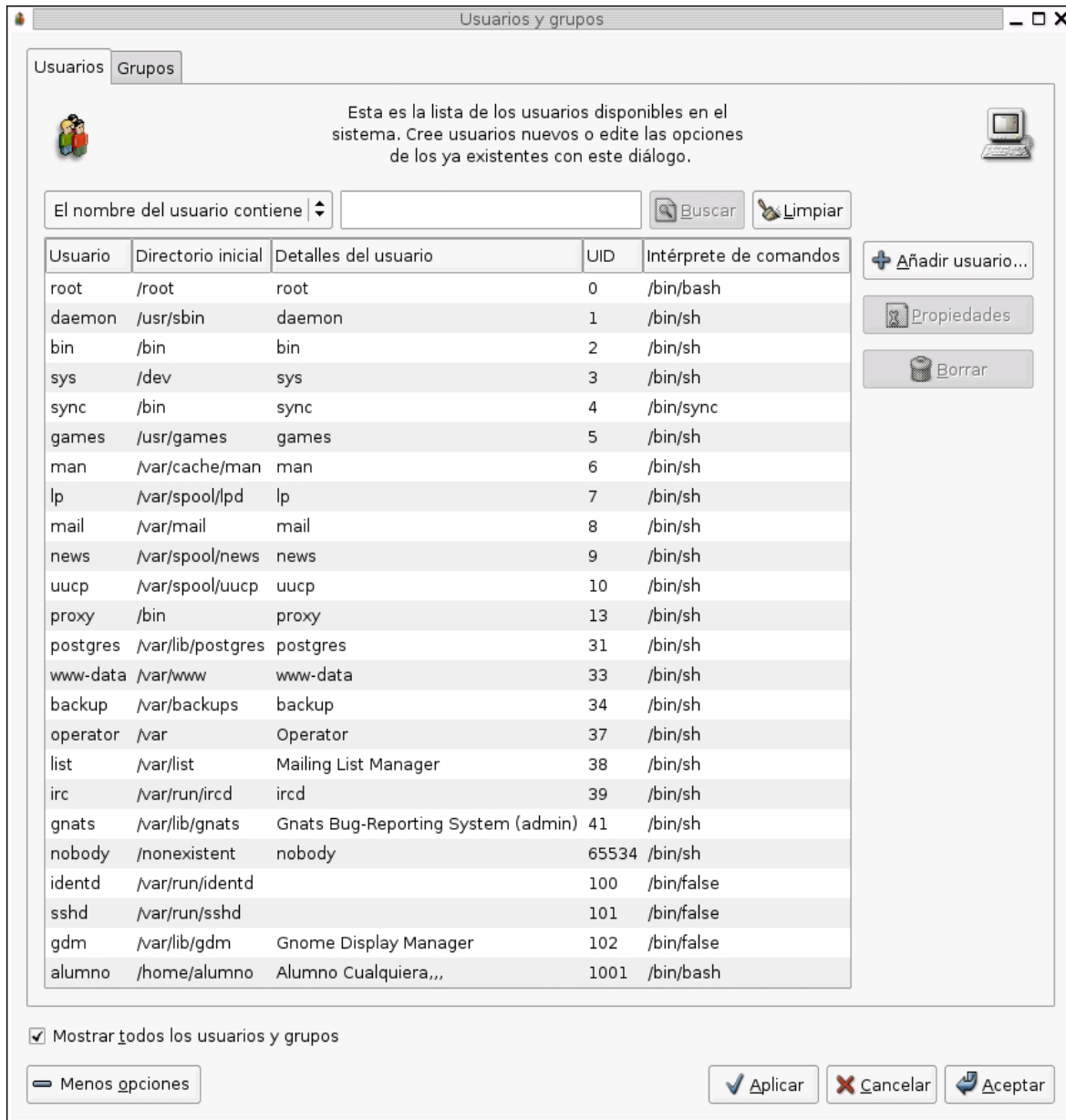
Antes se comentó que siempre existe un usuario llamado “root”, pero sin embargo en esta lista no aparece. Esto es debido a que por defecto y mientras no digamos lo contrario lo que vemos es la lista de usuarios “reales”, es decir, asociados a personas humanas. En realidad existen tres tipos de cuentas diferentes que se utilizan en Linux:

- Cuentas de usuario completas: También llamadas cuentas de “shell”, es la que acabamos de ver, asociada a una persona que puede introducir su login y contraseña y acceder al sistema para utilizar los programas y aplicaciones que hay en él.

- Cuentas de acceso restringido: Son cuentas en las que el usuario sólo tiene acceso a servicios específicos del sistema (por ejemplo el correo electrónico) y por ello no disponen de “shell” o intérprete de comandos. La razón para ello es que una “shell” es potencialmente peligrosa, ya que permite cierta libertad de movimientos y ejecución de ciertos programas.
- Cuentas para programas y “daemons” (demonios): Algunos programas necesitan identidades propias para funcionar; esta es una medida de seguridad ya que generalmente cuando un programa se ejecuta lo hace con los privilegios que tiene el usuario que lo ha ejecutado; esto es realmente peligroso para root. En su lugar se crean unos usuarios “ficticios” que sólo tienen los privilegios adecuados para realizar la tarea para la que han sido llamados.

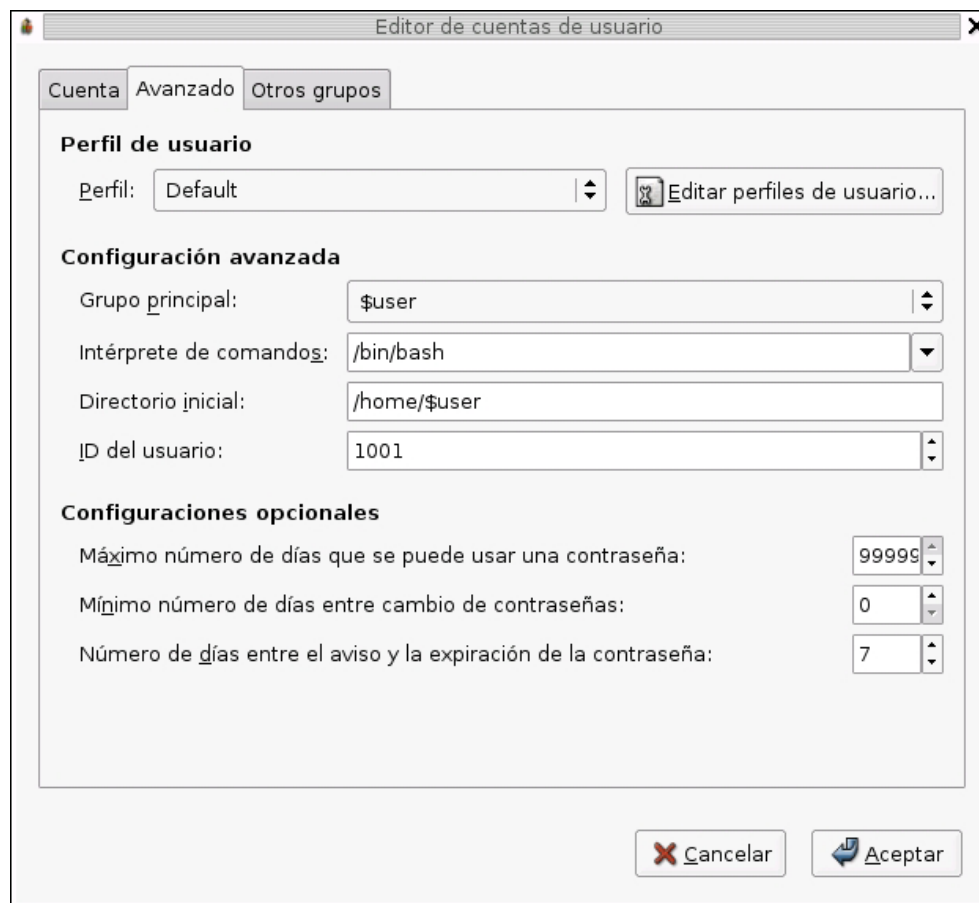
Para poder ver todas esas cuentas de usuario, debes pulsar en “Mas opciones”; verás la lista completa y además información adicional sobre cada uno. El último de la lista es el usuario que has creado; observa cómo en la última columna aparece “/bin/bash”, que es el intérprete de comandos que utilizan generalmente los usuarios. El primero de la lista es “root”, que también utiliza esa misma “shell”. El resto son cuentas para programas y demonios, de los cuales algunos tienen “shell” (/bin/sh) y otros no (/bin/false).

Además de esta información, vemos una columna denominada UID (siglas de “Usuary Identifier”, o identificador de usuario); cada usuario (sea del tipo que sea) dispone de un UID, que lo identifica de forma única mediante un número. Es de esta forma como el sistema se refiere a los usuarios (para él no somos más que un número), aunque a nuestros ojos utilicemos el nombre de usuario. Hay algunos número reservados, como el “0” para root; los usuarios “normales” se empiezan a numerar en Debian a partir del 1000 (en RedHat, por ejemplo, lo hacen a partir del 500), reservándose los números inferiores para programas y demonios (no te preocupes por esto, cada programa sabe qué UID usar).



Si el número de usuarios es muy elevado se pueden hacer búsquedas por nombre de usuario, por UID, ... En cualquier momento podemos volver a la vista básica desmarcando la opción “Mostrar todos los usuarios y grupos”

Ahora que tienes activada la opción de “Más opciones”, en la ventana de crear usuarios aparecen dos nuevas pestañas; observa la siguiente imagen:



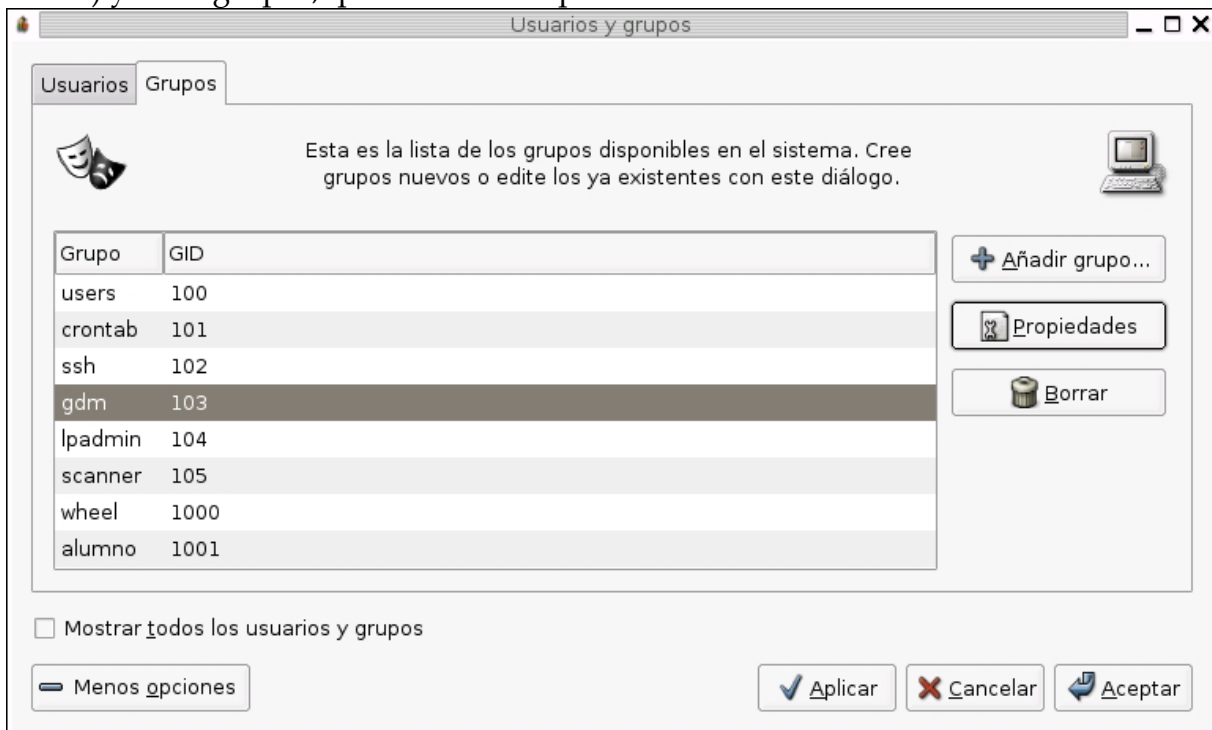
Ahora puedes modificar otros parámetros que antes ni se podían ver, como el intérprete de comandos a utilizar, el directorio inicial, el UID, ... Hablaremos de los grupos en el apartado siguiente. Las configuraciones opcionales tratan sobre la contraseña, y son:

- **Máximo número de días ...** : Indica cuándo caduca la contraseña; pasados esos días el usuario no podrá acceder al sistema; es una forma de obligar a que el usuario cambie la contraseña cada cierto tiempo.
- **Mínimo número de días ...** : Especifica cuántos días deben pasar para que el usuario pueda cambiar de nuevo la contraseña.
- **Número de días ...** : Se mostrará un aviso al usuario esta cantidad de días antes de que la contraseña expire.

Gestión de grupos

Después de lo que acabamos de ver la gestión de grupos te resultará sencilla, ya que guarda muchas similitudes con la de usuarios.

En la vista avanzada hay dos pestañas, una la de usuarios (que ya hemos visto) y la de grupos, que tiene este aspecto:



Si los usuarios tenían un identificador, los grupos no van a ser menos, lo que pasa que ahora se denomina GID (“Group Identifier”, identificador de grupo); de esta forma a cada grupo se le asigna a la hora de crearlo un número que es único. En la imagen anterior aparece la lista abreviada, para verla completa pulsa en “Mostrar todos los usuarios y grupos”; por ejemplo el grupo “gdm” se utiliza para definir los usuarios que tendrán login gráfico.

Hay una cosa que merece la pena resaltar; cada vez que se añade un usuario, se crea automáticamente un grupo con su mismo nombre dentro del cual se incluye dicho usuario. De esta forma, si creamos el usuario “alumno” (cuyo UID

es 1001) se creará un grupo (formado de momento por él únicamente) que se llama “alumno” (con GID 1001, observa la imagen anterior). Este grupo se denomina “grupo principal del usuario”; un usuario podrá pertenecer a uno o más grupos, pero nunca podrá abandonar su grupo principal (que, por otro lado, se puede cambiar, pero siempre debe estar definido este grupo principal).

Para añadir un usuario a un grupo, tan sólo selecciona el grupo y pulsa “Añadir grupo...”, donde podrás introducir el nombre, su GID (suele ser buena idea aceptar la sugerencia que se nos hace) y qué usuarios pertenecen al grupo; fíjate que sólo aparecerán en la lista los usuarios “reales”.

Una advertencia: a veces, para que los cambios sean visibles, hay que salir de la aplicación y volver a acceder a ella; otras veces basta con pulsar el botón de “Aceptar” o “Aplicar”.



Por último, recuerda no confundir usuarios con grupos, aunque tengan el mismo nombre se refieren a cosas distintas.

Ejercicios

- 1) Crea 5 usuarios, llamados usu1, usu2, usu3, usu4 y usu5. Crea contraseñas aleatorias para todos.
- 2) Modifica usu1 para que su contraseña sea la palabra “ejemplo”
- 3) Crea dos grupos, llamados “pares” e “impares”; añade usu2 y usu4 al grupo “pares” y usu1, usu3 y usu5 al “impares”.
- 4) Añade el usuario usu1 al grupo usu3 y el usuario usu3 al grupo usu1.
- 5) Borra el usuario usu3 y usu5. ¿Siguen existiendo los grupos usu3 y usu5? ¿Por qué crees que ocurre esto?
- 6) Vuelve a crear el usuario usu3 y establece para él un UID mayor que el que tenga usu4 (por ejemplo 1100). ¿Cuál es el grupo principal de usu3? ¿Se ha creado algún grupo nuevo? ¿Por qué?

Soluciones

- 3) Ten cuidado de no repetir el GID, recuerda que debe ser único; cualquier número por encima de 1000 (y que no esté ocupado) puede servirte.
- 4) Para incluir a un usuario en un grupo puedes hacerlo desde el menú de grupos o el de usuarios; observa cómo el grupo principal no aparece en la lista.
- 5) Si sales y vuelves a entrar de la aplicación verás que el grupo usu5 ha desaparecido, mientras que usu3 sigue existiendo. Esto es así porque en usu3 sigue habiendo usuarios (recuerda que metiste a usu1 en este grupo en el ejercicio anterior); un grupo (por encima del GID 1000) se elimina cuando no quedan usuarios en el grupo.
- 6) El grupo principal de usu3 resulta ser el grupo usu3 (que ya existía), pero observa que el UID y el GID no coinciden en número; esto no supone ningún problema para el sistema. Por eso no se crea ningún grupo nuevo (no puede haber dos grupos con el mismo nombre).

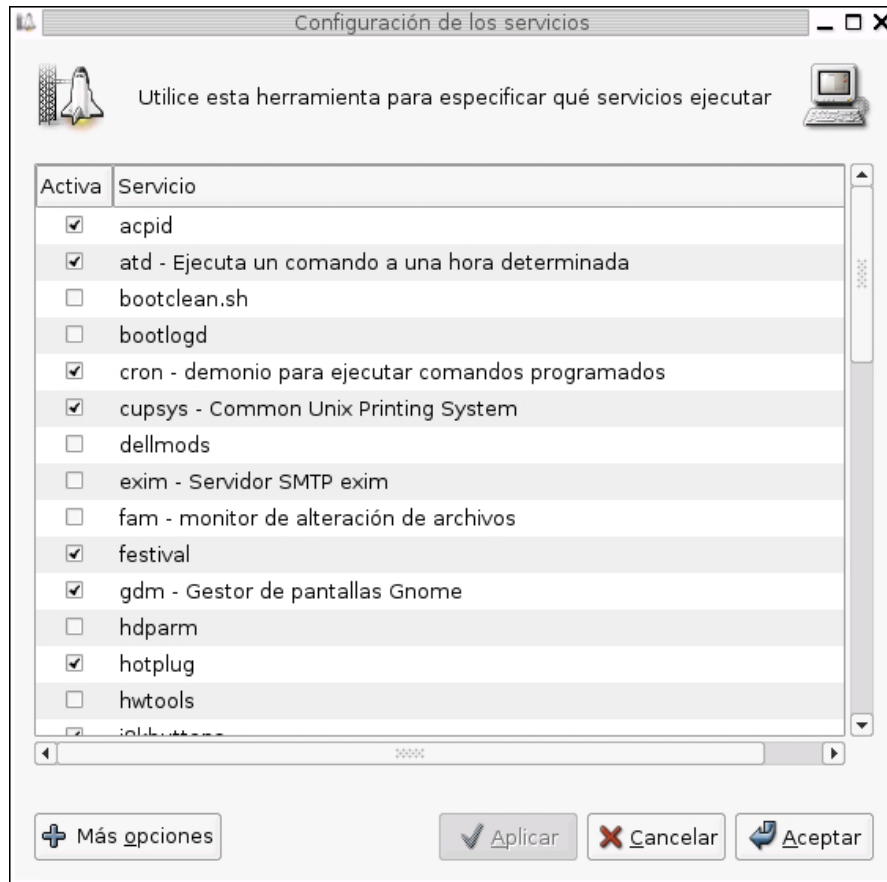
CAPÍTULO 3

Administración básica del sistema

Arranque y Parada de Servicios

Existen en un sistema Linux un conjunto de programas que están en continuo funcionamiento; estos programas son conocidos por demonios o “daemons”, que están asociados a determinados “servicios” y que se encargan de servir a determinados propósitos: los hay que ejecutan una tarea periódicamente, otros almacenan información, y otros simplemente están a la espera de que alguien los necesite (por ejemplo, el servicio de impresión). No todos los servicios tienen un demonio asociado, los hay que únicamente son scripts que ejecutan una serie de tareas (aquellos que lo tienen suelen terminar en una “d” para indicar eso precisamente).

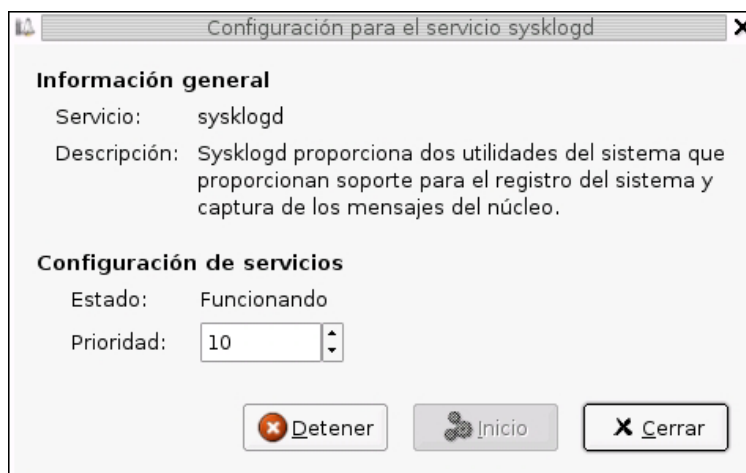
Para poder seleccionar qué servicios queremos tener funcionando en nuestro sistema y cuales disponemos de una aplicación gráfica denominada “Servicios”, situada en el menú: Aplicaciones -> Herramientas de Control -> Panel de Control, cuya imagen puede ver a continuación.



En la lista aparecen todos los servicios disponibles; con una marca se señalan los que están en funcionamiento actualmente. Iniciar o detener un servicio es tan simple como marcar o desmarcar la casilla correspondiente. Ten en cuenta que a medida que vayas instalando ciertos programas la lista irá creciendo.

No conviene tener activos servicios que no vamos a utilizar: tener servicios funcionando que no usamos puede hacer al sistema vulnerable (y puede hacer que funcione más lento), mientras que si detenemos algún servicio esencial el sistema puede funcionar mal.

Como puede que no sepamos qué hace un servicio determinado, disponemos de una pequeña ayuda, para lo que hay que activar el botón “Más opciones”; de esta forma aparece un botón de “Propiedades del servicio”, en el que una ventana emergente nos muestra información relevante de un servicio concreto. En la siguiente imagen se muestra esta ventana para “sysklogd”, un servicio encargado de almacenar los mensajes que emite el núcleo y que pueden servir para identificar un mal funcionamiento del sistema.



Además de proporcionarnos una manera alternativa de iniciar o detener el servicio, podemos establecer su prioridad, es decir, el orden en el que se inician respecto al resto de servicios; la prioridad oscila desde el 1 (el primero en iniciarse) hasta el 99 (el último en hacerlo).

Al mismo tiempo, el botón “Más opciones” enseña un menú donde se pueden seleccionar varias listas, cada una para configurar la lista de servicios a iniciar en diferentes situaciones:

- Deteniendo el sistema: son los scripts que se ejecutarán al apagar la máquina; generalmente desactivan alguna característica o apagan algunos dispositivos (desmontar unidades, por ejemplo)
- Modo gráfico: son los servicios a que se activarán al tener configurado el inicio en modo gráfico (gdm)
- Modo texto: servicios a iniciar al arrancar el sistema en modo texto (consola)
- Reiniciando el sistema: igual que el primero, pero cuando se va a reiniciar el sistema.

Puede parecer extraño la división anterior, pero la razón hay que buscarla en los sistemas Unix tradicionales, en los que es necesario distinguir entre un reinicio y un apagado total del sistema; para ordenadores personales ambas situaciones son equivalentes y no presenta más inconveniente que tener que configurar dos veces las mismas opciones. Del mismo modo hay servicios que no se necesitan en

caso de iniciar en modo texto (como por ejemplo el demonio “gdm”).

Ejecución periódica de comandos

Algunas veces surge la necesidad de ejecutar un programa de manera regular a intervalos periódicos, o que puedan ejecutarse en determinados momentos sin tener que estar delante del ordenador (por ejemplo, para realizar copias de seguridad a altas horas de la noche, cuando nadie esté usándolo). Afortunadamente Linux proporciona las herramientas necesarias para llevar a cabo estos dos tipos de tareas: `at` y `cron`, asociados a dos demonios; `atd` y `crond` respectivamente, que deben estar en funcionamiento para que esto sea posible. No es necesario reiniciar estos servicios tras realizar algún cambio porque ellos mismos comprueban periódicamente si tienen tareas a realizar.

at

Este comando permite utilizar la línea de comandos para configurar un evento individual. Tanto el administrador del sistema como los usuarios tienen acceso a esta herramienta.

El comando “`at`” dispone de una jerarquía de colas; cada una de estas colas se refiere a una prioridad o la cantidad de tiempo de procesamiento de la máquina que se le permite ocupar a ese trabajo. Los niveles están nombrados con un único carácter, que puede ir desde la “a” hasta la “z”, o desde la “A” a la “Z”. Cuanto más avanzada sea la letra en minúscula desde el principio del abecedario, menos tiempo se le permite ocupar. Si se utilizan las mayúsculas ocurre lo mismo, pero sólo se ejecutarán si la media de la carga de la CPU es inferior a 0.8 (al 80%).

Para configurar quién puede hacer uso de “`at`” se utilizan dos archivos; `at.allow` y `at.deny`; el funcionamiento es el siguiente: si existe `at.allow`, sólo los usuarios cuyos nombres se encuentren en él podrán utilizar el comando (el superusuario siempre podrá usarlo); si no existe, se comprueba `at.deny`, que designa los usuarios que no pueden utilizar “`at`”. Los nombres de usuario deben aparecer cada uno en una línea.

Para utilizar `at` la sintaxis es: `at` opciones fecha.

Algunas de las opciones disponibles son:

- c envía los comandos a la salida estándar.
- f coge la entrada del archivo especificado.
- m envía correo electrónico al usuario original cuando el trabajo se ha completado.
- q asigna una cola específica; la cola "=" se refiere a los trabajos que se están ejecutando en ese momento.
- V proporciona el número de versión de at.

Existen muchas formas de decirle a at cuándo queremos que ejecute un comando; las más usadas sin embargo son:

hh:mm hora y minuto
mmddyy mes, día y año; pueden ir separados por "/" o por "."

Ambas pueden combinarse para especificar una fecha completa.

Por ejemplo para crear un trabajo con prioridad "c" a las 15:30 del 15 de Agosto de 2004 debería ponerse:

```
at -q c 15:30 08/15/2004
```

Existe otra opción muy útil, "now + X unidades", que cuenta a partir de ahora X unidades (minutes, hours, days). Para ejecutar el comando dentro de 5 minutos:

```
at now + 5 minutes
```

Una vez puestas las opciones entraríamos en la línea de comando propia de at; todo lo que se escriba ahora y hasta que se pulse "Control-D" se considera parte del trabajo. Hay que tener en cuenta que sólo hay que poner comandos de consola, ya que las órdenes se ejecutarán con el intérprete de comandos /bin/sh.

Para ver los trabajos que están en la cola se utiliza atq; en primer lugar aparece el número de trabajo, seguido de la fecha y hora de ejecución, el nombre de la cola y el usuario a quien pertenece el trabajo.

Si se quiere eliminar un trabajo antes de que se lance, hay que utilizar atrm número_de_trabajo, donde éste número es el que muestra atq.

cron

Se utiliza para gestionar comandos que se ejecutan más de una vez; es una herramienta más completa que at y la forma de utilizarlo depende del usuario que lo haga. Veremos primeramente cómo lo usa root.

El archivo principal de cron es /etc/crontab; en él se define cuándo se ejecutan las tareas, teniendo en cuenta que se pueden configurar cada hora (hourly), día (daily), semana (weekly) o mes (monthly). Cada una de ellas posee un directorio propio denominado cron.periodo, siendo periodo uno de los anteriores. En estos directorios se encuentran los scripts que serán ejecutados.

El comando cron tiene una forma particular de representar el tiempo, distinta de la de at. Una sentencia de tiempo cron se divide en 5 partes: A H D M W; cada uno de estos elementos se representan numéricamente. Su significado es:

- A Especifica el número de minutos después de la hora; puede valer de 0 a 59.
- H Especifica la hora, en formato de 24 horas; vale de 0 a 23.
- D Especifica la fecha (día del mes); vale de 0 a 31.
- M Especifica el mes del año; vale de 1 a 12 o las 3 primeras letras del nombre del mes.
- W Especifica el día de la semana; vale de 0 (domingo) a 6 (sábado) o las tres primeras letras del nombre del día.

También se pueden utilizar unos modificadores; existen diferentes operadores que puede utilizar para expresar rangos y otros elementos que le permiten expresar valores de tiempo múltiples para su trabajo de cron:

- * expresa que todos los valores son posibles
- especifica un rango de valores (incluyendo a los que aparecen
- , valores individuales
- / especifica el incremento que debería seguir un grupo de valores en lugar de aumentar en uno

Puedes definir tu propia periodicidad, para lo que deberás crear un directorio, aunque realmente no es necesario ya que las 4 entradas que hay en crontab cubren prácticamente todas las necesidades.

Para crear un trabajo de cron como usuario (incluido el superusuario cuando el trabajo no es para el sistema), hay que ejecutar el comando:

`crontab -e`

Así abriremos el archivo de datos crontab para el usuario (si es la primera vez estará vacío); el formato de éste archivo es similar al que vimos anteriormente, salvo que no hay que especificar el nombre de usuario (ya que cada usuario posee el suyo propio).

Si hubo algún error o apareció alguna información al ejecutar un comando, tanto con at como con cron, puede consultarse revisando el correo del usuario.

Ejercicios

- 1) Seguramente tengas una impresora funcionando en tu sistema; prueba a detener el servicio llamado “cupsys” e intenta imprimir cualquier cosa; hasta que no actives de nuevo el servicio no podrás hacerlo.
- 2) Activa si no lo están ya los servicios atd y crond, para poder realizar los ejercicios siguientes.
- 3) Realiza una copia de seguridad programada para dentro de 5 minutos del archivo /etc/hostname en el directorio de root.
- 4) Observa que está programado correctamente.
- 5) Programa una copia del archivo /etc/hostname como antes pero para que se realice cada viernes a las 23:45.
- 6) Como usuario alumno crea una copia de seguridad programada de tu carpeta “Documents” cada 13 de cada mes a las 8:30 de la mañana; el comando es “tar cfz copia.tar.gz Documents”

Soluciones

- 3) El comando es: `at now +5 minutes`; una vez dentro de la interfaz la orden es: `cp`

/etc/hostname /root.

4) Para ver los comandos programados: atq, y observa

5) Introduce al final del archivo /etc/crontab la línea:

```
45 23 * * 6 root cp /etc/hostname /root
```

6) Ejecuta "crontab -e" y escribe la línea:

```
30 8 13 * * tar cfz copia.tar.gz Documents
```

CAPÍTULO 4

Manipulación de Paquetes Debian

Los paquetes son archivos que contienen programas y aplicaciones que podemos instalar en nuestros sistemas. Debian tiene un formato propio de paquetes, reconocidos por la extensión `.deb`. Existen básicamente dos tipos de herramientas para manipular paquetes: unas de bajo nivel (como `dpkg`), que manipulan paquetes individuales y otras de más alto nivel (`apt`, `dselect`) que lo hacen a través de una base de datos previamente creada. Estas últimas son las más convenientes, ya que resuelven, de forma automática, las posibles dependencias y conflictos entre paquetes, haciendo más fácil su instalación y gestión. En esta sección describiremos cómo emplear estas herramientas para realizar las tareas más comunes relacionadas con los paquetes.

Ya conoces la herramienta gráfica para la instalación y actualización de paquetes: `synaptic`, localizada en Aplicaciones -> Panel de Control. Ahora veremos las otras herramientas disponibles.

dselect

Es una interfaz para manejar `dpkg`, una herramienta que veremos más adelante. Para verla abre un terminal y ejecuta “`dselect`”, verás la imagen siguiente.

Para moverte por las opciones puedes utilizar las flechas de cursores, o puedes pulsar la letra que se encuentran entre corchetes ([]); para seleccionar una opción debes pulsar la tecla “enter”.

Los pasos numerados indican el orden en que deben realizarse las operaciones; en la primera debemos escoger el método de adquisición de nuevos paquetes; está seleccionado “apt”, y no debería cambiarse.

A continuación hay que actualizar la lista de paquetes para que podamos disponer de los más recientes. Una vez actualizada la lista, procedemos a seleccionar el paso 2 y pulsar “enter” para elegir los paquetes que queremos instalar. Se nos muestra una lista muy extensa pero organizada por temas, con los nombres de los paquetes, la versión instalada, la versión más actual disponible para actualizar y una descripción; si no puedes verla prueba a pulsar la flecha derecha, aunque se muestra entera en la parte inferior de la pantalla.

Para seleccionar un paquete hay que posicionarse sobre él y pulsar la tecla “+”, quedando marcado para instalarlo (o actualizarlo a una versión más nueva); si por el contrario queremos desinstalarlo pulsaremos “-”. Es posible que al seleccionar un paquete nos avise que es necesario seleccionar otros para resolver conflictos de dependencias; no te preocupes, dselect los selecciona por nosotros, y además nos propone algunos que, aunque no son esenciales, pueden completar el paquete que hemos seleccionado (un paquete de idiomas, por ejemplo). Para salir de esta ventana pulsa “enter”.

```

Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
dselect - lista principal de paquetes (disp., prio marca:+/=- literal:v ayuda:?)
EIOm Pri Sección Paquete Ver.inst. Ver.disp. Descripción
*** Req base base-files 3.0.12 3.0.12 Debian base system miscel
*** Req base base-passwd 3.5.5 3.5.5 Debian base system master
*** Req base bash 2.05b-12 2.05b-12 The GNU Bourne Again SHel
*** Req base bsdutils 2.12-6 2.12-6 Basic utilities from 4.4
*** Req base coreutils 5.0.91-2 5.0.91-2 The GNU core utilities
*** Req base debianutils 2.6.2 2.6.2 Miscellaneous utilities s
*** Req base diff 2.8.4-0.0 2.8.4-0.0 File comparison utilitie
*** Req base dpkg 1.10.18 1.10.18 Package maintenance syste
*** Req base dselect 1.10.18 1.10.18 a user tool to manage Deb
*** Req base e2fsprogs 1.34+1.35-W 1.34+1.35-W The EXT2 file system util
dpkg - instalado ; instalar (era: instalar). requeridos
dpkg - Package maintenance system for Debian

This package contains the programs which handle the installation and
removal of packages on your system.

The primary interface for the dpkg suite is the 'dselect' program; a more
low-level and less user-friendly interface is available in the form of the
'dpkg' command.

In order to unpack and build Debian source packages you will need to
descripción de dpkg -- 90%, pulse d para seguir.[]

```

Una vez seleccionados todos los paquetes que queremos instalar, volvemos a pulsar “enter”, para entrar en el paso “3. [I]nstalar”. Después de pedirnos confirmación los paquetes son descargados e instalados. Una vez que este proceso ha terminado sólo falta realizar una serie de configuraciones para que el programa o programas estén preparados para usarse. El proceso termina seleccionando la desinstalación de los paquetes que hubiéramos marcado para ello.

Si en algún momento te pierdes y te entra el pánico, no te preocupes, estas tres teclas te sacarán del problema:

- u: vuelve a las selecciones hechas por dselect
- r: cancela las selecciones hechas por nosotros en la pantalla actual
- x: descarta todos los cambios y sale de dselect.

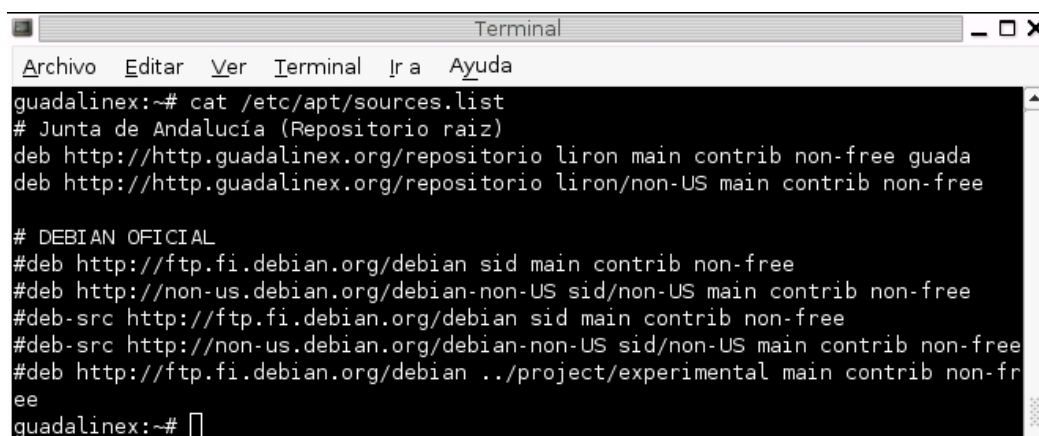
Como puedes ver la instalación de paquetes mediante dselect es una tarea un poco pesada ya que moverse por la lista resulta un tanto incómodo; puedes usar las teclas avanzar-página y retroceder-página (AvPag y RePag) , o realizar una búsqueda pulsando “/” e introduciendo la palabra a buscar (la búsqueda se realiza sólo en los nombres de los paquetes, no en la descripción), pero aún así es una tarea lenta. Esta herramienta existe desde hace tiempo y se mantiene porque

hay gente que está acostumbrada a utilizarla; aún así, es lo más parecido a synaptic que tenemos cuando no disponemos del entorno gráfico.

apt

Es un conjunto de herramientas avanzadas de gestión de paquetes; hace desde la consola lo que synaptic hace desde una ventana gráfica; en realidad muchas cosas más, porque synaptic simplemente es una herramienta gráfica basada en un subconjunto de comandos de apt. Los nombres, versiones y descripciones de los paquetes disponibles se guardan en una base de datos en el directorio `/var/cache/apt` para ser consultada en cualquier momento.

Lo primero que hay que hacer antes de usar apt y los programas que dependen de él es indicarle dónde debe buscar los paquetes Debian para crear su base de datos. Las fuentes de paquetes se indican en el archivo `/etc/apt/sources.list`; estas fuentes se denominan repositorios.



```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# cat /etc/apt/sources.list
# Junta de Andalucía (Repositorio raíz)
deb http://http.guadalinux.org/repositorio liron main contrib non-free guada
deb http://http.guadalinux.org/repositorio liron/non-US main contrib non-free

# DEBIAN OFICIAL
#deb http://ftp.fi.debian.org/debian sid main contrib non-free
#deb http://non-us.debian.org/debian-non-US sid/non-US main contrib non-free
#deb-src http://ftp.fi.debian.org/debian sid main contrib non-free
#deb-src http://non-us.debian.org/debian-non-US sid/non-US main contrib non-free
#deb http://ftp.fi.debian.org/debian ../project/experimental main contrib non-free
guadalinux:~#
```

Las líneas que comienzan con “#” se ignoran. En la imagen sólo están activados dos repositorios, correspondientes a los oficiales del proyecto GuadaLinux. El resto corresponden a unos repositorios oficiales de Debian. La estructura es siempre la misma: la línea empieza por “deb”, seguido de la URL del repositorio, y a continuación los directorios en los que se buscarán paquetes.

Alternativamente se puede añadir un CD con paquetes (por ejemplo los de la instalación oficial de Debian); éstas no se introducen manualmente, sino que se utiliza el comando:

```
# apt-cdrom add
```

Y automáticamente se añaden a la base de datos los paquetes del CD.

El sistema apt incluye potentes herramientas que permiten actualizar una gran cantidad de paquetes simultáneamente e incluso renovar la distribución completamente. Para actualizar la lista de paquetes de todos los repositorios se puede utilizar la orden:

```
# apt-get update
```

Y para actualizar a las nuevas versiones todos los paquetes de los que exista una versión más reciente:

```
# apt-get upgrade
```

Si la actualización supone un cambio a una versión completamente nueva de la distribución necesitamos otra opción capaz de manejar la actualización simultánea de todos los paquetes de la distribución:

```
# apt-get dist-upgrade
```

Ten en cuenta que la lista de paquetes a instalar en estas dos últimas operaciones puede ser bastante larga, por lo que puede conllevar la descarga de mucha cantidad de datos.

La instalación y desinstalación de paquetes con apt es extraordinariamente sencilla si se conoce el nombre del paquete a instalar. Basta con ejecutar el comando apt-get con el argumento "install" y la lista de paquetes a instalar; apt instalará los paquetes seleccionados y cualquier otro del que estos dependan, resolviendo todas las dependencias. Para la instalación considerará todas las fuentes listadas en el archivo sources.list y obtendrá las versiones más actualizadas de los paquetes, realizando las conexiones pertinentes y solicitando los CD que se necesiten. En caso de encontrar un paquete en varias fuentes, utilizará aquella que se especifique en primer lugar en el fichero. Por ejemplo si queremos instalar el paquete fortune-es:

```
# apt-get install fortune-es
```

Si estamos interesados en descargar un paquete (por ejemplo el fortunes-es) pero no queremos instalarlo, tenemos que incluir la opción “-d”, de esta forma:

```
# apt-get -d install fortunes-es
```

Para desinstalar paquetes se emplea la opción “remove” de apt-get. Por defecto se conservan los ficheros de configuración del paquete que se desinstala, pero se puede indicar que se borren con la opción “--purge”. Para desinstalar el paquete fortune-es sin eliminar los archivos de configuración:

```
# apt-get remove fortune-es
```

Y si quisiéramos eliminar también los archivos de configuración:

```
# apt-get --purge remove fortune-es
```

Los paquetes descargados se guardan por defecto en /var/cache/apt/archives, por si se necesita utilizarlos en otro ordenador, por ejemplo. Como los paquetes descargados pueden ocupar bastante espacio puede ser aconsejable eliminar los paquetes que ya se han instalado y no son necesarios; esto puede hacerse mediante el comando:

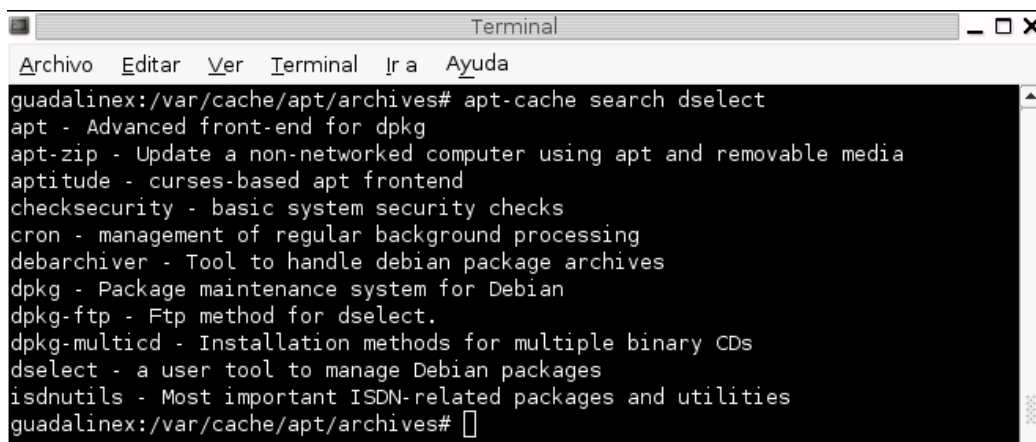
```
# apt-get clean
```

Existe un último programa disponible referente a apt, llamado apt-cache, que permite obtener información diversa sobre los paquetes y permite realizar búsquedas, tanto en los nombres como en la descripciones. Para ver la información referente a un paquete (nombre ,versión ,dependencias, descripción, ...) se utiliza el parámetro “show”:

```
# apt-cache show dselect
```

Si por el contrario queremos buscar todos los paquetes relacionados con “dselect” usaríamos:

```
# apt-cache search dselect
```

A terminal window titled "Terminal" with a menu bar containing "Archivo", "Editar", "Ver", "Terminal", "Ir a", and "Ayuda". The terminal shows the command "apt-cache search dselect" and its output, which lists various Debian packages and their descriptions, all containing the word "dselect". The packages listed are: apt, apt-zip, aptitude, checksecurity, cron, debarchiver, dpkg, dpkg-ftp, dpkg-multicd, dselect, isdnutils, and guadalinux. The prompt is "guadalinux:/var/cache/apt/archives#".

```
guadalinux:/var/cache/apt/archives# apt-cache search dselect
apt - Advanced front-end for dpkg
apt-zip - Update a non-networked computer using apt and removable media
aptitude - curses-based apt frontend
checksecurity - basic system security checks
cron - management of regular background processing
debarchiver - Tool to handle debian package archives
dpkg - Package maintenance system for Debian
dpkg-ftp - Ftp method for dselect.
dpkg-multicd - Installation methods for multiple binary CDs
dselect - a user tool to manage Debian packages
isdnutils - Most important ISDN-related packages and utilities
guadalinux:/var/cache/apt/archives#
```

En la imagen pueden verse todos los paquetes disponibles que, en el nombre o en su descripción, contienen la palabra “dselect”; aparece el paquete del propio programa, pero también otros que necesita para funcionar, como apt o dpkg.

dpkg

Es una herramienta que permite la manipulación directa de ficheros “.deb”. Gracias a la existencia de apt, sólo es necesario recurrir a ella cuando nos encontramos con paquetes sueltos, es decir, que no forman parte de ningún repositorio. También es útil si hemos usado apt-get con la opción “-d” y llevamos los paquetes descargados a otro ordenador para instalarlos.

Para instalar un paquete llamado “paquete.deb” la forma de hacerlo es:

```
# dpkg --install paquete.deb
```

En el caso de que se produzcan problemas de dependencias el paquete no será configurado pero sí desempaquetado; es necesario instalar los paquetes para resolver estas dependencias de otros paquetes. Si ese es el caso, una vez instalados todos hay que ejecutar:

```
# dpkg --configure --pending
```

De esta forma se configuran los paquetes pendientes de configuración.

Hay que ser muy cuidadoso con los paquetes instalados directamente con dpkg, ya que la lista de dependencias puede ser tan grande que haga inviable la instalación a mano (por eso se creó apt).

Para desinstalar paquetes se emplea:

```
# dpkg --remove paquete
```

Y si también se quieren eliminar los ficheros de configuración:

```
# dpkg --purge paquete
```

Fíjate que en ambos casos hay que poner el nombre del paquete, no el nombre del archivo .deb.

Ejercicios

- 1) Actualiza la base de datos de los paquetes de apt.
- 2) Busca todos los paquetes que contengan la palabra “guadalinux”.
- 3) Instala un parche para solucionar un problema con la hora.
- 4) Bájate (pero no instales) el paquete fortune para Guadalinux.
- 5) Localiza el archivo anterior e instálalo con dpkg
- 6) Elimina el paquete fortunes-guadalinux

Soluciones

- 1) apt-get update
- 2) apt-cache search guadalinux

- 3) `apt-get install guadalinux-hora`
- 4) `apt-get -d install fortunes-guadalinux`
- 5) `dpkg -i /var/cache/apt/archives/fortunes-guadalinux-1.0.2.deb`
- 6) `apt-get remove fortunes-guadalinux`

CAPÍTULO 5

Sistemas de Ficheros

Crear un sistema de ficheros

Un sistema de ficheros es una forma de organizar los datos en un dispositivo físico; este dispositivo físico puede ser cualquier dispositivo de almacenamiento: un disquete, un disco duro, un CD-ROM, un llavero USB, ... En un mismo dispositivo pueden coexistir distintos sistemas de ficheros, aunque por lo general será uno sólo (a excepción de los discos duros).

Lamentablemente la manipulación de los sistemas de ficheros necesita realizarse desde una consola; existe una herramienta gráfica llamada “qtparted”, pero se limita a manipular particiones en discos duros (lo que por otro lado viene muy bien en algunos casos, por ejemplo a la hora de instalar).

En Linux, la mayoría de los dispositivos físicos (discos, impresoras, módems, ...) están representados por ficheros especiales de tipo “device” (o dispositivo); estos se encuentran dentro del directorio /dev del sistema y permiten acceder directamente al hardware que representan, haciendo operaciones de lectura, escritura y control sobre dichos dispositivos; pero para que nosotros podamos utilizar estos elementos necesitamos crear un sistema de ficheros, que no es otra cosa que crear en el dispositivo unas estructuras especiales que nos permitan leer y escribir la información en ese dispositivo.

ADVERTENCIA: Cuando se crea un sistema de ficheros sobre un dispositivo, todo el contenido de dicho dispositivo se elimina y no se puede recuperar de ninguna forma (aunque existen algunas excepciones a esto). Ten esto en cuenta siempre presente porque no querrás comprobarlo por ti mismo.

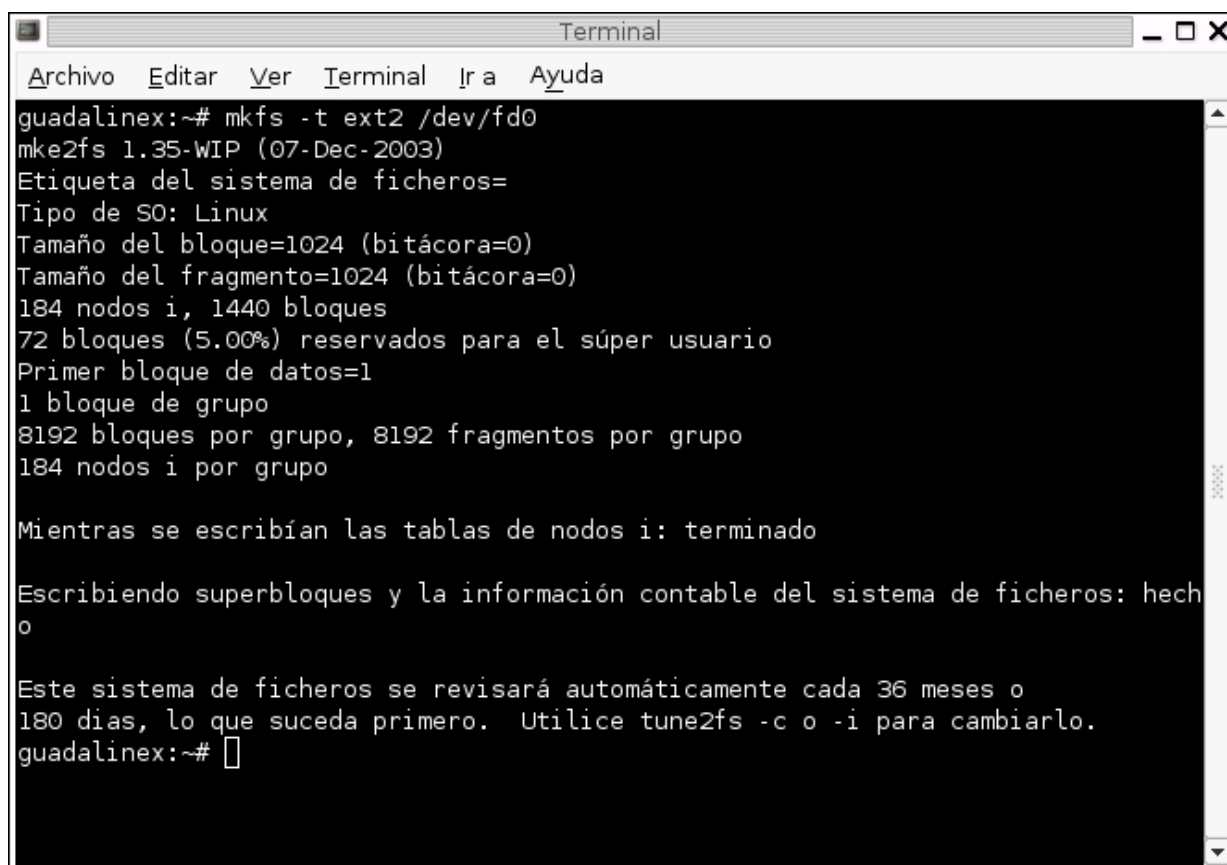
Existe una gran variedad de sistemas de ficheros, aunque en la práctica se usan sólo unos pocos; normalmente cada sistema operativo define uno o varios de ellos, que son los que utiliza para instalar dicho sistema o guardar sus datos. Los más

utilizados en la actualidad son:

- **ext2:** es un sistema de ficheros creado expresamente para ser utilizado por Linux; ofrece el mejor rendimiento (en términos de velocidad y utilización de CPU) de todos los sistemas bajo Linux, permite protección de archivos por usuarios y grupos, no es necesario defragmentarlo, ...
- **ext3:** es una versión de ext2 al que se le ha añadido un registro de transacciones (denominado “bitácora” o “journal”); esta bitácora es un registro donde se almacenan las operaciones que se realizan en el sistema de ficheros, de manera que ante una interrupción inesperada (por ejemplo un apagón) se evite en la medida de lo posible la pérdida de datos.
- **reiserfs:** otro sistema con bitácora, pero diseñado partiendo desde cero. Para manejarlas se necesita tener instalado el paquete “reiserfsprogs”.
- **xfs:** sistema de ficheros orientado a proporcionar alto rendimiento y escalabilidad (pensado para servidores). Para manejarlas es necesario instalar el paquete “xfsprogs”.
- **swap:** utilizado como memoria de intercambio o paginación en Linux.
- **msdos:** es el sistema usado en DOS y algunas versiones antiguas de Windows; los nombres de archivos están limitados a 8 caracteres, pudiendo estar seguidos de un punto y tres caracteres más. Está en desuso.
- **vfat:** es una extensión de msdos para soportar nombres largos de archivos que se utiliza en Windows 95 o NT y versiones superiores.
- **iso9660:** es un estándar utilizado primordialmente en los CD-ROM.
- **nfs:** sistema de ficheros de red utilizado para acceder discos localizados en máquina remotas.
- **smb:** es un sistema que soporta el protocolo SMB, utilizado en ordenadores Windows para compartir elementos por la red.

Para crear un sistema de ficheros hay que utilizar el comando “mkfs”; lo único que hay que hacer es saber cómo nombra Linux a ese dispositivo y decidir qué tipo de sistema de ficheros vamos a crear en él.

Coge un disquete que vamos a crear en él un sistema de ficheros; en Linux se suelen representar por /dev/fd0 (o /dev/fd1), así que para formatear un disquete con ext2 simplemente hay que escribir: “mkfs -t ext2 /dev/fd0”, como se muestra en la imagen (la opción “-t” es para especificar el tipo de sistema de ficheros que queremos). Recuerda que esto borrará todo el contenido del disquete.



```
Terminal
Archivo Editar Ver Terminal Ir a Ayuda
guadalinux:~# mkfs -t ext2 /dev/fd0
mke2fs 1.35-WIP (07-Dec-2003)
Etiqueta del sistema de ficheros=
Tipo de SO: Linux
Tamaño del bloque=1024 (bitácora=0)
Tamaño del fragmento=1024 (bitácora=0)
184 nodos i, 1440 bloques
72 bloques (5.00%) reservados para el súper usuario
Primer bloque de datos=1
1 bloque de grupo
8192 bloques por grupo, 8192 fragmentos por grupo
184 nodos i por grupo

Mientras se escribían las tablas de nodos i: terminado

Escribiendo superbloques y la información contable del sistema de ficheros: hecho
o

Este sistema de ficheros se revisará automáticamente cada 36 meses o
180 días, lo que suceda primero. Utilice tune2fs -c o -i para cambiarlo.
guadalinux:~#
```

Se pueden crear otros tipos de sistemas en un disquete, como msdos o vfat (ext3 y reiserfs están pensados para dispositivos de mayor capacidad).

Una lista de los dispositivos más comunes son:

disquetes	/dev/fd0			
discos IDE	/dev/hda	/dev/hdb	/dev/hdc	/dev/hdd
discos SCSI	/dev/sda	/dev/sdb	/dev/sdc	
cd-rom	/dev/cdrom			
llaveros	USB	como si fueran SCSI		

El orden de los discos IDE tal y como están nombrados se corresponden respectivamente con: máster primario, esclavo primario, máster secundario y esclavo secundario. Según esto el cd-rom puede ser accedido también mediante el dispositivo correspondiente, aunque por lo general se utiliza `/dev/cdrom`.

Para crear sistemas de ficheros en discos duros (tanto IDE como SCSI) hay que especificar además la partición en la que queremos crearla; esto se hace añadiendo el número de partición al nombre del disco. Por ejemplo, la primera partición del primer disco es `/dev/hda1`, mientras que la cuarta del tercero es `/dev/hdc4`. De esta forma, para crear un sistema de ficheros tipo `ext3` en la segunda partición del segundo disco escribiríamos: `mkfs -t ext3 /dev/hdb2`. Ten mucho cuidado si estás como root de no equivocarte para no perder archivos.

Seguramente te estarás preguntando cómo de importante es la elección del sistema de ficheros para el dispositivo; pues bien, además de la velocidad va a determinar no sólo la forma de almacenar la información, sino también la forma de gestionar cómo se hace y la disponibilidad de los datos almacenados. Esto quiere decir que si aplicas el formato `ext2` a un disquete, mantendrás los atributos de propiedad y permisos de los archivos, pero no podrás leerlo desde Windows, mientras que si lo formateas con `msdos` o `vfat` podrás leerlo pero perderás esa información asociada al archivo.

Lo mismo ocurrirá con las particiones del disco duro: Windows solo podrá usar particiones `msdos` o `vfat`, mientras que Linux podrá verlas todas, aunque sólo podrá utilizar los permisos de usuarios y grupos en particiones `ext2`, `ext3`, `reiserfs` o `xfs`.

El tipo `iso9660` no plantea problemas; puede leerse tanto en Windows como en Linux. El resto de sistemas los veremos más adelante en el presente manual.

La partición de sistema

El sistema de ficheros de Linux (independientemente del sistema utilizado) se compone de un único árbol de directorios que comienza en el directorio principal (designado por `"/`"); este árbol puede estar integrado por varios dispositivos físicos (e incluso dispositivos virtuales). A cada dispositivo integrado en el árbol de directorios se accede mediante un subdirectorio común del mismo

llamado punto de montaje del dispositivo. De esta forma resulta transparente para los usuarios y las aplicaciones qué dispositivos forman el árbol de directorios y dónde están montados.

```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:/# ls -l
total 262504
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 auto
drwxr-xr-x  2 root    root      4096 2004-05-12 16:23 bin
drwxr-xr-x  3 root    root      4096 2004-05-22 17:39 boot
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 cdrom
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 cdrom1
drwxr-xr-x  8 root    root     24576 2004-05-22 15:26 dev
drwxr-xr-x 95 root    root     8192 2004-05-22 18:59 etc
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 floppy
drwxrwsr-x  3 root    staff    4096 2004-05-21 19:49 home
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 initrd
drwxr-xr-x  6 root    root      4096 2004-05-12 16:19 lib
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 lost+found
drwxr-xr-x 10 root    root      4096 2004-05-16 02:13 mnt
drwxr-xr-x  2 root    root      4096 2004-05-12 16:19 opt
dr-xr-xr-x 98 root    root        0 2004-05-22 17:25 proc
drwxr-xr-x 10 root    root      4096 2004-05-22 18:59 root
drwxr-xr-x  2 root    root      4096 2004-05-22 13:20 sbin
-rw-r--r--  1 root    root    268435456 2004-05-12 16:00 swapfile
drwxrwxrwt 12 root    root      4096 2004-05-22 19:04 tmp
drwxr-xr-x 13 root    root      4096 2004-05-12 16:19 usr
drwxr-xr-x 16 root    root      4096 2004-05-12 16:19 var
lrwxrwxrwx  1 root    root        19 2004-05-12 16:04 vmlinuz -> boot/vmlinuz-2.4.23
guadalinux:/#
```

En la imagen se muestra el directorio raíz típico; este sistema cumple con una organización llamada FHS (Filesystem Hierarchy Standard, estándar jerárquico para el sistema de ficheros); los diferentes subdirectorios se suelen utilizar para tareas específicas:

/bin: aquí están los ejecutables necesarios para hacer funcionar el sistema y que tienen utilidad para todos los usuarios.

/boot: contiene los ficheros y datos necesarios para arrancar el sistema.

/cdrom y */floppy*: puntos de montaje del CD-ROM y la disquetera.

/dev: ficheros especiales que representan dispositivos hardware.

/etc: contiene los ficheros de configuración de diferentes programas.

/home: directorios de inicio de los usuarios.

/lib: librerías dinámicas esenciales del sistema.

/lost+found: aquí es donde aparecen los archivos o trozos recuperados tras ocurrir algún accidente; es una característica del sistema ext2.

/mnt: punto de montado transitorio de sistemas de ficheros.

/proc: sistema de ficheros virtual generado por el núcleo y que permite obtener y enviar información al hardware.

/root: directorio de inicio del superusuario.

/sbin: ejecutables de interés únicamente para el administrador.

/tmp: datos temporales empleados por diversos programas.

Hay dos directorios importante que nos queda por ver: */usr* y */var*; ambos tienen en común que no son esenciales para el funcionamiento del sistema, pero se diferencian en que mientras */usr* permanece invariable (salvo cuando se instalan programas), */var* cambia durante la operación normal del sistema. Por eso suelen residir en particiones independientes (y */usr* en modo sólo lectura). En ellos encontramos los siguientes subdirectorios:

/usr/X11R6: estructura similar a */usr* pero utilizado por los programas y componentes del entorno gráfico X-Window.

/usr/bin, */usr/sbin* y */usr/lib*: hacen la misma función que */bin*, */sbin* y */lib*, pero para programas que no son indispensables para el arranque y funcionamiento del sistema. Los programas que aquí se encuentran son comunes a todas las distribuciones del mismo grupo (todas las basadas en Debian, por ejemplo) mientras que los de */bin*, */sbin* y */lib* son comunes a todos los Linux.

/usr/doc: información adicional suministrada por algunos programas.

/usr/local: repite la estructura de */usr*, pero para programas y componentes particulares de nuestro sistema (que no estarán en otra Debian).

/var/log: ficheros que registran información generada por diversos procesos

del sistema, como mensajes de error.

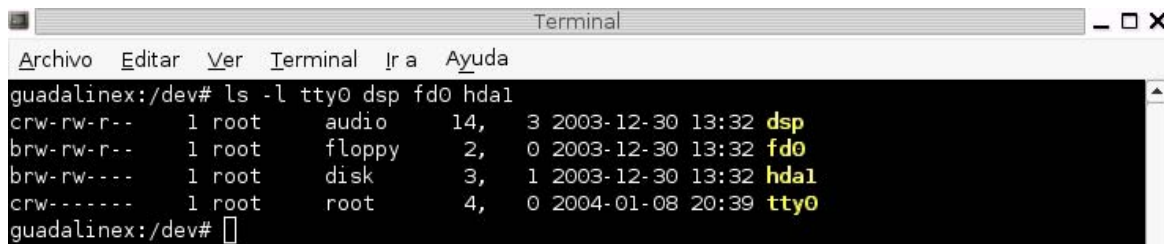
/var/spool: trabajos pendientes de realización, como por ejemplo las colas de impresión (*/var/spool/cups*) y los buzones correo (*/var/spool/exim*).

Existen más directorios, pero los principales los hemos expuesto aquí.

Ficheros especiales

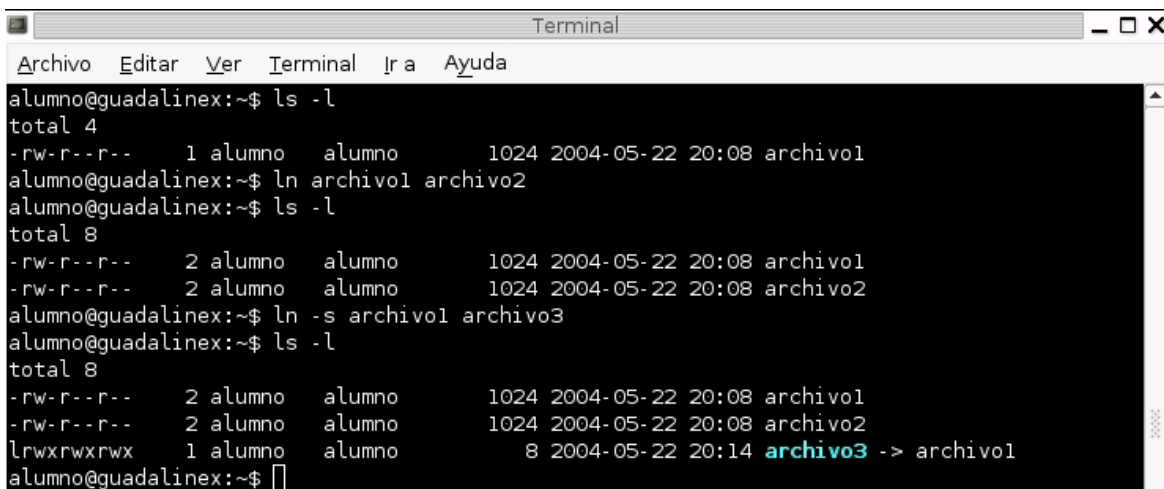
En Linux, aparte de los directorios y ficheros convencionales, existen una serie de ficheros “especiales”. Un grupo lo constituyen los ficheros de dispositivos que nombramos anteriormente, que representan ciertos dispositivos físicos. Los hay de dos tipos: de tipo carácter (marcados con una *c*) y de tipo bloque (marcados con una *b*). Con los de tipo carácter la transferencia se realiza byte a byte, mientras que con los de tipo bloque la transferencia se realiza por bloques completos de un determinado tamaño.

En la imagen puedes ver que los dispositivos “*dsp*” y “*tty0*” son de tipo carácter (se corresponden con el procesador digital de la tarjeta de sonido y la primera consola virtual, respectivamente), mientras que “*fd0*” y “*hda1*” (la disquetera y la primera partición del primer disco IDE) son de tipo bloque.



```
Terminal
Archivo Editar Ver Terminal Ir a Ayuda
guadalinux:/dev# ls -l tty0 dsp fd0 hda1
crw-rw-r-- 1 root audio 14, 3 2003-12-30 13:32 dsp
brw-rw-r-- 1 root floppy 2, 0 2003-12-30 13:32 fd0
brw-rw---- 1 root disk 3, 1 2003-12-30 13:32 hda1
crw----- 1 root root 4, 0 2004-01-08 20:39 tty0
guadalinux:/dev#
```

Existen otros tipos de ficheros, denominados enlaces; los hay también de dos tipos, duros y blandos. Los enlaces duros resultan en que un mismo fichero aparezca más de una vez, es decir, aparezca en dos sitios a la vez, pero haciendo referencia a los mismo datos. Sin embargo los enlaces simbólicos indican que los accesos deben redirigirse a otro fichero diferente. Ambos se crean con el comando “*ln*”, añadiendo el parámetro “*-s*” para el caso del enlace simbólico. Los duros



```

Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
alumno@guadalinux:~$ ls -l
total 4
-rw-r--r--  1 alumno  alumno      1024 2004-05-22 20:08 archivo1
alumno@guadalinux:~$ ln archivo1 archivo2
alumno@guadalinux:~$ ls -l
total 8
-rw-r--r--  2 alumno  alumno      1024 2004-05-22 20:08 archivo1
-rw-r--r--  2 alumno  alumno      1024 2004-05-22 20:08 archivo2
alumno@guadalinux:~$ ln -s archivo1 archivo3
alumno@guadalinux:~$ ls -l
total 8
-rw-r--r--  2 alumno  alumno      1024 2004-05-22 20:08 archivo1
-rw-r--r--  2 alumno  alumno      1024 2004-05-22 20:08 archivo2
lrwxrwxrwx  1 alumno  alumno        8 2004-05-22 20:14 archivo3 -> archivo1
alumno@guadalinux:~$

```

están restringidos al mismo sistema de ficheros (los simbólicos no).

En la imagen puedes ver cómo al fichero “archivo1” se le ha creado un enlace duro primero y simbólico después. El enlace duro aparece de forma idéntica a como lo hace el original; aunque aparezca con su mismo tamaño, la información no está duplicada, y de hecho podemos eliminar uno de los dos, y el otro seguiría teniendo la información original; la información no se elimina realmente hasta que no se borra el último enlace duro (con el comando “rm”). El número delante del usuario propietario del archivo indica el número de enlaces duros que posee.

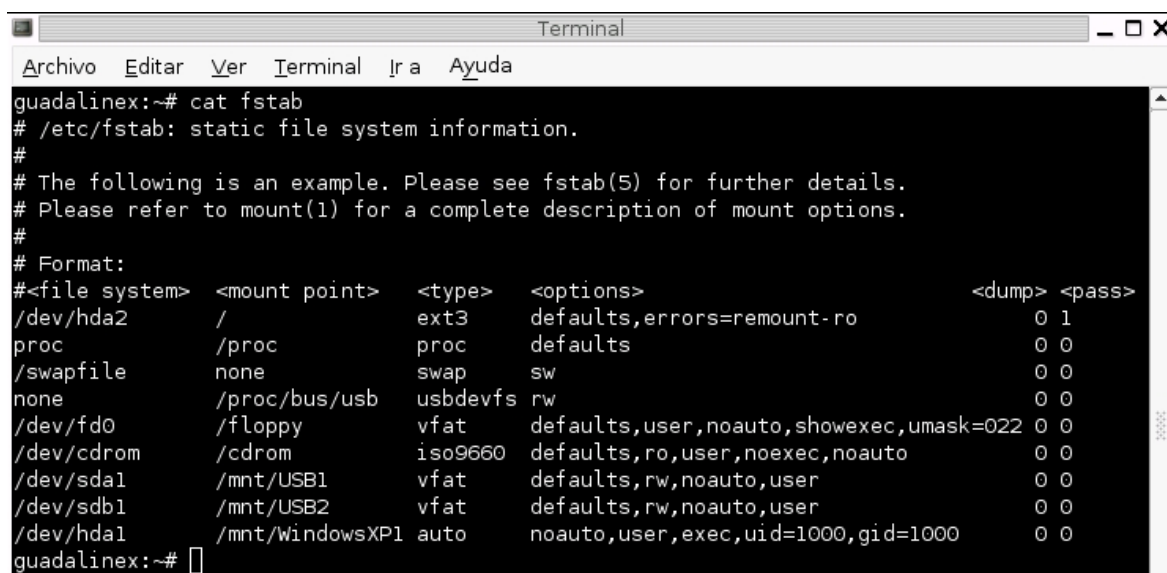
Sin embargo el simbólico aparece de forma distinta; la “l” del principio lo identifica como lo que es, y ocupa un pequeño espacio en disco, independientemente del tamaño del archivo al que apunta; dicho archivo se muestra marcado con el símbolo “->”, y representa el archivo que hay que buscar cuando se acceda al fichero “archivo3”. Los enlaces simbólicos se borran con el comando “unlink”.

Montar y desmontar dispositivos

Antes de poder utilizar un sistema de ficheros es necesario realizar una operación conocida como “montado”; igualmente antes de extraerlo de la unidad correspondiente hay que “desmontarlo”. De esta forma el sistema es informado de la presencia de la unidad y puede optimizar la transferencia de datos con él.

Para montar una partición hay que escribir: “mount -t tipo dispositivo directorio”, siendo “tipo” el tipo de sistema de ficheros, “dispositivo” su nombre en /dev y directorio el punto de montaje. Pero seguramente ya conozcas que desde el entorno gráfico es posible montar y desmontar ciertas unidades, pero no es posible configurarlas ni añadir unidades nuevas. Pulsa con el botón derecho sobre una zona vacía del escritorio y selecciona la opción “Discos”, verás la lista de unidades disponibles (no intentes montar ninguna que esté vacía porque dará un error).

Esa lista que aparece se controla desde un archivo, en concreto el /etc/fstab, que puedes ver en la imagen siguiente.



```

guadalinux:~# cat fstab
# /etc/fstab: static file system information.
#
# The following is an example. Please see fstab(5) for further details.
# Please refer to mount(1) for a complete description of mount options.
#
# Format:
#<file system> <mount point> <type> <options> <dump> <pass>
/dev/hda2 / ext3 defaults,errors=remount-ro 0 1
proc /proc proc defaults 0 0
/swapfile none swap sw 0 0
none /proc/bus/usb usbdevfs rw 0 0
/dev/fd0 /floppy vfat defaults,user,noauto,showexec,umask=022 0 0
/dev/cdrom /cdrom iso9660 defaults,ro,user,noexec,noauto 0 0
/dev/sda1 /mnt/USB1 vfat defaults,rw,noauto,user 0 0
/dev/sdb1 /mnt/USB2 vfat defaults,rw,noauto,user 0 0
/dev/hda1 /mnt/WindowsXP1 auto noauto,user,exec,uid=1000,gid=1000 0 0
guadalinux:~#

```

Cada fila representa una unidad diferente; la primera columna identifica el fichero de dispositivo asociado a la unidad; por ejemplo vemos las particiones del disco duro (hda1 y hda2), la disquetera y el CD-ROM (fd0 y cdrom), entradas para dispositivos USB como llaveros, reproductores MP3, cámaras, ... (sda1 y sdb1), y un par de entradas especiales (necesarias para el archivo swap y el sistema USB).

La segunda columna indica el subdirectorio a través del cual estará disponible el contenido de la unidad o “punto de montaje”, y debe estar creado con anterioridad (aunque su contenido aparecerá como vacío hasta que se monte la unidad correspondiente).

La tercera columna define el sistema de ficheros a utilizar para acceder a la unidad, y evidentemente debería coincidir con el que realmente existe en la

unidad; podemos ver los ya conocidos ext3, iso9660, swap y vfat. El parámetro “auto” indica que debe autodetectarse cada vez que se vaya a proceder al montaje; en este caso se realiza esta operación sobre hda1, aunque no es mala idea hacer lo mismo con la disquetera (fd0) si planeamos utilizar alternativamente discos formateados en vfat y ext2.

La siguiente columna, la de opciones, se utiliza para modificar de alguna manera el comportamiento estándar al montar la unidad. Por ejemplo el parámetro “noauto” indica que esa unidad no debe montarse automáticamente al iniciar el sistema, “user” si la puede montar un usuario, “ro” si se monta para sólo lectura, “rw” si se hace en modo lectura-escritura,... No todas las opciones son válidas para todos los sistemas de ficheros (cada uno tiene las suyas propias).

Las dos últimas columnas son utilizadas por dos herramientas, “dump” y “fsck” respectivamente, y generalmente se encuentran siempre con el valor 0, salvo para la partición de sistema, que vale 1 en la última columna.

Con este archivo configurado correctamente los dispositivos se pueden montar simplemente con “mount dispositivo”, siendo dispositivo la entrada en la primera columna.

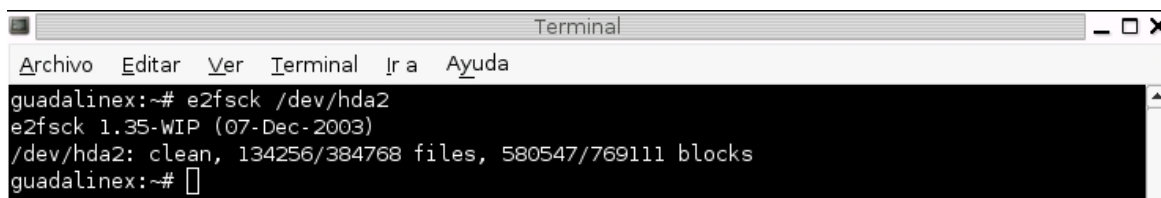
Para desmontar una unidad siempre puedes escribir “umount dispositivo” o pulsar con el botón derecho sobre él si estás en el entorno gráfico y seleccionar “Desmontar el volumen”. Asegúrate de hacerlo antes de sacar la unidad porque puedes perder algún dato en el peor de los casos estropear el sistema de ficheros de esa unidad.

Manejando ext2

A diferencia de otros sistemas operativos, Linux no posee herramientas para defragmentar el sistema de ficheros, realmente porque por las propias características de control que posee ext2 no suele ser necesario realizar este proceso.

Sin embargo sí existen herramientas de reparación que, incluso, se ejecutan automáticamente cada vez que al arrancar se detecta que en la última sesión no fueron cerrados apropiadamente. El comando para reparar el sistema de ficheros

es e2fsck, y tan sólo hay que decirle el sistema a reparar (o simplemente comprobar), en este caso /dev/hda2. Tras la comprobación se nos muestra un pequeño resumen del número de archivos y bloques respecto al máximo posible. Hay que tener en cuenta que para realizar esta comprobación el sistema de ficheros debe estar desmontado o montado como sólo lectura, ya que de otra forma podría producirse la pérdida de datos.



```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# e2fsck /dev/hda2
e2fsck 1.35-WIP (07-Dec-2003)
/dev/hda2: clean, 134256/384768 files, 580547/769111 blocks
guadalinux:~#
```

Los sistemas de archivos ext2 pueden convertirse de forma sencilla a ext3, sin tener que formatear particiones ni perder la información en ella contenida. Lo único que hay que hacer es añadir el registro de transacciones al sistema de ficheros que ya existe; tampoco es necesario desmontar el sistema de ficheros. Si la partición que se quiere convertir es hda2, la orden es:

```
# tune2fs -j /dev/hda2
```

No hay que olvidar modificar el archivo /etc/fstab para reflejar este cambio (modificar ext2 por ext3).

Este comando también puede usarse para establecer otras opciones del sistema de ficheros ext2 (o ext3), por ejemplo cada cuánto tiempo realizar una comprobación automática. Para realizar esta operación automáticamente cada vez que se monte 30 veces:

```
# tune2fs -c 30 /dev/hda2
```

Y para realizarlo cada 7 días:

```
# tune2fs -i 7d /dev/hda2
```

Si se le añade la letra “w”, la cuenta se realiza en semanas (si es la “m” en meses).

Gestión de la memoria virtual

La memoria es uno de los bienes más preciados de una máquina, ya que es en ella donde se ejecutan los programas y se almacenan temporalmente ciertos datos. Por otro lado es un recurso que aunque puede ser más o menos grande, es limitado. Esto significa que una vez agotada la memoria no se pueden ejecutar más programas.

Sin embargo los sistemas operativos desde hace tiempo proveen un mecanismo para que esta situación no ocurra (o al menos suceda con menos frecuencia): la memoria virtual. Este tipo de memoria no es memoria real, sino que se utiliza un espacio del disco duro para que haga las funciones de memoria, de manera que datos o programas que están abiertos pero no se están utilizando pasan a un segundo plano (a la memoria virtual) dejando libre la memoria principal para otro programa. Cuando estos datos vuelven a ser necesarios se intercambian con otros que ya no lo son. Aunque se pierde algo de tiempo en estos intercambios, se obtiene un mejor funcionamiento en conjunto, sobre todo si se trabaja con varias aplicaciones que necesitan una gran cantidad de memoria para funcionar.

La memoria virtual se denomina genéricamente en Linux como “swap” y generalmente se crea durante el proceso de instalación. Si más adelante se necesita mayor cantidad puede crearse otro espacio virtual.

```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
top - 21:09:38 up 5:38, 2 users, load average: 0.62, 0.41, 0.32
Tasks: 69 total, 1 running, 68 sleeping, 0 stopped, 0 zombie
Cpu(s): 26.0% user, 5.6% system, 0.0% nice, 68.4% idle
Mem: 256452k total, 251900k used, 4552k free, 9400k buffers
Swap: 377488k total, 49220k used, 328268k free, 69000k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM   TIME+  COMMAND
 607 root        5   10  101m  63m 5880 S   6.6  25.3   8:10.00 XFree86
21411 alfabet    15    0  6616  6616 5372 S   4.3   2.6   0:00.16 screenshot
 836 alfabet    16    0  9896  9296 7860 S   3.6   3.6   0:25.30 wnck-applet
 717 alfabet    15    0 72204  66m  31m S   1.3  26.4   5:46.43 soffice.bin
20998 root        16    0  1024  1024  820 R   1.0   0.4   0:00.34 top
 715 alfabet    19    0  5788  4536 4080 S   0.7   1.8   2:45.54 gkrellm
1003 root        15    0 10252  8520 7084 S   0.7   3.3   0:11.97 gnome-terminal
 647 alfabet    15    0   624   600  556 S   0.3   0.2   0:44.28 busymouse
 689 alfabet    15    0  2568  2188 2080 S   0.3   0.9   0:01.34 gnome-smproxy
 709 alfabet    16    0  6932  6356 5556 S   0.3   2.5   0:14.57 metacity
   1 root        19    0   488   460  436 S   0.0   0.2   0:18.64 init
   2 root        15    0     0     0    0 S   0.0   0.0   0:00.51 keventd
   3 root        34   19     0     0    0 S   0.0   0.0   0:00.00 ksoftirqd_CPU0
   4 root        15    0     0     0    0 S   0.0   0.0   0:00.75 kswapd
   5 root        25    0     0     0    0 S   0.0   0.0   0:00.00 bdfush
   6 root        15    0     0     0    0 S   0.0   0.0   0:00.14 kupdated
   7 root        16    0     0     0    0 S   0.0   0.0   0:00.34 kjournald
```

En la imagen puedes ver el comando “top” en acción, que monitoriza la carga del sistema; observa que tengo 256 MB de memoria, de los cuales sólo tengo libres menos de 5 MB; sin embargo tengo unos 377 MB de swap de los que se utilizan unos 50 MB, con lo que podría seguir arrancando aplicaciones sin que me saliera el mensaje “Memoria insuficiente”.

Hay dos alternativas para establecer un espacio de memoria virtual: crear una partición independiente en el disco duro, o crear un archivo en alguna partición ya existente. La primera opción resultará en un mejor rendimiento, pero la segunda es más cómoda y fácil de hacer.

Para establecer una partición como swap primero hay que crearla; puede utilizarse el programa “qtparted” a través del lanzador “Gksu” (Aplicaciones -> Herramientas del sistema”) para el usuario root.

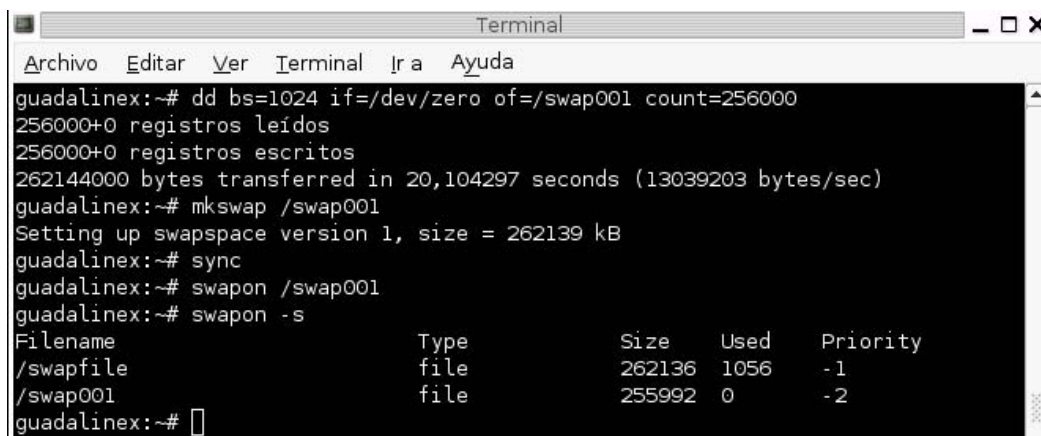
Lo único que hay que hacer es crear una partición en el espacio libre o seleccionar una que ya exista y formatearla como “Linux swap”. El tamaño puede ser cualquiera, pero se suele poner entre 1,5 y 2,5 veces la cantidad de RAM, dependiendo del uso que se le de al ordenador y la cantidad de ésta.

ADVERTENCIA: Al realizar este proceso se eliminarán todos los datos de la

partición seleccionada. Piensa dos veces la partición que vas a destinara a swap.

La otra opción es crear un archivo en la partición del sistema; la explicación excede las pretensiones de este curso, aunque los pasos son muy sencillos, como puedes ver en la imagen. Si queremos un archivo swap de unos 256 MB aproximadamente (se especifica en el parámetro “count”):

```
# dd bs=1024 if=/dev/zero of=/swap001 count=256000
# mkswap /swap001
# sync
# swapon /swap001
```



```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# dd bs=1024 if=/dev/zero of=/swap001 count=256000
256000+0 registros leídos
256000+0 registros escritos
262144000 bytes transferred in 20,104297 seconds (13039203 bytes/sec)
guadalinux:~# mkswap /swap001
Setting up swapspace version 1, size = 262139 kB
guadalinux:~# sync
guadalinux:~# swapon /swap001
guadalinux:~# swapon -s
```

Filename	Type	Size	Used	Priority
/swapfile	file	262136	1056	-1
/swap001	file	255992	0	-2

```
guadalinux:~#
```

Si ahora escribes “swapon -s” verás en la lista el nuevo espacio swap, además del que ya existía. De esta forma podríamos añadir cuantos archivos o particiones swap vayamos necesitando.

Quotas y límites

En los sistemas Linux existe la posibilidad de limitar los recursos a usuarios o grupos, por ejemplo el máximo número de logins que pueden realizar simultáneamente, el máximo tiempo de CPU, ... Estos límites se controlan a través del fichero /etc/security/limits.conf. Concretamente, en este fichero se controlan los límites sobre los procesos de un usuario.

Abre el programa “nautilus” desde el menú Aplicaciones -> Accesorios y

localiza el archivo limits.conf. Aunque el archivo está en inglés es muy fácil modificarlo; en primer lugar todas las líneas que empiezan por “#” son comentarios (es decir, se ignoran), que como puedes ver son todas (no hay establecido ningún límite todavía). Cada línea que vamos a poner tiene cuatro campos:

- domino (domain): será el nombre de usuario o de grupo (precedido por una “@”) al que le vamos a poner algún tipo de restricción; el “*” significa todo el mundo.

- tipo (type): puede tomar dos valores, “soft” y “hard”. Representan respectivamente el límite “suave” y el límite “duro”. El suave indica la cantidad que no debe ser sobrepasada, pero que se puede superar de forma temporal; el duro es el límite que nunca puede superarse. Por decirlo de algún modo, la diferencia entre ellos nos da un cierto margen que puede traspasarse durante algún tiempo, pasado el cual debemos bajar del límite suave.

- objeto (item): es el elemento que vamos a limitar; de entre las opciones disponibles las más utilizadas son:

- fsize: tamaño máximo de archivos (en KiloBytes).
- memlock: máxima cantidad de memoria (en KiloBytes).
- cpu: máximo tiempo de CPU (en minutos).
- nproc: número máximo de procesos.
- maxlogins: número máximo de logins simultáneos.
- priority: la prioridad con la que ejecutar los procesos de este usuario.

De esta forma, si queremos que nadie realice más de cuatro logins podemos añadir:

```
*      hard      maxlogins      4
```

Y si queremos limitar el tamaño máximo de archivo para el grupo “internet” a 2 MB pero que pueda llegar a 4 de forma puntual:

```
@internet      soft      fsize      2048
@internet      hard      fsize      4096
```

Ten en cuenta que no es posible (ni conveniente) establecer límites para root.

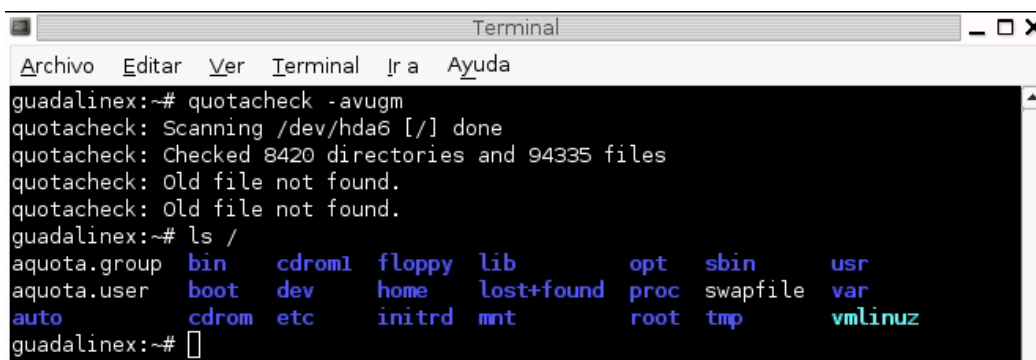
Existe otro mecanismo para establecer límites, pero está pensado para limitar y

controlar el uso del espacio en disco disponible en un sistema. Se pueden establecer límites en la cantidad de espacio y el número de ficheros de que puede disponer un usuario o grupo. Para poder utilizarlo se necesitan cuatro elementos:

- Configurar el núcleo con la opción “Disk QUOTA support”; los núcleos que se distribuyen ya disponen de esta opción activada.
- Disponer de herramientas para manipular las quotas; el paquete se denomina “quota” y puede instalarse mediante apt (apt-get install quota). Si no lo tienes instalado ya ahora es el momento de hacerlo.
- Habilitar las quotas sobre uno o más sistemas de ficheros; para ello hay que editar el archivo /etc/fstab y añadir las opciones “usrquota” y “grpquota” al sistema de ficheros en el que queramos activar las quotas (por ejemplo el montado como “/”). Antes de que se te olvide modifica ese archivo ahora; para activar los cambios y no tener que reiniciar escribe “mount -o remount /” (o la partición que corresponda).
- Configurar las quotas. Para realizar este último punto hay que ejecutar:

```
# quotacheck -avugm
```

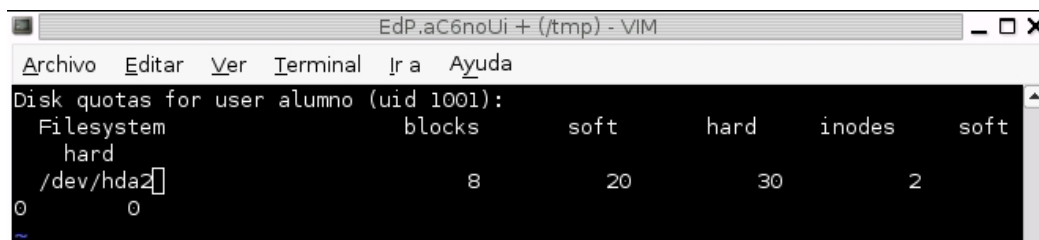
La primera vez que se ejecuta va a crear dos archivos en el directorio raíz: aquota.user y aquota.group, como puedes ver en la imagen.



```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# quotacheck -avugm
quotacheck: Scanning /dev/hda6 [/] done
quotacheck: Checked 8420 directories and 94335 files
quotacheck: Old file not found.
quotacheck: Old file not found.
guadalinux:~# ls /
aquota.group  bin      cdrom1  floppy  lib      opt      sbin     usr
aquota.user   boot     dev     home    lost+found  proc     swapfile var
auto          cdrom    etc     initrd  mnt      root     tmp      vmlinuz
guadalinux:~#
```

Para editar la quota de un usuario o grupo se usa el programa “edquota” con la opción “-u” para usuarios y “-g” para grupos; lo único que hay que modificar son los límites “soft” y “hard”, que tienen el mismo significado que antes. Pueden limitarse el número de bloques y el número de nodos-i; los números que aparecen son los que está utilizando el usuario en ese momento.

Este editor es similar a “vi”, por lo que si no estás familiarizado con él te costará un poco manejarlo. De cualquier manera sigue este orden: pulsa la tecla “i”, muévete con las flechas del cursor hasta la cifra del límite soft, escribe 20, colócate sobre el límite hard y escribe 30; después pulsa la tecla de escape (esc) y teclea “:qw” (sin las comillas); si lo has hecho bien volverás al símbolo del sistema.



The screenshot shows a terminal window titled "EdP.aC6noUi + (/tmp) - VIM". The menu bar includes "Archivo", "Editar", "Ver", "Terminal", "Ir a", and "Ayuda". The terminal output displays disk quotas for user 'alumno' (uid 1001) on the filesystem 'hard' located at '/dev/hda2'. The table shows the following values:

Filesystem	blocks	soft	hard	inodes	soft
hard	8	20	30	2	

Se puede cambiar de la misma forma el periodo de gracia con “edquota -t”.

Puedes ver las cuotas establecidas para un usuario llamado “alumno” con “quota -u alumno”, y la de todos los usuarios con “repquota sistema_de_ficheros”.

Para eliminar las cuotas basta con establecer sus límites soft y hard a cero.

Ejercicios

- 1) Introduce un disquete vacío y crea en él un sistema de archivos vfat.
- 2) Ya conoces el nombre del explorador de carpetas “nautilus”; ¿en qué directorio crees que estará instalado el programa encargado de lanzarlo?
- 3) Introduce un CD-ROM y móntalo
- 4) Crea un enlace simbólico dentro de tu directorio principal que apunte al CD-ROM. Entra en él y comprueba que puedes ver los archivos.
- 5) Establecer una comprobación automática del sistema principal cada 15 días.

- 6) Crea un usuario llamado “ejemplo” y limita el número de procesos que puede ejecutar a 30, el tiempo de uso de la CPU a 1 hora y el tamaño de la memoria que puede utilizar a 8 MB.
- 7) Establece una cuota de disco en la partición principal del sistema (raíz) para el usuario “ejemplo” de 200 bloques y 40 nodos-i, los cuales podrá sobrepasar en igual cantidad durante 2 semanas.
- 8) Comprueba el resultado visualizando las cuotas para el usuario “ejemplo”

Soluciones

- 1) El comando es “mkfs -t vfat /dev/fd0”
- 2) Ese programa es propio de distribuciones como GuadaLinux, pero no es común a todos los Linux, así que empezariamos por buscarla en /usr; por otro lado, aunque es una aplicación gráfica, no es una aplicación que pertenezca al sistema gráfico (no es indispensable), así que no estará en /usr/X11R6. Utiliza la orden “which nautilus” para conocer su ubicación exacta.
- 3) Como ya existe la entrada en /etc/fstab, puedes poner simplemente “mount /cdrom”.
- 4) ln -s /cdrom /home/alumno/cdrom, suponiendo que el usuario sea “alumno”.
- 5) El comando es: “tune2fs -i 2w /dev/hda2”; alternativamente se puede utilizar 15d en lugar de 2w.
- 6) En limits.conf debes escribir:
ejemplo hard nproc 30
ejemplo hard cpu 60
ejemplo hard memlock 8192

Recuerda que el tiempo de CPU hay que ponerlo en minutos y memlock en KB.

- 7) Se supone que ya tienes activada la cuota para /; debes escribir “edquota ejemplo” y poner:

```
/dev/hda2                      x        200    400                      x        40       80
```

Las “x” depende de tu sistema y no hay que modificarlas. Por otro lado debes

ejecutar “edquota -t” y poner un valor de 14 días.

8) Tan sólo escribe “quota -u ejemplo”

CAPÍTULO 6

Sistema X-Window

Xfree86

Todos los sistemas operativos actuales tienen la posibilidad de ser manejados mediante una interfaz de usuario gráfica (denominada GUI, del inglés Graphic Usuary Interface). Algunas máquinas de servidor realmente no necesitan un componente GUI, e incluso algunas máquinas están perjudicadas porque la GUI ocupa recursos (memoria, CPU, ...) que de otra forma podrían tener un mejor empleo. Sin embargo, en máquinas de usuarios o aquellas que requieren componentes multimedia para desarrollar sus objetivos se presenta como una herramienta indispensable.

La GUI utilizada en Linux tiene una estructura de cliente/servidor; el cliente reside en la máquina final que utiliza la GUI, pero el servidor puede estar en otra máquina, la cuál aceptaría conexiones de varios clientes (aunque también puede residir en la misma máquina, que es el caso más común).

El servidor en Linux recibe el nombre de Xfree86, también conocido como Sistema X-Window o simplemente X; está basado en un servidor comercial llamado X11, que es el sistema X-Windows original que se utiliza en muchos sistemas Unix. Su trabajo consiste en su mayor parte en dibujar los elementos en la pantalla. Proporciona una base común a todas las aplicaciones ya que es el encargado de comunicarse con el hardware (tarjeta gráfica, teclado, ratón, ...).

Un elemento instalado en la parte del cliente es el gestor de ventanas; esta herramienta gestiona el movimiento de las ventanas, el cambio de tamaño, pero no el dibujo y la apariencia verdadera de las ventanas; sin embargo, el gestor de ventanas no es el cliente verdadero, sino que es un intermediario entre el servidor y el verdadero cliente X que coge las instrucciones del movimiento de las ventanas

y las traduce en verdadero movimiento. Algunos gestores de ventanas son los siguientes: Afterstep, Blackbox, Enlightenment, Fluxbox, IceWM, Sawmill, Window Maker, Twm, Xfce, ...

El otro lado de la moneda del cliente X es el gestor de pantalla, que gestiona la apariencia del escritorio. Incluye cuestiones como los registros de usuario (cuándo entra o sale un usuario del sistema), ya que dichos usuarios pueden entrar y salir de sus sesiones sin tener que abandonar la GUI; también permite seleccionar el gestor de ventanas a utilizar. Los gestores de pantalla más comunes son: xdm (gestor de pantalla de X), gdm (para Gnome) y kdm (para Kde). Todos se ejecutan en forma de demonios, con lo que al cerrar la sesión siempre se vuelve a la ventana inicial.

Por otro lado se dispone de la posibilidad de utilizar un entorno de escritorio (aunque no es imprescindible), que proveen gestores de ventanas y pantalla, además de sus propios temas, herramientas, iconos, sonidos y mucho más (incluso aplicaciones). Existen dos entornos de escritorio: GNOME (Gnu Network Object Model Environment) y KDE (K Desktop Environment). Cada distribución opta por uno de ellos, aunque siempre puede instalarse el otro; se pueden tener instalados simultáneamente y además ambos permiten cambiar el gestor de ventanas que utilizan (aunque como ya hemos dicho tienen el suyo propio). Guadalinex viene con Gnome ya instalado.

Hay un número de archivos utilizados por el sistema X-Windows; aunque existe alguna variación entre las distintas distribuciones la localización de ellos está más o menos estandarizada

/etc/X11/XF86Config-4: archivo de configuración X principal para la versión 4 de Xfree86 (existe uno para la versión 3, XF86Config, obsoleto); dada su importancia lo veremos en detalle en la sección siguiente.

/usr/X11R6/lib/X11/fonts: directorio que contiene una serie de subdirectorios que guardan todas las fuentes utilizadas en el sistema X-Window.

/usr/X11R6/lib/X11/app-defaults: contiene archivos de configuración para las diferentes aplicaciones disponibles en X y en qué condiciones se abrirán.

/etc/X11/Xsession: es el script maestro que lanza el gestor de ventanas; también lanza programas que aparecerán en ejecución al entrar en el entorno gráfico; puede existir un archivo .Xsession en el directorio /home

del usuario que se ejecuta después del principal.

/etc/gdm/Sessions: directorio que almacena los scripts de inicio de los distintos entornos disponibles para gdm.

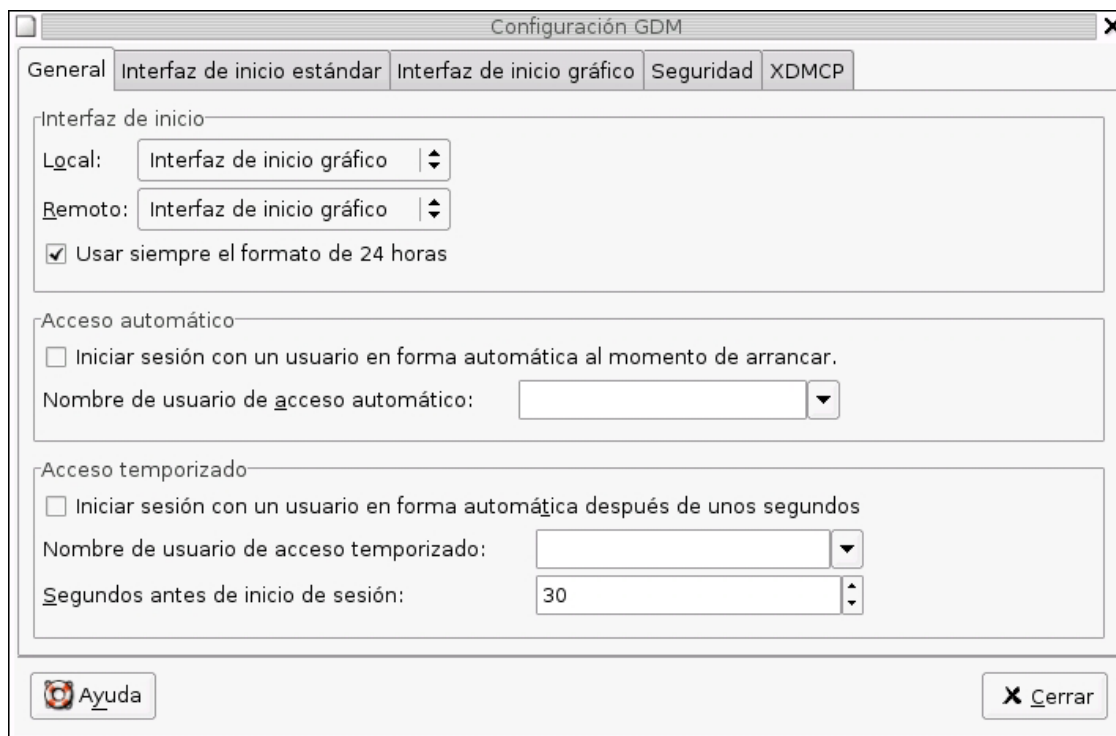
Junto a los archivos y directorios nombrados existen otros que también utilizan las X, los cuales no hay que olvidar, aunque su importancia es menor (no olvide recurrir a ellos si tiene algún problema con las X).

Configuración X-Window

Gdm:

Como ya dijimos es el gestor de pantalla que viene con Gnome; afortunadamente disponemos de una herramienta gráfica para configurarla; se encuentra en el menú Aplicaciones -> Herramientas del Sistema -> Panel de Control y se llama "Configuración de GDM".

Dispone de varias pestañas dedicadas a otras tantas apartados diferentes. La primera es la configuración general, en la que podemos configurar la interfaz que se muestra al entrar en el sistema en el caso de abrir sesión en la misma máquina o abrirla en otra (recuerda que XFree permite que el servidor se encuentre en una máquina remota; también puede especificarse el nombre de un usuario para que entre automáticamente o si lo hará transcurrido un cierto periodo de tiempo (utilízalo con precaución porque no pedirá la contraseña). Las dos pestañas siguiente permiten seleccionar la imagen de fondo en los dos tipos de inicios de sesión disponibles: inicio estándar e inicio gráfico (éste es el que aparece por defecto).



La pestaña de seguridad permite configurar varios aspectos importantes, como si se permite a root acceder al entorno gráfico o de forma remota (ambas opciones se consideran altamente inseguras), si se permite ver el menú de sistema en la ventana de login (que permite apagar la máquina) o si esta ventana de configuración está disponible desde allí.

Todas estas opciones se guardan en el archivo `/etc/gdm/gdm.conf`, donde aparecen opciones adicionales que habría que cambiar con un editor de texto.

XF86config-4:

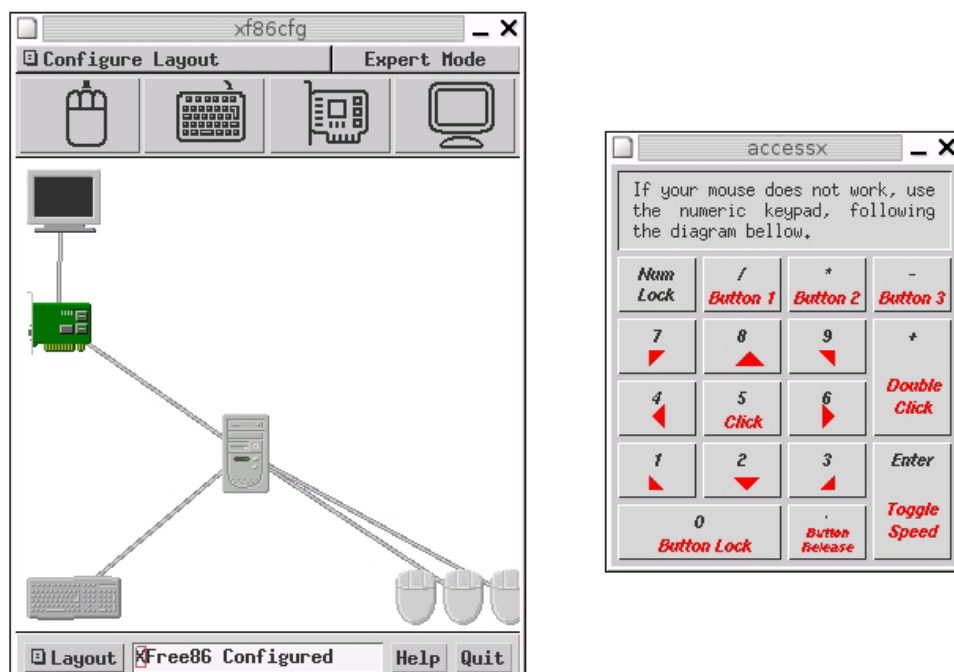
Es con diferencia el archivo más importante, ya que si éste no es correcto no se iniciará el entorno gráfico o no funcionará alguno de nuestros dispositivos hardware.

El archivo (que puede modificarse con cualquier editor de texto) está dividido en secciones; cada sección se identifica por un nombre y tiene una determinada función. Las líneas que comienzan por “#” se ignoran. Modificar este archivo puede resultar confuso, por lo que se utilizan un par de herramientas que

ayudan en la tarea: `xf86config` y `xf86cfg`.

La primera es una aplicación de consola que nos va guiando por los diferentes pasos y elecciones para configurar tanto el teclado, como el ratón, la tarjeta gráfica, el monitor ... Lamentablemente sólo está disponible en inglés, por lo que seguramente preferirás la segunda opción, que es una herramienta gráfica (por otro lado, si no te funcionan las X, no tendrás más remedio que utilizar la primera o modificar el archivo a mano).

Si optamos por utilizar `xf86cfg` debemos ejecutarlo desde un terminal de root; se nos mostrarán las ventanas de la imagen; la derecha simplemente es una ayuda para el caso de que no funcione el ratón. El modo experto (Expert Mode) abre un esquema con las secciones del archivo `XF86Config-4` y su utilización sólo se recomienda para usuarios expertos (de ahí el nombre).



Pulsando sobre "Configure Layout" podemos elegir qué elemento queremos configurar; la disposición de los elementos (layout) la veremos más adelante. Selecciona la pantalla (Configure Screen); pulsando con el botón derecho obtenemos un menú en el que podemos configurar (configure), añadir opciones (options), deshabilitarla temporalmente (disable) o eliminarla completamente (remove); la opción interesante es la de configurar, ya que aquí seleccionamos la resoluciones de pantalla (poniéndolas en la columna derecha) que queremos utilizar; ten en cuenta que tu monitor puede que no soporte las resoluciones más

elevadas; consulta el manual de tu monitor y selecciona las apropiadas.

La sección “Configure ModeLine” sirve para ajustar los modos de vídeo a tu monitor; es una sección avanzada que sólo te recomiendo visitar si en el manual de tu monitor aparecen los modos de línea que utiliza; no intentes usar los de otro monitor (puedes llegar incluso a estropearlo), así que te recomiendo que lo dejes en automático (auto) tal y como está seleccionado por defecto.

La última sección (AccessX) permite configurar las opciones de accesibilidad del entorno gráfico: las teclas para el ratón y la repetición de teclas.

Volviendo a “Configure Layout”, podemos añadir o configurar los cuatro elementos que vemos, que de izquierda a derecha son: el ratón, el teclado, la tarjeta gráfica y el monitor; puede pulsarse sobre estos iconos o sobre los que ya están colocados en la ventana; en cualquier caso seleccionando “configure” podemos elegir la configuración de nuestro hardware. Hay que tener en cuenta que si el dispositivo no aparece específicamente con su nombre es posible que no funcione (no vale con coger el que más se le parezca, aunque a veces funciona); es entonces el momento de probar con dispositivos genéricos que aunque no aprovechen toda la potencia de nuestros dispositivos al menos los hagan funcionar en sus prestaciones básicas.

Las secciones a configurar en cada caso son:

- Ratón: dispositivo (generalmente /dev/psaux) y protocolo (mejor en “auto”)
- Teclado: Xkb rules = xfree86 (teclado para xfree86)
Keyboard model (modelo de teclado) = El que mejor se aproxime al tuyo; si no aparece entonces “Generic 101-key PC” debería funcionar.
Keyboard layout = Spanish
Las otras dos opciones pueden dejarse en blanco.
- Tarjeta gráfica: busca el chipset de tu tarjeta (consulta al manual) y selecciona el tuyo; el identificador es simplemente un nombre que se puede modificar. También es importante el driver, es posible que necesites descargarlo de algún repositorio (si no aparece selecciona “vesa” o “vga”).
- Monitor: Selecciona uno de la lista y de nuevo escribe un nombre en “Identifier”.

ADVERTENCIA: No selecciones uno que tenga una frecuencia horizontal

(Horizontal sync) mayor que la que soporte tu monitor, podrías estropearlo. Consulta el manual de tu monitor para asegurarte; en cualquier caso el primero de la lista (Standar VGA) no debería dañarlo.

Una vez aplicados los cambios te recomiendo ahora sí que eches un vistazo al archivo `/etc/X11/XF86Config-4` para que veas los parámetros que has introducido.

Es preciso resaltar que este archivo es común para todos los usuarios y entornos que utilicen este servidor X; una vez dentro de la sesión el usuario (ya sea local o remoto) puede establecer una configuración personal (que se guardará en su directorio `/home`) mediante los lanzadores situados en Aplicaciones -> Herramientas del Sistema -> Preferencias de Escritorio, por ejemplo la resolución de pantalla.

CAPÍTULO 7

Introducción a las redes TCP/IP

TCP/IP

Una de las características más importantes de Linux es su cualidad como servidor de redes. La red en Linux (y el propio Linux de forma interna) habla el mismo idioma que Internet, lo que hace que el hecho de dar servicios de Internet y acoplar una red de área local no suponga prácticamente ningún esfuerzo, ya que en el proceso no se necesitan máquinas traductoras. Este lenguaje se conoce de forma genérica como TCP/IP (de las siglas en inglés de Protocolo de Control de Transmisión/Protocolo de Internet); en realidad son un conjunto de protocolos agrupados para crear una comunicación más fluida por la red, cada uno de ellos realizando un trabajo muy específico.

El funcionamiento de TCP/IP se basa en el uso de unas direcciones, denominadas direcciones IP, para identificar de qué máquina procede cada mensaje y a qué máquina se dirigen. Estas direcciones tienen el formato: ###.###.###.###, donde cada grupo de tres dígitos (denominado octeto ya que se puede representar por un Byte) puede estar comprendido entre 0 y 255; de ella una parte se dedica a la dirección de red y el resto identifica a la máquina dentro de esa red. El número de cifras de la dirección dedicada a cada uno de estas componentes (red y máquina) varía de un tipo de red a otra. En cierto modo se asemeja a un número de teléfono, donde las primeras cifras forman el prefijo de zona y el resto el número de abonado dentro de la zona.

Para separar las dos partes se utiliza la máscara de red, que no es más que un número con la misma estructura de dirección IP pero con un significado distinto. Así, se distinguen varios tipos de direcciones que definen otras tantas clases de redes, cada una con una máscara de red distinta:

A:	0.0.0.0 a 127.255.255.255	255.0.0.0
B:	128.0.0.0 a 191.255.255.255	255.255.0.0
C:	192.0.0.0 a 223.255.255.255	255.255.255.0

Existen dos tipos de redes más (tipo D, de 224.0.0.0 a 239.255.255.255, y tipo E, de 240.0.0.0 a 247.255.255.255, pero no se utilizan)

En una red de tipo A el primer octeto numera a la red, mientras que los otros tres numeran las máquinas. En redes de tipo B sin embargo se reservan dos octetos para la red y dos para las máquinas. En las de tipo C son tres los octetos para numerar la red y uno para la máquina. Esto puede observarse por la máscara propia de cada tipo de red.

Hay una dirección especial compartida por todas las máquinas que utilizan TCP/IP; es la dirección 127.0.0.1 y se denomina dirección de circuito cerrado o dirección de lazo local; siempre es una referencia a la propia máquina en la que se trabaja y es comúnmente conocida como "localhost".

En toda red hay un par de direcciones "reservadas", que no pueden utilizarse para numerar a ninguna máquina. Estas dos direcciones se conocen con el nombre de "dirección de red" (que siempre es la menor de toda la red) y "dirección de difusión" o "dirección de broadcast" (que siempre es la última). La primera sirve para nombrar e identificar a la red como un bloque, mientras que la dirección de difusión se utiliza para enviar datos de manera simultánea a todas las máquinas de la red.

De esta forma, podemos ver que existen 126 redes distintas de tipo A cada una con algo más de un millón y medio de máquinas posibles. Existirán del mismo modo unas 16000 redes de tipo B, cada una con 65000 máquinas. Por último, hay algo más de dos millones de redes de tipo C en las que caben 254 máquinas en cada una.

Para poder utilizar una IP es necesario solicitarla, y sólo se conceden a organismos o empresas capacitadas para ello (por ejemplo los proveedores de internet); cada vez es más difícil disponer de una ya que se está llegando a una situación en la que casi no quedan IP disponibles, debido a que en el actual sistema de numeración se desaprovechan muchas direcciones (si tenemos una red de tipo A y poseemos 700000 máquinas estamos desaprovechando todavía 16 millones de direcciones que nadie puede utilizar. Esto se ha solucionado implantando un sistema de numeración de 64 bits (en lugar de los 32 actuales); este sistema, a parte

de ser virtualmente inagotable (equivaldría a tener unos 200 aparatos por metro cuadrado cubriendo toda la superficie de la tierra) es compatible con el actual. Ésta y otras mejoras se introducen en la versión 6 del protocolo IP (conocida como IPv6, la actual es IPv4), que tímidamente se está empezando a implantar en algunos sistemas (Linux es totalmente compatible con él desde hace tiempo).

Por otro lado, la estructura de la comunicación entre dos máquinas responde a un modelo de capas, una sobre otra, de manera que el conjunto de tareas se divide en partes que son ejecutadas en una determinada capa. Cada capa se comunica con una similar en el otro extremo de la comunicación intercambiando una serie de datos, que son utilizados por la capa superior; cada capa además posee una información de control que es entendida únicamente por su capa pareja y que es la que permite que se produzca el intercambio de datos.

En TCP/IP existen cuatro capas diferenciadas; la primera es la capa de red local, que es la encargada de llevar la información de una máquina a otra, resolviendo los problemas que puedan ocurrir en la transmisión (ya sea por cable, inalámbrica, óptica, ...). Los elementos que residen en esta capa son los controladores de dispositivos de red, como la tarjeta Ethernet o el módem.

Sobre ella se encuentra la capa internet o “entre redes”, que es dónde reside IP. Es la responsable de decirle a los datos dónde tienen que ir para llegar a su destino, para lo que utiliza las direcciones IP (lo veremos un poco más adelante); el envío de datos se realiza a una máquina accesible directamente de la red.

La capa de transporte es la primera que conoce la transmisión extremo a extremo, no a máquinas próximas como hace la capa de red. Aquí es donde trabaja TCP, que se encarga de fragmentar los paquetes en trozos más pequeños que son enviados junto con información para su ensambladores. TCP se ocupa también de servicios que requieren una conexión ininterrumpida con comprobaciones de los errores y recuperación ante ellos. Para servicios que no necesitan controlar si hubo o no un error o que no necesitan de una conexión establecida permanentemente con el receptor se utiliza el protocolo UDP (Protocolo de Datagrama de Usuario), más rápido que TCP en ambientes fiables (como una red de área local, por ejemplo).

Por último tenemos la capa de Aplicación, que es donde residen los diferentes programas que utilizan Internet; cada una de estas aplicaciones ofrece unos servicios al usuario (o a otros programas), para lo que cual necesitan utilizar uno de los protocolos de la capa de transporte, ya sea TCP o UDP. Por ejemplo los

servicios FTP, POP3 y SMTP utilizan TCP, DNS y NFS usan UDP y HTTP utiliza ambos. Para poder conocer a qué aplicación se destinan los datos se utiliza un número de puerto, de manera que la pareja dirección IP/puerto determina de forma única la aplicación a la que se destinarán los datos.

No te preocupes si no conoces el significado de todas estas siglas, las veremos en detalle más adelante; de todas formas existe una lista exhaustiva de los servicios existentes y los puertos y protocolos que utilizan en el archivo `/etc/services`. Lo que debes tener muy claro es que TCP y UDP poseen puertos independientes, es decir, cada uno tiene su propia numeración.

Configuración del Hardware

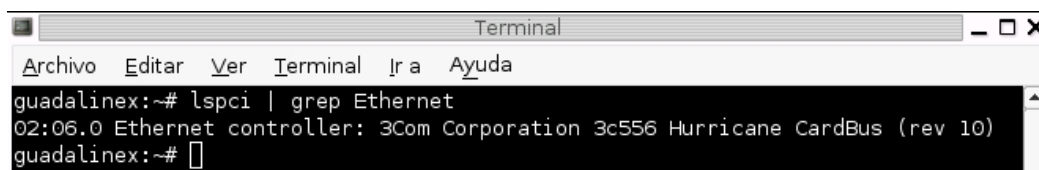
Para la interconexión física de ordenadores se emplea algún tipo de tecnología de red de área local; la más común con diferencia son las redes Ethernet. Para construir una red de este tipo cada ordenador debe poseer una tarjeta de red o interfaz de red; esta interfaz posee una dirección única que viene grabada de fábrica en cada tarjeta, por lo que se suele llamar dirección física; en redes Ethernet se compone de 6 bytes, que se suelen representar en hexadecimal (por ejemplo 06:00:1b:4c:48:33).

Otro mecanismo muy común para conectarse a una red es mediante una conexión punto a punto a otro ordenador (típicamente con un módem); en este caso la red local está formada únicamente por dos ordenadores. La tecnología equivalente para este tipo de conexiones se denomina PPP (Protocolo Punto a Punto). Para este tipo de conexión existe un icono en el escritorio llamado “Conexión a Internet”, el cual posee instrucciones precisas para el caso de módem analógico, módem USB o router USB, por lo que el resto del capítulo nos centraremos en el caso de redes locales.

Para configurar una conexión de red es necesario ejecutar dos tareas básicas: configurar la interfaz de red y configurar el TCP/IP.

La configuración de la interfaz consiste básicamente en cargar el driver apropiado para la tarjeta o tarjetas que tengamos. Estos módulos se cargan como cualquier otro módulo del núcleo (generalmente se hace automáticamente en el arranque); para saber qué tipo de tarjeta de red tenemos podemos recurrir a la

salida del comando “lscpi”.

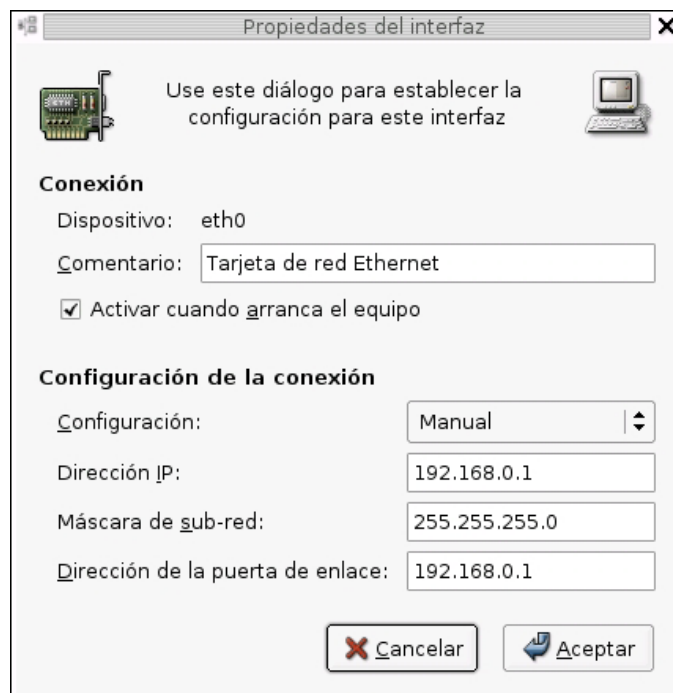


```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# lspci | grep Ethernet
02:06.0 Ethernet controller: 3Com Corporation 3c556 Hurricane CardBus (rev 10)
guadalinux:~#
```

A continuación debemos conocer el módulo necesario para esta tarjeta; para este caso resulta ser el 3c59x (se suele encontrar esta información con ayuda de cualquier buscador de Internet), de manera que para cargar el módulo tan sólo hay que poner “modprobe 3c59x” y el módulo será cargado en memoria. Cada módulo cargado asignará una vez configurado un nombre de dispositivo a la tarjeta, empezando por eth0 y siguiendo por eth1, etc (para equipos con varias tarjetas, si no estará sólo disponible eth0).

Respecto a la configuración de TCP/IP, disponemos de una herramienta gráfica en Aplicaciones -> Herramientas del Sistema -> Panel de Control -> Red. En primer lugar ponemos el nombre que la máquina tendrá en la red; este valor se guarda en el archivo /etc/hostname. Los siguientes apartados se refieren a la inclusión de nuestro Linux en una red con máquinas Windows y lo trataremos en un capítulo posterior.

La siguiente pestaña es quizá la más importante, ya que desde aquí se controlan las distintas interfaces de red de que disponemos; podemos añadir, borrar, activarla o desactivarla, y la principal, modificar sus propiedades, como su dirección IP, máscara de red y la puerta de enlace. Esta última es la dirección IP de una máquina que hace de puente entre nuestra red y otra, de manera que cuando intentemos comunicar con una máquina que no esté en nuestra red los paquetes se encaminen a través de esta máquina; el caso típico es el de una máquina o router que se utiliza para dar conexión a Internet a toda una red local (Internet no es más que una conexión entre muchas redes).



Los valores que se muestran no están elegidos al azar. En principio una red local aislada del resto de redes de Internet podría tomar cualquier valor para sus direcciones IP (aunque todas dentro de la misma red), ya que esta red no va a interferir con el resto de redes; sin embargo es costumbre utilizar una serie de rangos IP para configurar redes locales (que no se utilizan para máquinas conectadas directamente a Internet), rangos que se reservan precisamente para este uso. Existen 3 rangos disponibles: 10.0.0.0 a 10.255.255.255, 172.16.0.0 a 172.31.255.255, y 192.168.0.0 a 192.168.255.255 (como puedes observar hay un rango de tipo A, otro de tipo B y otro de tipo C, para que escojamos según nuestras necesidades).

También se puede configurar si la interfaz se activará al iniciar la máquina o si se realiza una selección automática de los valores en el menú desplegable "Configuración"; aquí las opciones son manual, DHCP o BOOTP (para utilizar las dos últimas es necesario que en la red existan sendos servidores dedicados a estos servicios).

Aunque ya podemos comunicarnos con el resto de máquinas, nos falta todavía activar una característica para hacer nuestros movimientos por la red más cómodos. Estarás de acuerdo conmigo que recordar las IP de la máquinas a las que queremos conectarnos es prácticamente imposible; por ello disponemos de dos características que nos resuelven esta tarea: los anfitriones y los servidores de nombres.

La lista de anfitriones es simplemente una lista que asocia a una IP un nombre, de manera que podemos introducir las IP de las máquinas de nuestra red con sus respectivas IP, y a partir de entonces nos podemos referir a esa máquina con ese nombre. Hay que tener en cuenta que esta lista debe introducirse en cada máquina de la red que se quiera que la utilice; en realidad sólo es necesario hacerlo una vez, ya que la lista se almacena en el archivo `/etc/hosts`, con lo que se puede copiar a todas las máquinas de nuestra red. Puedes ver que hay una entrada para `"localhost"` (el resto es necesario para poder usar IPv6).

Ahora bien, estos nombres sólo tienen ámbito local. Si queremos acceder a máquinas de fuera de nuestra red la lista debería ser bastante larga; en su lugar se utilizan unos servidores de Internet dedicados a almacenar esas extensas listas; son los denominados "Servidores de Nombres de Dominio" o DNS. Sus IP deben ser proporcionadas por tu proveedor de Internet (cada proveedor tiene las suyas, aunque en realidad puedes usar las de cualquiera). Simplemente puedes añadirlos desde la pestaña que reza "DNS".

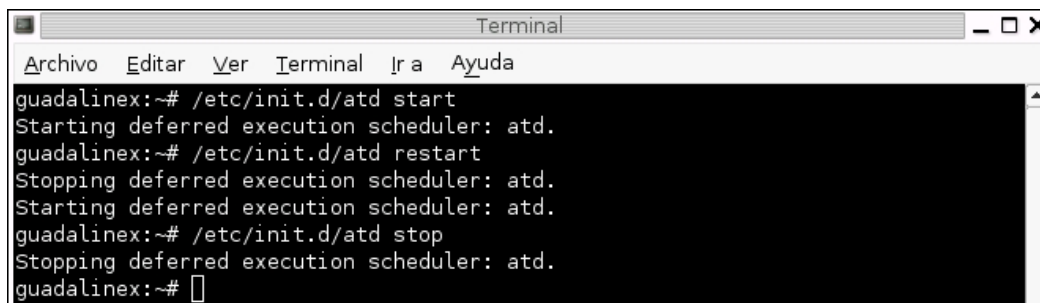
Servicios y control de acceso

Ya sabes que existen una serie de servicios que pueden estar ejecutándose en tu máquina y que ofrecen algún tipo de utilidad; algunos de ellos están pensados para dar servicio a otras máquinas de la red además de la propia. Ya sabes cómo activar y desactivar estos servicios desde el entorno gráfico, pero ahora lo veremos desde la consola.

Típicamente los servicios (tanto de red como los demás) se controlan mediante el uso de un script de texto localizado en `/etc/init.d`; ejecutando este script se puede iniciar el servicio, detenerlo o reiniciarlo (por ejemplo después de realizar alguna modificación en su configuración), tan sólo es necesario conocer el nombre de ese script.

Ya conoces el demonio `atd`, encargado de la ejecución programada de comandos; para iniciarlo es necesario invocar desde la línea de comandos: `/etc/init.d/atd start`, para detenerlo: `/etc/init.d/atd stop`, y para reiniciarlo

/etc/init.d/restart.



```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# /etc/init.d/atd start
Starting deferred execution scheduler: atd.
guadalinux:~# /etc/init.d/atd restart
Stopping deferred execution scheduler: atd.
Starting deferred execution scheduler: atd.
guadalinux:~# /etc/init.d/atd stop
Stopping deferred execution scheduler: atd.
guadalinux:~#
```

Este funcionamiento de los servicios recibe el nombre de “standalone” ya que el demonio estará a la espera de que se necesiten sus servicios. Pero existe otra forma de tener estos servicios en funcionamiento conocida como activación “bajo demanda”; este modo de funcionamiento está reservado a los servicios de red y se basan a su vez en un servicio llamado “inetd”.

El programa inetd es un demonio (conocido a veces como “superdaemon” o superdemonio) que escucha peticiones en muchos puertos a la vez; cuando detecta actividad en alguno de los puertos arranca el servidor correspondiente al servicio solicitado. Este funcionamiento es conveniente por dos razones: evita tener en continuo funcionamiento un gran número de servidores que sólo se emplean ocasionalmente, y permite conceder o denegar el acceso en función del origen de la conexión antes de iniciar el servidor correspondiente.

Este servicio se configura en el archivo /etc/inetd.conf. Cada línea se utiliza para iniciar un servicio; si consultas este archivo encontrarás estas dos líneas:

smtp	stream	tcp	nowait	mail	/usr/sbin/exim	exim -bs
ftp	stream	tcp	nowait	root	/usr/sbin/tcpd	/usr/sbin/in.ftpd

La primera línea configura el servidor de correo electrónico de correo saliente o MTA (siglas en inglés de Agente de Transporte de Correo). El primer valor es el nombre del servicio tal y como aparece en /etc/services. Los dos siguientes indican el tipo de conexión y el protocolo a utilizar, y generalmente valen “stream tcp” o “dgram udp”, según que el servicio utilice un flujo (stream) TCP o datagramas UDP. El campo wait/nowait sólo tiene validez para datagramas e indica si hay que esperar a que se cierre la conexión abierta para atender otras peticiones.

A continuación se encuentra el usuario (y eventualmente el grupo) a través

del cual se ejecutará el servidor; recuerda que vimos que existían usuarios “ficticios” que no se correspondían con una persona física sino que eran utilizados por programas y demonios para funcionar. En este caso el servicio smtp tomará la personalidad del usuario y grupo “mail”.

Para finalizar se encuentra la ruta completa del programa ejecutable para ese servicio seguido en su caso de los argumentos necesarios. En general no es necesario modificar este archivo a mano, puesto que al instalar el servidor correspondiente añadirá la línea adecuada en el archivo inetd.conf.

En ocasiones no aparece el programa ejecutable del servidor directamente sino que aparece “/usr/sbin/tcpd” antes de él, como ocurre en la segunda de las líneas; de esta forma se puede realizar el control de acceso que veremos a continuación. Dicho control de acceso permite o deniega el acceso a los servicios en función del origen de la conexión; este mecanismo se configura en los archivos /etc/hosts.allow y /etc/hosts.deny, donde se especifican unas sencillas reglas que permiten aceptar o rechazar conexiones entrantes. Además de los servicios activados mediante inetd hay algunos servidores que siempre utilizan este método de control de acceso (como sshd), conocido como TCP wrappers (cubiertas o envolturas para TCP).

Estos dos ficheros tienen una lista de la forma:

demonio : cliente

“Demonio” es el nombre del ejecutable, no el nombre del servicio (cuidado con esto, es fuente de numerosos quebraderos de cabeza) y el cliente es el origen al que se le permite o deniega el acceso a ese servicio.

Como demonio se pueden poner uno o más nombres separados por espacios o comas, y pueden usarse comodines como “?” (coincidencia con un carácter) o “*” (coincidencia con cualquier grupo de caracteres); también puede usarse la palabra “ALL” para indicar “cualquier servidor”. El cliente por su parte puede ser una o más direcciones IP, redes o nombres de máquinas separadas por espacios o comas; se pueden usar las palabras LOCAL (para especificar conexiones desde la misma máquina) o EXCEPT, para escribir excepciones.

El funcionamiento de ambos ficheros es muy sencillo; cuando el servicio arranca, se comprueba el contenido de /hosts.allow en busca de alguna regla que coincida; si se encuentra se permite el acceso. Si no se encuentra ninguna, se

comprueba `hosts.deny`; si se encuentra una regla coincidente, se deniega el acceso. Observa que el acceso puede concederse de dos maneras distintas: por coincidir en el primero o por no coincidir en el segundo; ten esto en cuenta para no encontrarte con accesos no permitidos.

Veamos en un ejemplo; imaginemos que tenemos configurado el servidor FTP para ejecutarse a través de `inetd` (ya veremos cómo hacerlo en un capítulo posterior), con una línea como vimos anteriormente; si queremos que sólo accedan a él las máquinas de nuestra red excepto `192.168.0.13` tendremos que poner:

```
# contenido de /etc/host.allow
in.ftpd: LOCAL, 192.168.0 EXCEPT 192.168.0.13
```

```
# contenido de /etc/host.deny
ALL: ALL
```

No hay que olvidar esto último, ya que si no estaríamos permitiendo el acceso desde fuera de nuestra red.

Existen dos tipos de políticas de seguridad frecuentemente utilizadas: una de casi todo abierto y otra de casi todo cerrado. La primera consiste en permitir todo lo que no esté expresamente denegado; se implementa dejando vacío `hosts.allow` y denegando el servicio concreto en `hosts.deny`. La segunda por el contrario consiste en denegar todo lo que no esté permitido y se establece como en el ejemplo anterior, poniendo los accesos permitidos en el `hosts.allow` y denegando todo en el `hosts.deny`; esta última es en general la más adecuada.

Es posible que te tropieces alguna vez con un servicio especial llamado “`portmap`”; este servicio se encarga de encontrar un puerto disponible para servicios que no utilizan un número de puerto fijo sino que se acceden mediante un nombre; estos servicios se denominan de forma genérica RPC (Remote Procedure Call, o llamada a procedimiento remoto); en este caso es necesario indicar el origen mediante una IP, no es posible poner un nombre. Para protegerse ante posibles conexiones desde máquinas haciéndose pasar por otras se puede usar un nombre de cliente denominado `PARANOID`, que coincide con todas las máquinas cuyo nombre no coincida con su IP.

Ejercicios

- 1) Calcular exactamente el número de redes y máquinas disponibles en los tipos de redes A, B y C.
- 2) Escribe los puertos y el protocolo que utilizan los siguientes servicios: POP3, SMTP, HTTP, FTP.
- 3) Suponiendo que existe la siguiente línea en /etc/inetd.conf:
pop3 stream tcp nowait root /usr/sbin/tcpd /usr/sbin/ipop3d
imap2 stream tcp nowait root /usr/sbin/tcpd /usr/sbin/imapd

configura los TCP wrappers para que el primero no sea utilizado por la red 192.168.10.0, y el segundo pueda ser utilizado por los clientes 192.168.10.14 y 192.168.10.15; la política de seguridad debe ser del tipo “casi todo abierto”.
- 4) Repite el ejercicio anterior con una política de “casi todo cerrado”.

Soluciones

- 1) En las expresiones siguientes se han sustituidos por “x” los valores que van cambiando en cada caso:

Tipo A: 0.0.0.0 a 127.255.255.255, máscara 255.0.0.0

Primera red: 0.x.x.x, última: 127.x.x.x -> 128 redes, pero 0.0.0.0 y 127.x.x.x están reservadas = 126 redes

Máquinas: de x.0.0.0 a x.255.255.255, pero hay que restar 2 (direcciones de red y difusión) -> $256 \times 256 \times 256 = 16777216$, menos 2 = 16777214 máquinas.

Tipo B: 128.0.0.0 a 191.255.255.255, máscara 255.255.0.0

Primera red: 128.0.x.x; última red: 191.255.x.x -> $(191-128) \times 256 = 16128$

Máquinas: x.x.0.0 a x.x.255.255, a las que hay que restar 2 -> $(256 \times 256) - 2 = 65534$ máquinas.

Tipo C: 192.0.0.0 a 223.255.255.255, máscara 255.255.255.0

Primera red: 192.0.0.x; última: 223.255.255.x -> $(223-192)*256*256 = 2031616$ redes.

Máquinas: x.x.x.0 a x.x.x.255, menos 2 -> $256 - 2 = 254$ máquinas.

2) Los puertos y protocolos son (consultar /etc/services): POP3 = 110 (tcp y udp), SMTP = 25 (tcp), HTTP o WWW = 80 (tcp y udp), FTP = 21(tcp). Debes ir familiarizándote con ellos porque los usaremos más adelante.

3) Política “casi todo abierto”: hosts.allow vacío; host.deny con las líneas:

pop3: 192.168.10.0

imap2: ALL EXCEPT 192.168.10.14,192.168.10.15

4) Política “casi todo cerrado”: host.deny con “ALL: ALL”; host.allow:

pop3: ALL EXCEPT 192.168.10.0

imap2: 192.168.10.14,192.168.10.15

CAPÍTULO 8

Configuración del correo electrónico

El correo electrónico es, posiblemente, el servicio más popular en Internet. Linux puede convertirse en un magnífico servidor de correo, aunque tradicionalmente configurarlo suele considerarse complicado por la gran cantidad de sutilezas que aparecen. Aquí no vamos a describir cómo utilizar el cliente de correo electrónico, que se supone que ya sabes manejar (en GuadaLinux están Evolution y Mozilla Mail), sino que vamos a atacar la tarea de convertir Linux en un servidor de correo (para leer tu correo de Internet no necesitas tener instalado un servidor de correo, sólo el cliente). Vamos primero a describir cómo es el funcionamiento del proceso para que luego se entienda mejor lo que estamos haciendo.

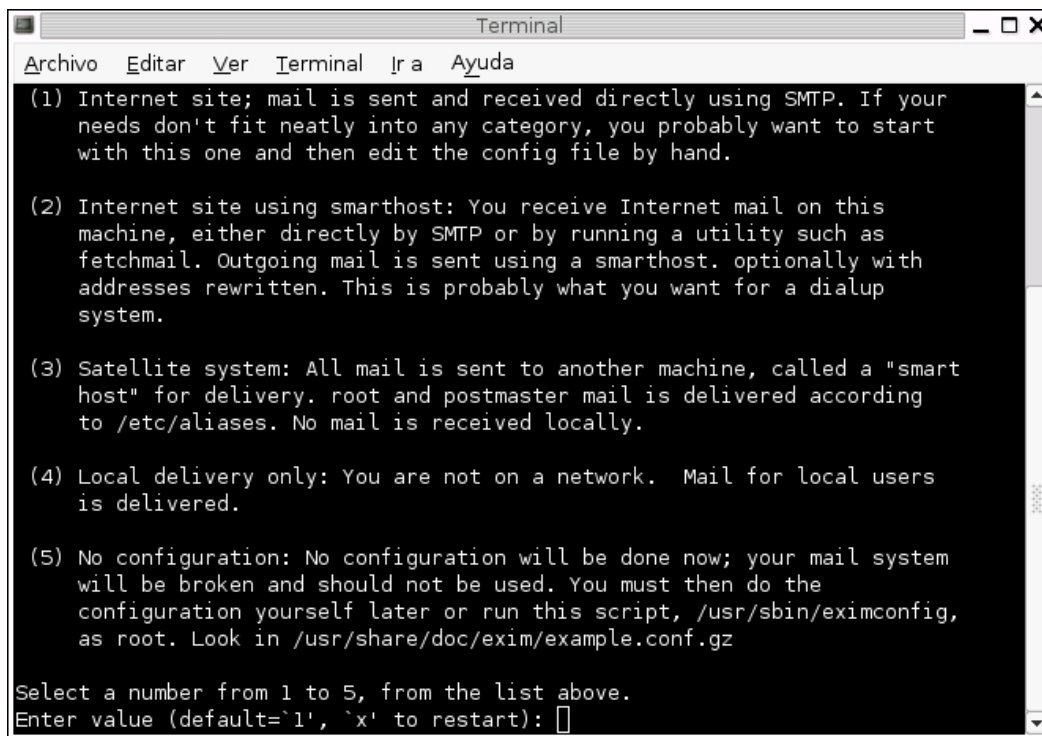
La vida de un correo electrónico comienza cuando un usuario abre un documento de correo nuevo en su cliente de correo. Este cliente, independiente de cuál sea, te solicita de forma inmediata dos informaciones: la dirección de correo del receptor (generalmente en la forma usuario@domino.extensión) y el asunto (que resulta irrelevante para el proceso de entrega del correo electrónico, como también lo es el contenido mismo del mensaje).

Cuando el usuario termina de escribir el mensaje pulsa algún botón para enviarlo; entonces el cliente entrega el correo a un Agente de Transferencia de Correo o MTA (Mail Transfer Agent); si el correo va destinado a otro usuario de la misma máquina, el gestor de correo local se encarga de él, pero si el correo se envía a otra máquina (ya sea de la misma red o de Internet) el correo pasa al servidor de correo. Entonces el MTA utiliza su traductor de direcciones para localizar la dirección de destino, enviando el mensaje mediante el protocolo SMTP (Simple Mail Transfer Protocol, o Protocolo Sencillo de Transferencia de Correo); el servidor del receptor analiza la información de encabezado para comprobar que todo es correcto, y entonces lo deja disponible para que el receptor pueda acceder a él mediante su cliente de correo electrónico.

Tradicionalmente el servidor de correo utilizado en Linux (y en otros sistemas) es "sendmail"; este servidor es difícil de configurar, aunque se facilita algo la tarea mediante las macros; se dice que uno no es un verdadero "administrador" hasta que configura sendmail una vez, y que estarás loco si lo haces una segunda. Existen otras posibilidades que están ganando adeptos debido su versatilidad, como "qmail" o "exim"; aquí describiremos exim, más sencillo que los otros dos (que, además, ya está instalado en GuadaLinux).

La configuración se realiza a través del programa "eximconfig"; este programa se ejecuta automáticamente tras instalar exim y puede ejecutarse posteriormente para modificar su configuración inicial. El programa está en inglés, pero los pasos que hay que dar son muy claros y, salvo para casos excepcionales, las opciones son siempre similares.

Tras ejecutar el programa, lo primero que hace es advertirnos que ya existe un archivo de configuración que será sobrescrito, aunque el antiguo lo renombrará añadiendo la extensión ".O"; simplemente pulsa "enter" para acceder al menú.



```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda

(1) Internet site; mail is sent and received directly using SMTP. If your
needs don't fit neatly into any category, you probably want to start
with this one and then edit the config file by hand.

(2) Internet site using smarthost: You receive Internet mail on this
machine, either directly by SMTP or by running a utility such as
fetchmail. Outgoing mail is sent using a smarthost. optionally with
addresses rewritten. This is probably what you want for a dialup
system.

(3) Satellite system: All mail is sent to another machine, called a "smart
host" for delivery. root and postmaster mail is delivered according
to /etc/aliases. No mail is received locally.

(4) Local delivery only: You are not on a network. Mail for local users
is delivered.

(5) No configuration: No configuration will be done now; your mail system
will be broken and should not be used. You must then do the
configuration yourself later or run this script, /usr/sbin/eximconfig,
as root. Look in /usr/share/doc/exim/example.conf.gz

Select a number from 1 to 5, from the list above.
Enter value (default='1', 'x' to restart):
```

En primer lugar debemos elegir el uso y situación del servidor; de las 5 debemos optar por una. La primera es para configurar un servidor de correo para internet; la dos siguientes se utilizan para redirigir los correos salientes a otro

servidor (la segunda) o redirigir los correos entrantes y salientes (la tercera), mientras que la cuarta se utiliza para usar únicamente correo local; la última es para no utilizar el MTA. Aquí vamos a configurarlo para Internet, así que selecciona la primera; en cualquier momento pulsando la “x” volveremos a este menú.

Lo primero que nos pregunta es el nombre “visible” de nuestro sistema, que es el que aparecerá como remite en los mensajes enviados; si disponemos de un nombre de dominio deberemos utilizarlo; si no dispones de uno deja “guadalinux”. Acto seguido se nos pregunta si nuestro sistema es conocido por otro nombre además del anterior (si tiene un alias); debes introducirlos separados por comas, o poner “none” si no existe ninguno.

A continuación se nos informa que nuestro MTA aceptará correo que venga destinado a él y que enviará el generado dentro de él, pero no efectuará reenvíos; esto es conveniente para evitar que nuestro sistema se utilice para enviar “spam” o correo no deseado;; simplemente pon “none”.

Ahora podemos elegir para qué dominios queremos actuar como servidor de correo, además del nuestro propio; esto es útil para dar servicio a más de una red; la forma de expresar la pareja IP/máscara se realiza en notación dirección/longitud (por ejemplo 194.222.242.0/24, donde 24 significa 255.255.255.0, es decir, el número de bits activados de la máscara).

El paso siguiente es decirle el nombre del usuario al que se dirigirá el correo del superusuario; ésta debe ser la cuenta personal del administrador, para que no sea necesario usar la cuenta de root para leer este correo. Si ya tenemos un archivo /etc/aliases (para los alias) se nos preguntará si queremos conservarlo o instalar uno nuevo, perdiendo las modificaciones que ya tenía (aunque se guarda una copia con extensión “.O”); responde que si (“y”).

Para finalizar se nos muestra un resumen de las opciones configuradas, pudiendo elegir repetir el cuestionario o aceptar las que hemos seleccionado; pulsa “y”.

Y eso es todo; ya sólo queda iniciar o reiniciar el demonio “exim” para empezar a ofrecer este servicio. Antes puedes observar que se han modificado los archivos /etc/exim/exim.conf, /etc/aliases y /etc/mailname, y que se han creado otros con extensión “.O” con las versiones anteriores.

Si las máquinas para las que realizamos las tareas de servidor de correo están muy dispersas en redes y subredes es mejor guardar la lista de estos clientes en un archivo, para que añadir o quitar clientes sea más cómodo; para ello hay que modificar el archivo `/etc/exim/exim.conf` y añadir modificar esta línea añadiendo la última parte:

```
host_accept_relay = 127.0.0.1 : :::: 1 : /etc/exim/host_accept_relay
```

A continuación creamos el archivo `/etc/exim/host_accept_relay` e introducimos la lista de máquinas o redes que pueden usarnos como servidor de correo, cada una en una línea; se pueden poner IP concretas (194.222.242.15), redes completas (194.222.242.0/24) o nombres de máquinas (maquina3.undominio.com), aunque siempre que sea posible conviene utilizar el formato numérico para no depender del servicio de nombres.

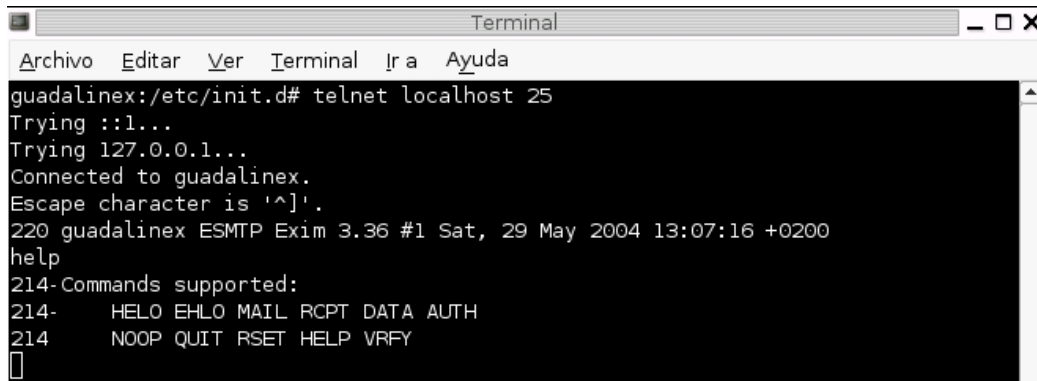
Este sistema funciona cuando los clientes se conecten desde máquinas con direcciones fijas, pero impide que un usuario pueda utilizar el servidor desde cualquier lugar del mundo; para evitar esto se puede habilitar el control por la dirección de correo de origen. Para ello de nuevo hay que ir a `/etc/exim/exim.conf` y añadir las siguientes líneas:

```
relay_match_host_or_sender  
sender_address_relay = /etc/exim/sender_address_relay
```

Ahora creamos el archivo `/etc/exim/sender_address_relay` con las direcciones necesarias (una en cada línea), como por ejemplo `alumno@undominio.com`.

Por defecto exim se inicia bajo demanda controlado por `inetd` (recuerde la entrada en este archivo que vimos anteriormente), que es la configuración adecuada para poco tráfico. Para sistemas con mucho tráfico es mejor tener el servidor activo todo el tiempo; para ello basta con comentar (anteponiendo “#”) la línea del servicio `smtp` en `/etc/inetd.conf` e iniciar el demonio de forma convencional.

Existe una manera de comprobar que el servidor está funcionando a la espera de ser utilizado; para ellos debes utilizar la herramienta “telnet” sobre el puerto 25 (el que utiliza SMTP). “telnet” es una herramienta para establecer una comunicación con otra máquina; en este caso se utiliza sobre la máquina local en el puerto 25. Es una herramienta poco segura, pero útil a nivel de pruebas.

A terminal window titled "Terminal" with a menu bar containing "Archivo", "Editar", "Ver", "Terminal", "Ir a", and "Ayuda". The terminal output shows a telnet connection to localhost 25. It displays the connection status, the escape character, the server version (220 guadalinux ESMTP Exim 3.36 #1 Sat, 29 May 2004 13:07:16 +0200), and the supported commands (214-Commands supported: HELO EHLO MAIL RCPT DATA AUTH, NOOP QUIT RSET HELP VRFY).

```
guadalinux:/etc/init.d# telnet localhost 25
Trying ::1...
Trying 127.0.0.1...
Connected to guadalinux.
Escape character is '^]'.
220 guadalinux ESMTP Exim 3.36 #1 Sat, 29 May 2004 13:07:16 +0200
help
214-Commands supported:
214-   HELO EHLO MAIL RCPT DATA AUTH
214-   NOOP QUIT RSET HELP VRFY
```

Si el servidor está funcionando, veremos la imagen anterior, que es una consola de comandos desde la que se pueden enviar correos; se muestran también los comandos disponibles, tras haber introducido la orden “help”. Esta consola es plenamente funcional aunque bastante incómoda de utilizar, por lo que sólo debería usarse para comprobar el funcionamiento; mejor utiliza tu cliente de correo preferido, seleccionando como servidor de correo saliente tu recién configurado servidor.

CAPÍTULO 9

Configuración de un servidor web

Entre los usos más populares de Internet en segundo lugar nos encontramos con la navegación web, por detrás del correo electrónico. El servidor web más popular es con diferencia Apache; no solamente es gratis, un aspecto de máximo interés, sino que también es fuerte y rico en características. Además está disponible para cualquier plataforma.

El servidor web Apache tiene disponibles numerosas funcionalidades, que van aumentando con cada versión. Entre ellas están:

- Soporte para “cookies” (galletitas).
- Soporte para CGI (Common Gateway Interface, o Interfaz de Puerta Común).
- Autenticación de contraseña para las páginas o sitios web.
- Servidor proxy caché.
- Soporte de PHP y Perl (dos lenguajes de servidor para páginas dinámicas).
- Soporte de Java (a través de los proyectos “Jakarta” o “Java Apache Project”).
- Autenticación MySQL.
- Hosting virtual (varios sitios web en el mismo servidor).
- Soporte de XML (Extended Markup Language, o Lenguaje de Marcado Extendido).

La lista es mucho más extensa, todo gracias a que Apache ha sido concebido bajo un diseño modular; esto quiere decir que existe un núcleo central que gobierna el servidor, y una serie de módulos cada uno con una funcionalidad específica o un grupo de funcionalidades; aunque esto resulta en una ejecución más lenta, siempre se puede no ejecutar aquellos módulos que no necesitamos, por lo que al desactivar estos módulos sí obtenemos un rendimiento superior del servidor web.

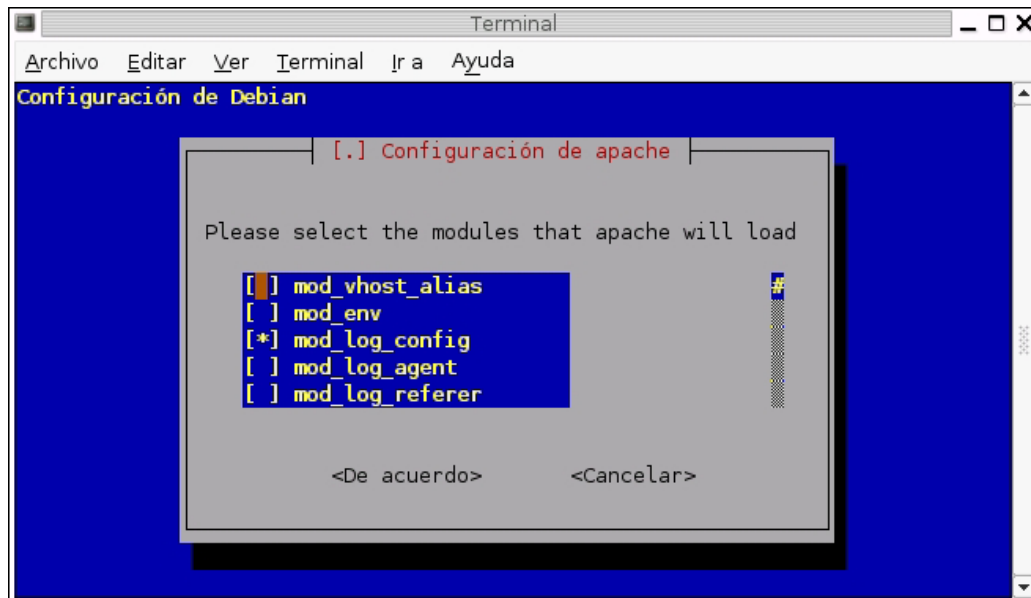
Para instalar Apache puedes utilizar apt (apt-get install apache) o synaptic; se instalarán varios paquetes para resolver las dependencias. Una vez concluida la descarga empieza la configuración con una serie de preguntas. Si alguna de estas

preguntas no aparece, siempre puedes ejecutar “dpkg-reconfigure apache”, porque depende de la versión el que te haga todas las preguntas o sólo algunas durante la instalación.

La primera nos pregunta si queremos ejecutar el servidor Apache en el arranque del ordenador; contesta según quieras, pero recuerda que siempre podrás volver a cambiar esta decisión desde la aplicación “Servicios”, ya que Apache se ejecuta a través de un demonio propio.

La siguiente consiste en activar la opción de “suExec”; esta opción se utiliza para que los usuarios ejecuten los CGI (que son programas ejecutables) con los permisos del propietario del CGI en lugar de los suyos propios. Sigue el consejo y selecciona “no”.

A continuación debemos seleccionar los módulos que Apache cargará cuando se inicie; por ejemplo en la imagen puedes ver seleccionado el módulo “mod_log_config, encargado de almacenar información de funcionamiento de Apache; más adelante veremos algunos más de ellos. Como Apache ya viene con una serie de módulos necesarios en la mayoría de los casos, pulsa el tabulador y “De acuerdo” (si sabes ya que vas a necesitar alguno que no está marcado puedes activarlo ahora).

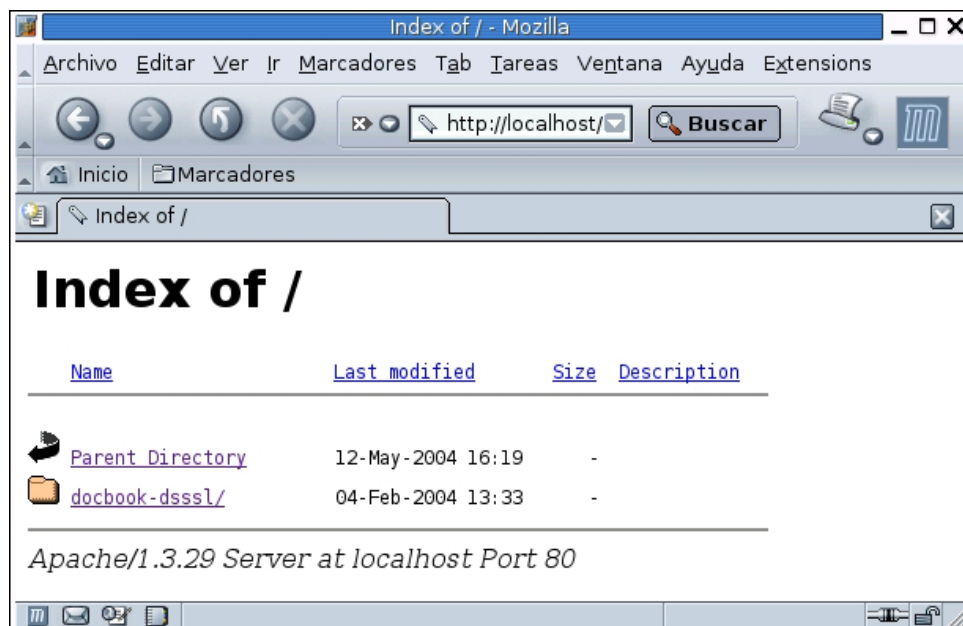


Debemos introducir ahora el nombre de nuestro dominio, si tenemos alguno;

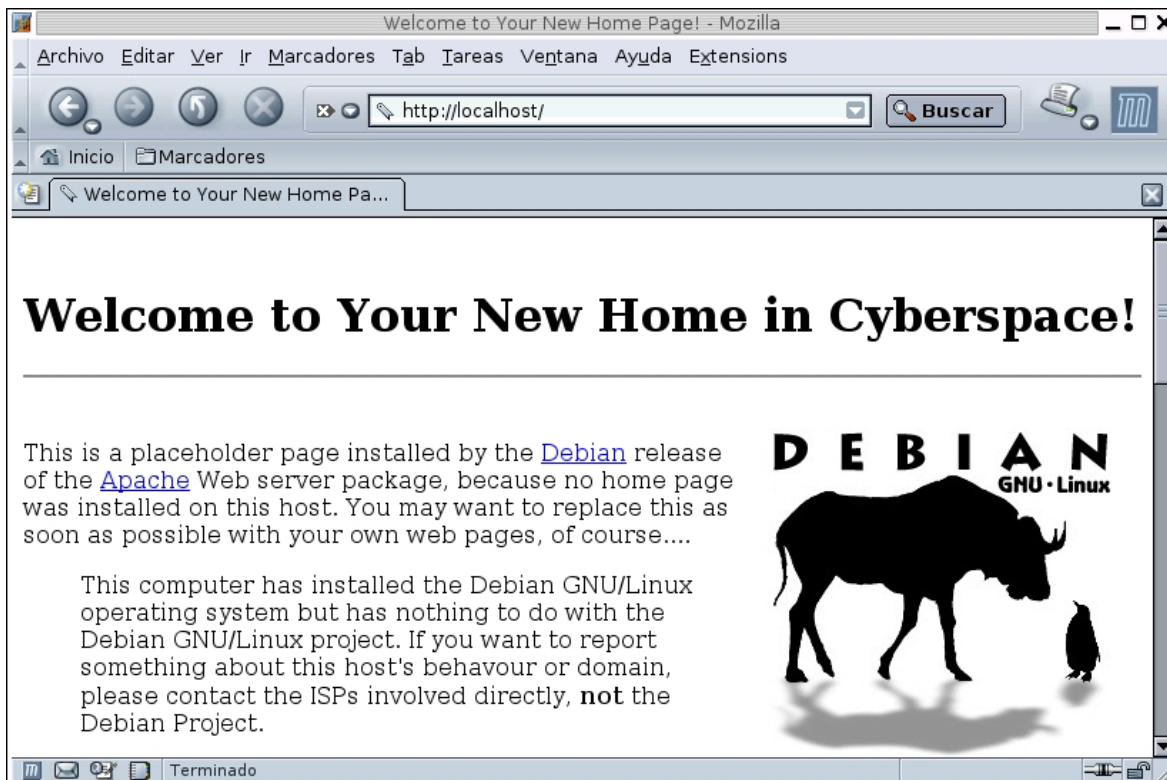
esto es lo que se conoce como FQND (Fully Qualified Domain Name, o nombre de Dominio Totalmente Cualificado); si no dispones de ninguno deja "localhost".

Lo siguiente que se necesita es la dirección del administrador de Apache, al que serán enviados los problemas en caso de haberlos; acto seguido hay que escribir la ruta local donde residirán las páginas web que Apache servirá; es buena idea dejar /var/www que es la ruta por defecto.

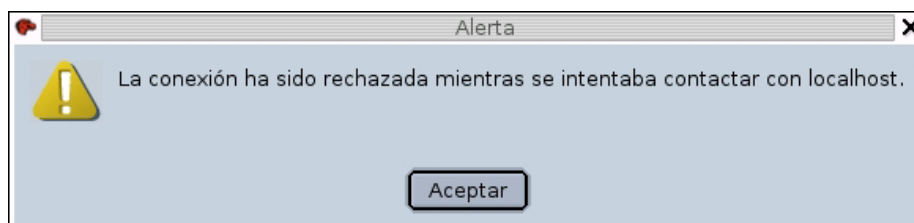
Por último nos pregunta el puerto en el que debe escuchar Apache; evidentemente debería ser el 80, a menos que tengas alguna buena razón para esconder tu servidor web (los navegadores siempre buscan en este puerto).



Ya puedes probar tu servidor apache; arranca tu navegador favorito y carga la dirección "localhost" (o tu nombre de dominio si lo tienes). Si utilizas Mozilla, deberías ver la imagen anterior.



Dependiendo de la versión de Apache podrías ver algo distinto, pero lo que nunca deberías ver es el mensaje de alerta; si es así, repite los pasos desde el principio, porque Apache se pone a funcionar en cuanto termina de instalarse.



Los archivos de configuración de Apache se encuentran en `/etc/apache`; hubo un tiempo en que se utilizaban tres archivos para configurar el servidor: `access.conf`, `httpd.conf` y `srm.conf`; el primero ofrecía el control de seguridad, el segundo controlaba el propio servidor, y el último de ellos seguía la pista de todos los elementos externos que necesitaba el servidor para cargar una página. Sin embargo las versiones más recientes centran toda la configuración en el archivo `httpd.conf` (los otros dos archivos siguen existiendo por compatibilidad con versiones anteriores, pero se recomienda dejarlos tal y como están).

Este archivo es muy extenso, pues dispone de muchas líneas de comentarios que sirven de descripción de las diferentes opciones. La primera de las directivas es “ServerType standalone”, que hace que funcione como demonio independiente; la otra opción es “inetd”, pero no es recomendable utilizar Apache así (además en la versión 1.3 está restringida esta opción a sistemas Unix).

A continuación se encuentran una serie de directivas para definir la localización de algunos archivos y directorios, que es mejor no modificar; a partir de “Timeout 300” es probable que si quieres modificar alguna de ellas, por ejemplo “StartServers 5” que es el número de servidores que se crearán al iniciar Apache, o “MaxClients 150”, que especifica el número máximo de peticiones HTTP simultáneas que aceptara el servidor (cuanto mayor sea más memoria se necesitará).

La directiva “Listen” se utiliza para definir puertos o IPs de escucha adicionales a los utilizados por defecto; junto con “BindAddress” se utiliza para definir espacios virtuales; los veremos más adelante.

Acto seguido aparecen un par de opciones antes de llegar a la sección 2, que define la configuración principal del servidor. La primera de las directivas está bastante clara pues es “Port 80”, pero siempre debes recordar que esta directiva no se utiliza si se definen sitios virtuales (a menos que añadir una entrada “Listen” para el puerto 80). Las dos siguientes definen al usuario y el grupo bajo el cuál se ejecutará Apache (por lo general “www-data”). Tras ellas aparecen las directivas que contienen la dirección del administrador (el responsable del servidor, no del administrador del sistema) y el nombre del servidor, así como el directorio donde se encuentran las páginas web (/var/www).

A continuación se configuran los parámetros para diferentes partes del sistema de archivos; la etiqueta “directory ruta” se utiliza para permitir o bloquear que una funcionalidad del servidor pueda acceder a dicha ruta del sistema de ficheros. Para toda la estructura del sistema (/) se establece no utilizar los enlaces simbólicos sólo si el destino es propiedad del que creó dicho enlace; es lo que hace “Options SymLinksIfOwnerMatch”; otras opciones son “FollowSymLinks”, con la que se permite seguir los vínculos simbólicos, “ExecCGI”, que permite que se ejecuten scripts CGI en esa ubicación, y “none” (ninguna opción activada).

Por otro lado “AllowOverride” habilita las funcionalidades de autorización (“AuthConfig”), tipo de documento (“FileInfo”), indexación (“Indexes”), o la elegida en éste caso, ninguna (“none”), lo que es muy conveniente para el

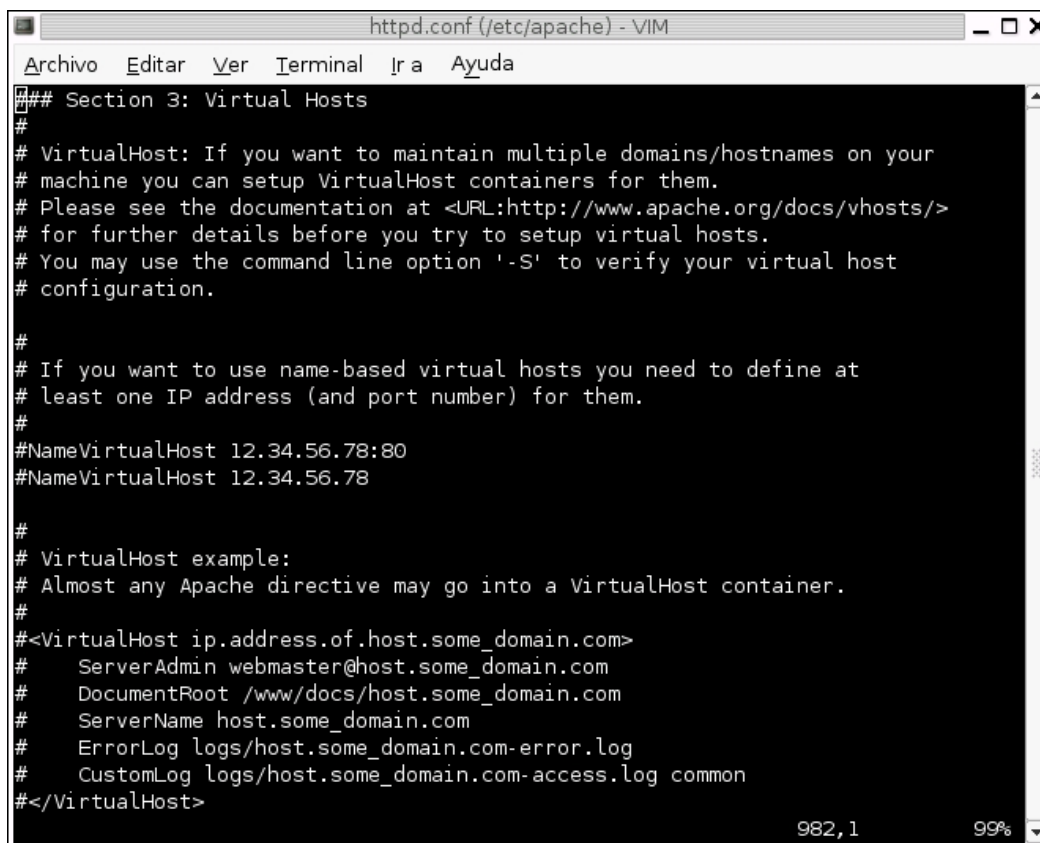
directorio raíz.

El siguiente directorio en aparecer es `/var/www`, que debe modificarse en el caso de haber definido otro en `"DocumentRoot"`, que está configurado adecuadamente para la mayoría de los casos, ya que sigue la política de tener todos los sitios cerrados inicialmente, y decidir uno a uno los que tienen permiso; observa que las directivas de este tipo terminan en `"</Directory>"`.

Así se definen varios directorios hasta llegar a la configuración del módulo `"mod_dir.c"`, en el que se define cuál es la página inicial de los sitios web (DirectoryIndex), es decir, el nombre de la página principal que será cargado al acceder al sitio; el primero es `"index.html"`, pero nada impide añadir delante `"home.html"` por ejemplo, o `"principal.html"`. Si no existe la primera, se busca la siguiente en la lista hasta que se encuentra una; si no coincide ninguna se carga la página de error.

El resto son opciones variadas como `"DefaultType text/plain"`, que le indica a Apache que si se encuentra con información de un tipo desconocido lo trate como texto, o los `"LogFormat"`, que definen el nivel de detalle de los logs o información que se almacenará sobre el funcionamiento del servidor.

Avanza en el archivo y verás por ejemplo cómo se definen las extensiones de archivos y cómo asociarles un icono en Apache (directivas `"AddIconByType"`, `"AddIcon"`, y `"AddDescription"`); el resto son configuraciones que se realizan si se va a proceder a cargar el módulo (`"IfModule"`).



```

httpd.conf (/etc/apache) - VIM
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them.
# Please see the documentation at <URL:http://www.apache.org/docs/vhosts/>
# for further details before you try to setup virtual hosts.
# You may use the command line option '-S' to verify your virtual host
# configuration.
#
# If you want to use name-based virtual hosts you need to define at
# least one IP address (and port number) for them.
#
#NameVirtualHost 12.34.56.78:80
#NameVirtualHost 12.34.56.78
#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
#
#<VirtualHost ip.address.of.host.some_domain.com>
#   ServerAdmin webmaster@host.some_domain.com
#   DocumentRoot /www/docs/host.some_domain.com
#   ServerName host.some_domain.com
#   ErrorLog logs/host.some_domain.com-error.log
#   CustomLog logs/host.some_domain.com-access.log common
#</VirtualHost>
982,1 99%

```

La última parte del archivo es la que permite definir espacios virtuales (“Virtual Hosts”); hay que definir una IP (o IP:puerto) para cada sitio virtual, lo que se realiza con la directiva “NameVirtualHost”. Una vez configurado esto deberás introducir cierta información relativa a cada sitio virtual que definas mediante directivas “<VirtualHost>”.

Vamos a realizar una configuración típica: un segundo sitio web en el puerto 8080 (además del estándar del 80); para ellos introducimos lo siguiente:

```
NameVirtualHost 192.168.0.1:8080
```

```

<VirtualHost 192.168.0.1:8080>
    ServerAdmin webmaster@midominio.com
    DocumentRoot /var/www2
    ServerName midominio.com
</VirtualHost>

```

Dentro de “<VirtualHosts>” se pueden introducir cualquier directriz de las

que hemos visto anteriormente; las que no se redefinen aquí toman el valor definido en la sección general (por ejemplo “DirectoryIndex” que podría haberse utilizado de nuevo).

No olvides definir la directiva “Listen” para el nuevo sitio virtual.

Recuerda que después de cualquier modificación en este archivos debes reiniciar el servidor.

Ejercicios

- 1) Crea un usuario llamado dominio; añade un sitio web virtual en su directorio bajo /home; este sitio deberá estar disponible en el puerto 8081 y la dirección de correo del responsable será dominio@undominio.com.
- 2) Copia un archivo html al directorio principal del sitio y haz que sea la página que se cargue al acceder al sitio web.

Soluciones

- 1) Ya sabes cómo crear un usuario; si no revisa el capítulo apropiado; ahora debes añadir casi al principio del archivo /etc/apache/httpd-conf:

Listen localhost:8081

Y al final del archivo:

NameVirtualHost localhost:8081

```
<VirtualHost localhost:8081>  
    ServerAdmin dominio@undominio.com  
    DocumentRoot /home/dominio  
</VirtualHost>
```

Si obtienes una ventana emergente con el mensaje “la conexión ha sido

rechazada" es que no está bien configurado; si por el contrario obtienes una página "Forbidden" (prohibido) pero con la versión de Apache abajo junto con el puerto, entonces es correcto.

- 2) Puedes coger cualquier html; debes definir dentro de la directiva "VirtualHost" anterior "DirectoryIndex nombre_de_archivo.html", o puedes renombrar el archivo a "index.html".

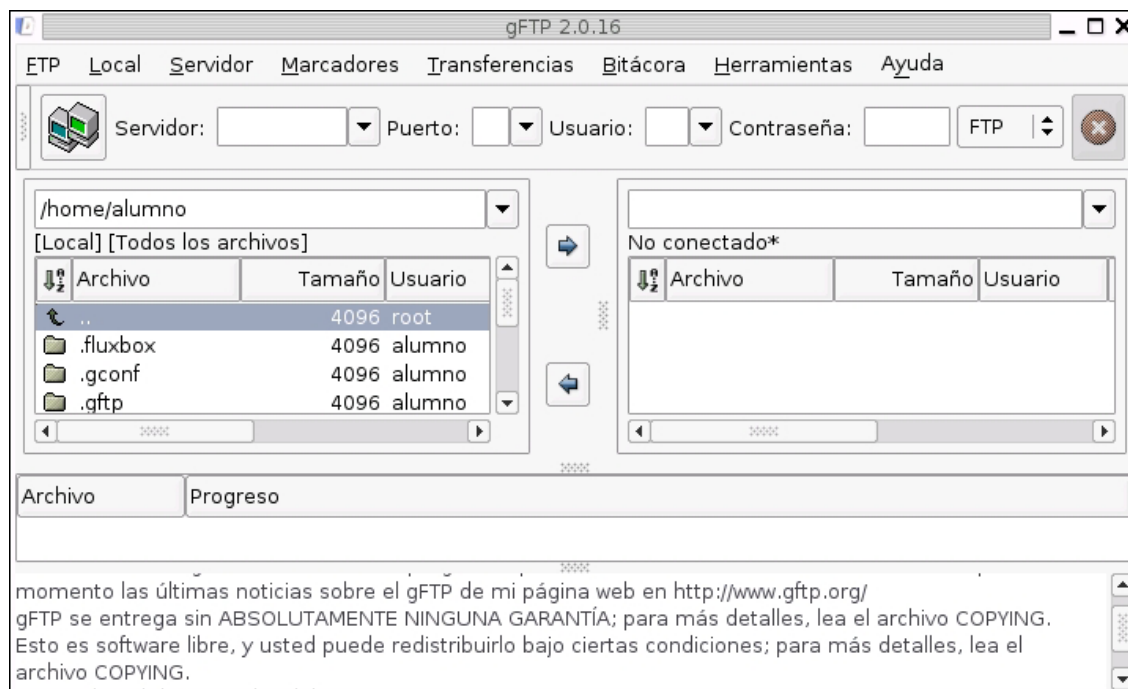
CAPÍTULO 10

Configuración de un servidor FTP

El servicio FTP (File Transfer Protocol, o Protocolo de Transferencia de Ficheros) permite que determinados usuarios puedan acceder a determinadas zonas de nuestro servidor donde podrán enviar y coger archivos.

Existen dos formas de utilizar FTP; una de ellas consiste en que los usuarios con cuentas de “shell” accedan a su propio sistema de archivos y carpetas (su directorio en /home) y puedan manipularlo; a esto se le conoce como “acceso FTP autorizado”, ya que el usuario utiliza su nombre de usuario y contraseña. La otra permite que cualquiera se conecte a una sección del sistema de ficheros y cargue o descargue información; esto es lo que se conoce como “acceso FTP anónimo”.

Todas las versiones de Linux vienen desde el principio con un cliente muy sencillo de FTP, llamado “ftp” y se puede utilizar desde la consola; además GuadaLinux dispone de una aplicación gráfica en Aplicaciones -> Internet llamada gFTP, más potente y fácil de utilizar y que se supone sabes manejar (aunque siempre deberías aprender cómo usar el de la consola “por si acaso”). Acude a su página de manual para aprender su funcionamiento.



Existen varios servidores de FTP; hasta hace poco el más utilizado era wu-
ftpd, pero tenía algunos problemas de seguridad; aunque esos problema se han
solventado ha crecido el uso del servidor ProFTPD, más sencillo, que además tiene
un diseño parecido al de Apache. Para instalarlo como de costumbre se puede
utilizar synaptic o apt (“apt-get install proftpd”); en ambos casos puede que se
instale algún paquete adicional para resolver dependencias.

La única pregunta que nos realizará la instalación es, como el resto de
servicios que hemos visto hasta ahora, si queremos ejecutar el demonio de forma
independiente (“StandaAlone”) o a través del superdaemon inetd. Mi consejo es
que mientras estés configurándolo lo hagas de forma independiente, ya que
puedes evitarte quebraderos de cabeza debidos a problemas de acceso con los
“TCP wrappers”.

En la versión que hemos probado la instalación fallaba debido a que
buscaba un grupo que no existía; para solucionarlo debes crear un grupo con el
GID 65534 llamado “nogroup”; puedes hacerlo desde la aplicación gráfica o desde
la línea de comandos con “addgroup --gid 65534 nogroup”. Para volver a ejecutar
la instalación puedes ejecutar “dpkg-reconfigure proftpd”

```
Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:~# addgroup --gid 65534 nogroup
Añadiendo el grupo nogroup (65534)...
Hecho.
guadalinux:~# dpkg-reconfigure proftpd
Stopping ProFTPD ftp daemon: proftpd.
Añadiendo usuario del sistema ftp...
Añadiendo nuevo usuario ftp (103) con grupo nogroup.
Creando el directorio home $.
`/usr/share/doc/proftpd/examples/welcome.msg' -> `/home/ftp/welcome.msg.proftpd-
new'
Starting ProFTPD ftp daemon: proftpd.
guadalinux:~#
```

Tras la instalación el demonio se cargará en memoria y estará dispuesto a aceptar conexiones; el archivo de configuración es `/etc/proftpd.conf` y las opciones más relevantes que presenta son:

ServerName:	Nombre del servidor (el que aparece cuando te conectas).
ServerType:	Modo de funcionamiento (standalone o inetd).
ShowSymlinks	Muestra los enlaces simbólicos para que sean accedidos.
AllowOverwrite	Permite la escritura, es decir, subir o borrar archivos.
DisplayLogin	Mensaje de bienvenida (está en <code>/home/ftp</code>).
Port	Puerto para escuchar conexiones entrantes (por defecto el 21).
MaxInstances	Número máximo de conexiones simultáneas (sólo standalone).
User/Group	Usuario/grupo para ejecutar el demonio (nobody/nogroup).
Umask	Permisos que tendrán los archivos subidos.
AllowOverwrite	Indica si archivos existente pueden ser sobreescritos (on/off).

Al igual que Apache pueden definirse directivas sobre directorios; por ejemplo siempre aparece la siguiente, que establece los permisos `rw-r--r--` para los nuevos archivos, y permite que los archivos sean sobreescritos por los nuevos.

```
<Directory /*>
    Umask 022
    AllowOverwrite on
</Directory>
```

Si queremos habilitar el acceso anónimo debemos definir una directiva `<Anonymous ~ftp>`, donde especificar el usuario/grupo de este usuario además de otras opciones; un caso típico sería :

```
<Anonymous ~ftp>
```

```
User          ftp
Group         nogroup
UserAlias      anonymous ftp
RequireValidShell off
MaxClients    10
DisplayLogin   welcome.msg
DisplayFirstChdir .message
```

```
<Directory *>
  <Limit WRITE>
    DenyAll
  </Limit>
</Directory>
```

</Anonymous>

Esto permitirá a usuarios anónimos con nombre de usuario “ftp” o “anonymous” acceder para leer los archivos, pero no para escribir, debido a la directiva “Limit WRITE”. Los usuarios no tienen por qué tener cuenta en nuestro sistema al estar “RequireValidShell” a “off”. También se limita el número de accesos anónimos a 10.

El directorio inicial para este tipo de accesos es /home/ftp, ya que durante la instalación se creó un usuario (sin “shell”, puedes comprobarlo) llamado “ftp”, siendo éste su directorio principal.

Si además queremos que estos usuarios puedan subir archivos, habría que añadir antes de cerrar la directiva <Anonymous>:

```
<Directory incoming>
  <Limit READ WRITE>
    DenyAll
  </Limit>
  <Limit STOR>
    AllowAll
  </Limit>
</Directory>
```

Para que esto funcione debes crear el directorio /home/ftp/incoming y ponerle los permisos apropiados:

```
# mkdir /home/ftp/incoming  
# chown ftp:nogroup /home/ftp/incoming
```

Ejercicios

- 1) Limita el número de conexiones simultáneas a 10; activa el acceso anónimo para lectura y escritura y limita el número de estas conexiones a 4.
- 2) Modifica la configuración anterior para eliminar el acceso anónimo de escritura.

Soluciones

- 1) Tan sólo debes repetir el ejemplo propuesto y modificar las opciones:
MaxInstances 10
MaxClients 4
- 2) Elimina (o mejor añade delante de cada línea el símbolo “#”) la sección <Directory incoming>

CAPÍTULO 11

Configuración de un Proxy

Los usuarios de un sistema navegan por la red de dos formas distintas; el primer método consiste en utilizar el navegador de Internet para solicitar páginas directamente desde servidores web remotos (método directo).

Pero existe otro modo que implica el almacenamiento o “cacheo” (caching) de la web. Siempre que un usuario quiera visitar una página web, mejor que hacer que su cliente vaya al servidor remoto, el cliente solicita la página al servidor de caché; el servidor entonces mira a ver si la página ya ha sido descargada. Si es así, la página estará almacenada en la caché del servidor o almacén de datos. El servidor comprobará si hay disponible una nueva versión de la página y si no existe, pasará la versión que tiene guardada al navegador cliente.

Proporcionar un servicio como éste puede acelerar enormemente el tiempo de descarga de una página web, ya que sólo aquellas que es necesario se descargan realmente de nuevo. Por otro lado las descargas de las páginas que todavía no han sido almacenadas no se ralentizan.

El programa Squid es un servidor proxy con dos propósitos básicos; una la de proporcionar servicio proxy a máquinas que comparten el acceso a Internet; la otra es la de almacenar. Los navegadores actuales suelen proporcionar este servicio pero a cada usuario independientemente; squid lo hace para todo el que pueda acceder a la red. Squid puede actuar de proxy para los protocolos FTP, HTTP, HTTPS y DNS.

Lo primero que debes hacer es instalarlo, porque no viene en GuadaLinux; el paquete se llama “squid” y puedes instalarlo por ejemplo mediante apt. Tras la instalación aparecerá el script que controla el demonio en /etc/init.d y el archivo de configuración, llamado /etc/squid.conf; es un archivo relativamente largo, pero la mayoría son comentarios (en inglés) de las opciones.

Es posible que aparezca un error y no arranque el servidor; no te preocupes,

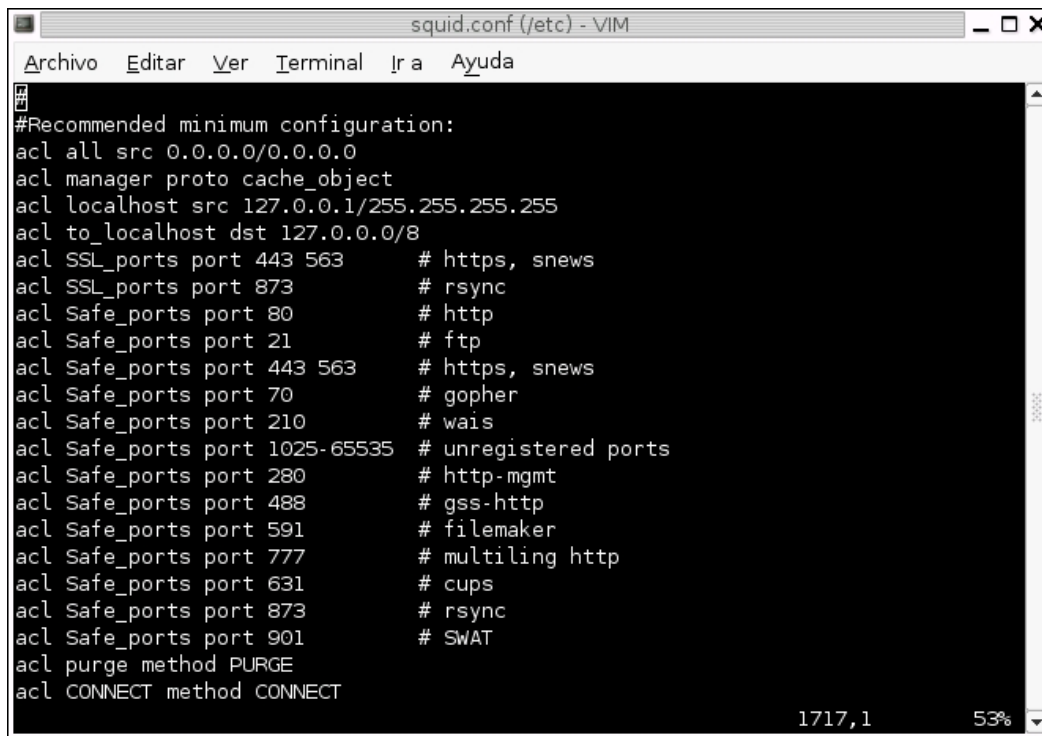
más adelante veremos cómo solucionar esto (el error te avisa sobre definir algo llamado “visible_hostname”).

Editando el archivo se observa que todos las directivas de control de los servicios están deshabilitadas; esto es así porque squid funciona con unos puertos predeterminados para cada uno de los servicios soportados (si se especifica, porque puede aparecer “none” si no está activado por defecto). En el caso de coincidir con otro puerto que se esté utilizando hay que borrar el símbolo “#” e introducir el número de puerto a utilizar; por ejemplo para utilizar el puerto 3128 (el que se usa mientras no se diga lo contrario) hay que poner en la línea correspondiente:

```
http_port 3128
```

Y a partir de entonces se ofrece el servicio proxy para http en ese puerto. Ten siempre presente cuando cambies un puerto comprobar que no se utiliza en algún servicio; consulta el archivo /etc/services para no tener ninguna duda; también recuerda evitar los números de puertos inferiores al 1024 porque suelen estar reservados.

El archivo tiene muchas opciones, pero verdaderamente interesante resulta comentar el sistema de control de acceso, que es realmente complejo y potente; su funcionamiento se basa en unas listas de control de acceso o ACL (Access Control List). Para acceder a esta sección debes ir a la línea 1590 aproximadamente (aunque dependerá de la versión que instales), donde se define la directiva “acl”; existen algunas ya creadas como puedes ver en la imagen.



```

#Recommended minimum configuration:
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl to_localhost dst 127.0.0.0/8
acl SSL_ports port 443 563      # https, snwews
acl SSL_ports port 873         # rsync
acl Safe_ports port 80         # http
acl Safe_ports port 21         # ftp
acl Safe_ports port 443 563    # https, snwews
acl Safe_ports port 70         # gopher
acl Safe_ports port 210        # wais
acl Safe_ports port 1025-65535 # unregistered ports
acl Safe_ports port 280        # http-mgmt
acl Safe_ports port 488        # gss-http
acl Safe_ports port 591        # filemaker
acl Safe_ports port 777        # multiling http
acl Safe_ports port 631        # cups
acl Safe_ports port 873        # rsync
acl Safe_ports port 901        # SWAT
acl purge method PURGE
acl CONNECT method CONNECT

```

La estructura de esta directiva es:

`acl nombre_lista tipo_lista elemento`

El valor de `nombre_lista` es el que le vamos a dar a esa lista; puede ser cualquiera, pero suele identificar al grupo que queremos controlar; el `tipo_lista` es el tipo de elementos a incluir en la lista, y puede ser entre otros uno de los siguientes:

<code>dst</code>	especifica la IP de la máquina o red de destino (con o sin máscara).
<code>dstdomain</code>	nombre de dominio del destino.
<code>port</code>	indica los puertos a buscar.
<code>proto</code>	protocolo a buscar
<code>src</code>	especifica la IP de la máquina o red de origen (con o sin máscara).
<code>srcdomain</code>	nombre de dominio del destino.
<code>time</code>	tiempo a buscar.

Este último se expresa con una letra que indica el día de la semana seguido de `hora1:minuto1-hora2:minuto2`; las letras para los días son S (Domingo), M (Lunes), T (Martes), W (Miércoles), H (Jueves), F (Viernes) y A (Sábado).

Como ejemplos, para crear una lista para acceder los lunes de 9 a 2:

```
acl    Mañanas          time      M      9:00-14:00
```

Y para crear otra para aplicarla a los puertos web y ftp:

```
acl    DosPuertos  port      80 21
```

Para realizar alguna acción sobre los elementos de una lista, se utiliza la directiva `http_access`, seguida de la acción a tomar (permitir o “allow”, y denegar o “deny”); si quisiéramos permitir el uso de los puertos web y ftp pero no por las mañanas:

```
http_access  allow      DosPuertos
http_access  deny       Mañanas
```

Hay que hacer un par de precisiones aquí; la primera es que el orden es muy importante, ya que las listas de control (directivas `http_access`) se aplican en orden consecutivo. En el ejemplo anterior efectivamente se permite utilizar el proxy para los puertos 80 y 21, pero se deniega su uso los lunes por la mañana; si hubiéramos cambiado el orden primero se denegaría el uso del proxy los lunes por la mañana (de todo el proxy), mientras que se permitiría usar esos puertos en otro momento; la diferencia es que otro puerto (por ejemplo el 443, que es el que utiliza https) no podría ser usado tampoco los lunes por la mañana, ya que la denegación no se realiza por puertos sino por horario.

La segunda observación es más evidente: un servidor squid bien configurado puede evitar el uso de un cortafuegos. El caso típico es una máquina conectada a Internet y a la red local dedicada únicamente a tareas de proxy.

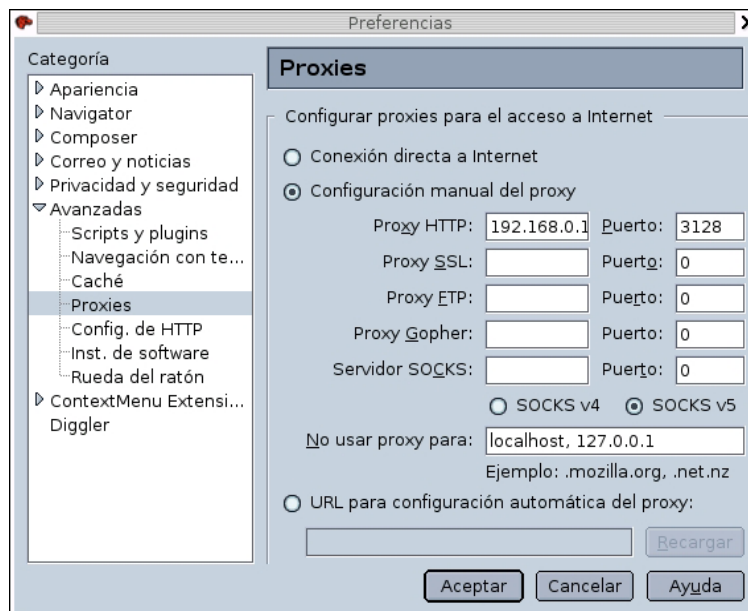
Si no lo has hecho ya define la directiva “`visible_hostname`” con el nombre completo de tu dominio. Otras opciones que quizá te gustaría modificar son:

- `cache_mem` 8 MB: cantidad de memoria Ram que se utilizará para uso interno.
- `maximum_object_size` 4096 KB: no almacenar objetos con un tamaño mayor.
- `fqdn_cache_size` 1024: cantidad de nombres de dominio que se almacenan.
- `cache_dir` /var/spool/squid 100 16 256: por defecto se usa ese directorio y 100 MB de espacio para la caché.

Squid posee características avanzadas entre las que se encuentran:

- Posibilidad de limitar los recursos a utilizar en la máquina (memoria, espacio en disco, procesos, ...)
- Posibilidad de configurar políticas de reemplazos de los datos en la almacenados en la caché.
- Puede utilizarse en una estructura jerárquica, con otras máquinas con squid también instalado.
- Es posible utilizarlo para balancear la carga de un sitio web (redirigir peticiones a diferentes máquinas para no colapsar ningún servidor).

Para poder utilizar el servidor proxy, lo único que tienen que hacer las máquinas clientes es comunicarle al navegador la dirección IP del servidor y el puerto que debe utilizar. Por ejemplo en Mozilla está en el menú Editar -> Preferencias -> Avanzadas -> Proxies.



Ejercicios

- 1) Crea una lista de acceso para utilizar el servicio web los días laborables de 4 a 7 de la tarde.
- 2) Establece un tamaño máximo de utilización del disco de 150 MB

Soluciones

1) Deberás crear las siguiente entradas:

```
acl ejercicio time M 16:00-19:00
acl ejercicio port 80
http_access allow ejercicio
```

2) Simplemente modifica esta línea y sustituye el 100 por 150:
cache_dir /var/spool/squid 150 16 256

CAPÍTULO 12

Sistemas de ficheros en red: NFS

El sistema de ficheros de red o NFS (Network File System) se utiliza para hacer accesible información a través de la red. Permite montar particiones o directorios de otras máquinas de la misma forma que se monta un disquete o un CD-ROM.

NFS es un servicio; aunque es un programa diseñado bajo arquitectura cliente/servidor, no trata con un único servidor y muchos clientes. Cada máquina que contiene un segmento del sistema de ficheros que desea compartir se convierte en un servidor NFS. Normalmente todas las máquinas de la red tienden a ser clientes NFS.

El archivo de configuración de NFS es `/etc/exports`, que controla las porciones de un sistema de ficheros de una máquina que están disponibles como exportaciones de NFS y quién puede acceder a ellas. Este archivo consiste en una serie de sentencias de una única línea, cada una de las cuales define un elemento particular a ofrecer y especifica a quién ofrecerlo.

Todas estas sentencias tienen el formato “ruta reglas”; la ruta simplemente es la ruta al directorio que quiere exportar. La parte de las reglas es la sustancia de la definición de la exportación; se descompone en dos partes: quien puede acceder a la información y cuáles son las directrices que tiene ese usuario o sitio (estas últimas entre paréntesis y separadas por comas). Por lo tanto la sentencia de exportación se parece a la siguiente, una vez descompuesta:

ruta	quién (cómo)
------	--------------

La forma de designar el “quién” tiene permiso de acceso puede ser un nombre de máquina, una IP o una pareja IP/máscara. Por ejemplo para exportar el directorio `/home` para que esté disponible en toda la red `192.168.0.0`:

<code>/home</code>	<code>192.168.0.0/255255255.0</code>
--------------------	--------------------------------------

El “cómo” permite configurar diversas opciones a la hora de permitir el acceso:

all_squash	todos los usuarios que accedan serán considerados anónimos
anongid=xxx	asigna el gid xxx al usuario anónimo
anonuid = xxx	asigna el uid xxx al usuario anónimo
async	el servidor escribe los datos cuando cree conveniente
insecure	permite que las solicitudes lleguen por cualquier puerto
no_root_squash	las peticiones de root se asignan a root en la máquina servidor
ro	configura la exportación en modo de sólo lectura
root_squash	asigna a las peticiones de root un UID y GID anónimos
rw	configura la exportación en modo lectura/escritura
secure	las peticiones deben llegar por un puerto inferior al 1024
sync	el servidor pone los datos en el disco de forma inmediata

Por ejemplo, para exportar el directorio /usr y permitir el acceso en nuestra red local en modo de sólo lectura, que los datos se escriban cuando el servidor lo considere oportuno y se utilice el modo “seguro” deberemos poner en el archivo exports:

```
/usr      192.168.0.0/255.255.255.0 (ro,async,secure)
```

Para poder utilizar este directorio en nuestra máquina habría que ejecutar (suponiendo que el servidor se llama “guadalinux”):

```
mount -t nfs guadalinux:/usr/local /usr
```

De esta forma montaríamos el directorio del servidor en nuestra ruta local /usr. No es necesario escribirlo cada vez que se inicie el sistema, podemos incluir una línea en el archivo /etc/fstab para que se monte automáticamente en cada inicio. La línea para este caso particular sería:

```
guadalinux:/usr /usr nfs opciones
```

Las opciones son específicas para el sistema de ficheros NFS; las más útiles son:

rsize=8192, wsize=8192: especifica un tamaño de búffer de 8192 en lugar de los 4096 que se utilizan por defecto; acelerará la velocidad de las conexiones.

hard: el programa que accede a un fichero al sistema de ficheros se quedará bloqueado si el servidor cae por alguna razón, y el proceso no podrá detenerse de ninguna manera. Cuando el servidor vuelve a estar operativo la transferencia continúa desde donde estaba. Deberías poner esta opción.

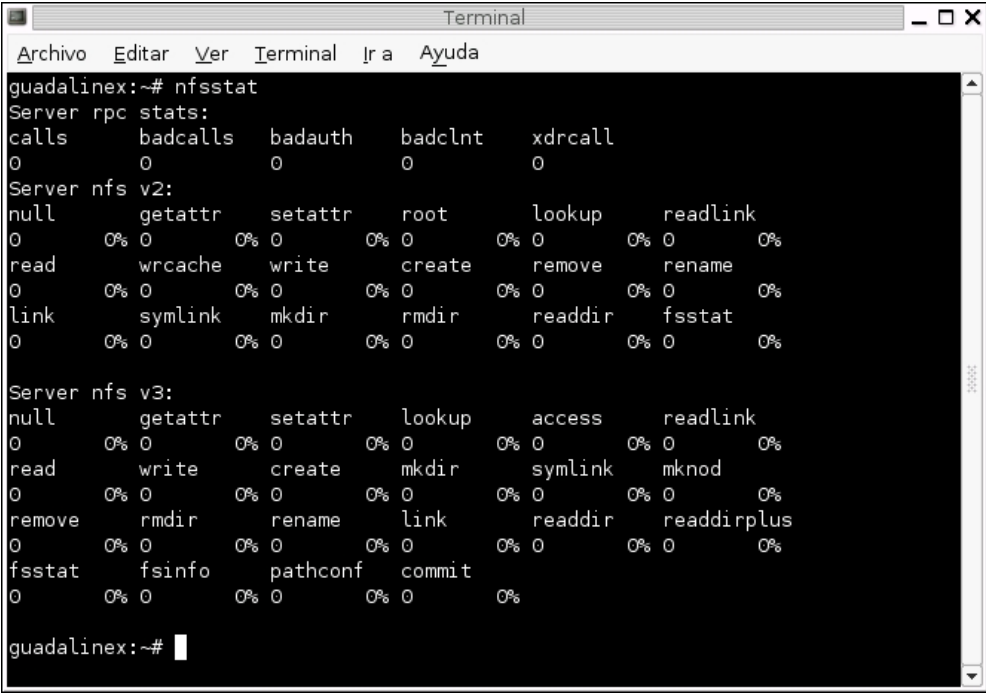
soft: permite que el núcleo cierre la conexión con el servidor si éste no responde durante algún tiempo; generalmente es fuente de problemas.

De esta forma la línea que debería ponerse en `fstab` con las opciones recomendadas es:

```
guadalinux:/usr /usr nfs rsize=8192,wsiz=8192,hard 0 0
```

Para que el servicio esté disponible, debes iniciar los servicios “nfs-common” y “nfs-kernel-server”.

Existe un comando llamado `nfsstat`, que proporciona información sobre el sistema de archivos compartido; la salida es como la de la imagen (en este servidor no se ha realizado ningún acceso todavía).



```
Terminal
Archivo  Editor  Ver      Terminal  Ir a  Ayuda
guadalinux:~# nfsstat
Server rpc stats:
calls      badcalls   badauth    badclnt    xdrcll
0           0           0           0           0
Server nfs v2:
null       getattr    setattr    root        lookup      readlink
0          0% 0        0% 0        0% 0        0% 0        0% 0
read       wr-cache   write       create      remove      rename
0          0% 0        0% 0        0% 0        0% 0        0% 0
link       symlink    mkdir       rmdir       readdir     fsstat
0          0% 0        0% 0        0% 0        0% 0        0% 0
Server nfs v3:
null       getattr    setattr    lookup      access      readlink
0          0% 0        0% 0        0% 0        0% 0        0% 0
read       write      create      mkdir       symlink      mknod
0          0% 0        0% 0        0% 0        0% 0        0% 0
remove     rmdir      rename      link        readdir     readdirplus
0          0% 0        0% 0        0% 0        0% 0        0% 0
fsstat     fsinfo     pathconf    commit
0          0% 0        0% 0        0% 0
guadalinux:~#
```

Ejercicios

- 1) Exporta el directorio /lib con las opciones que consideres más seguras; se supone que existe un servidor llamado “guadaserver” y que tu IP es la 192.168.30.40.
- 2) Escribe la línea que permite montar el directorio de este servidor desde otra máquina de la red.

Soluciones

- 1) La línea en /etc/exports de la máquina “guadaserver” debería ser:

/lib 192.168.30.40/255.255.255.0(all_squash, ro, root_squash, secure, sync)

- 2) Para montar el directorio, en /etc/fstab de nuestra máquina

guadaserver:/lib /lib rsise=8192, wsize=8192,hard 0 0

CAPÍTULO 13

Comunicación con redes Microsoft (Samba)

Samba es una herramienta útil para cualquiera que necesite integrar máquinas que ejecuten múltiples sistemas operativos. Aunque compartir una conexión entre múltiples sistemas operativos es relativamente sencillo cuando todos utilizan TCP/IP como protocolo de red, conseguir que cada una de estas máquinas lea los archivos de las otras y utilice las impresoras compartidas no es tan fácil; a menos claro está que utilices Samba.

Samba es la ejecución de Linux del protocolo SMB (Server Message Block, o Servidor de Bloques de Mensajes); este protocolo se utiliza para permitir la interacción del sistema de archivos y de la impresora a través de redes con múltiples sistemas operativos. Entre estos sistemas se encuentran todos los Linux y Unix, los Macintosh o los Microsoft Windows. Las principales acciones que permite Samba son:

- Servicios de archivos e impresoras; esto quiere decir que se pueden montar directorios Linux en otros sistemas operativos y utilizar sus impresoras, pero también a la inversa.
- Servicios de autenticación (validar el acceso de usuarios con nombre y contraseña).
- Servicios de resolución de nombres (utilización de nombres dentro de la red).
- Servicios de listado o “browsing” (conocer qué máquinas están disponibles en la red).

Existen dos demonios diferentes que proporcionan servicio SMB a una máquina Linux: `smbd` y `nmbd`; cada uno de estos demonios tiene una función específica que, unidas, proporcionan todo lo que se necesita para ejecutar Samba.

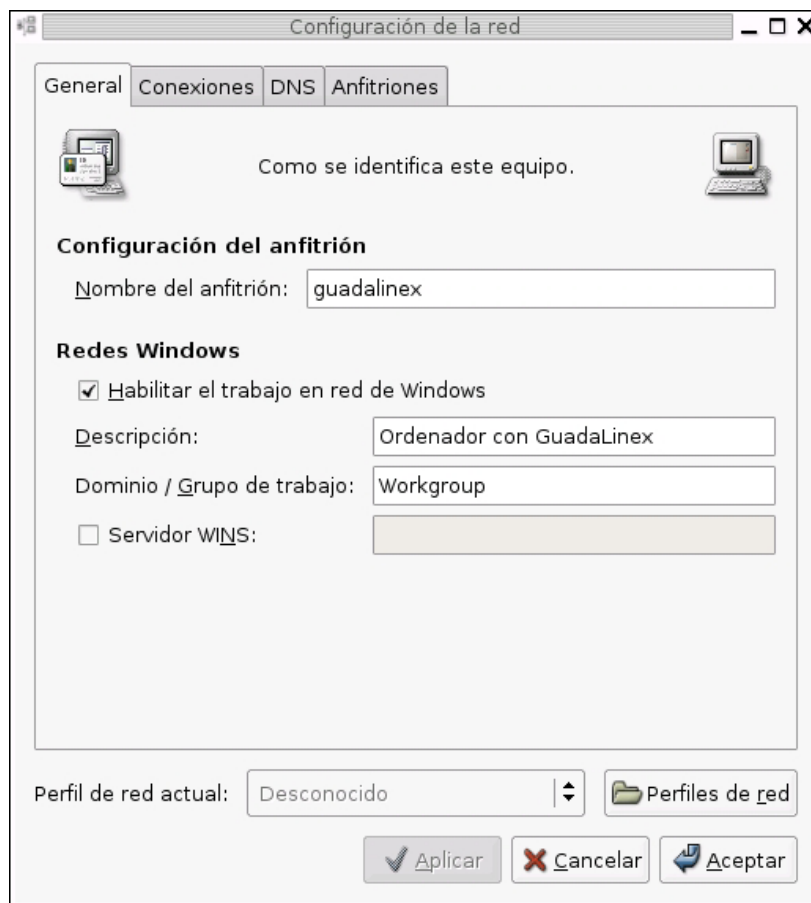
El demonio “`smbd`” es el servicio que gestiona la compartición de archivos e impresoras; recibe solicitudes de clientes y reacciona de acuerdo a la configuración

del servidor. Por otro lado “nmbd” gestiona la traducción de nombres NetBIOS (Network Basic Input/Output System, Sistema de Entrada/Salida Básico de Red), que es un sistema para nombrar máquinas que permite que éstas sean reconocidas mediante un nombre en lugar de tener que conocer en qué tipo de red se encuentra la máquina. Otro sistema que ofrece nmbd es el Servicio WINS (Windows Internet Name Service, o Servicio de Nombres de Internet de Windows), que puede considerarse como una especie de servidor de nombres para NetBIOS.

Samba ya viene instalado en GuadaLinux; pero si lo reinstalas o actualizas de versión la instalación va a preguntarte si quieres ejecutar los demonios de forma independiente o desde inetd; sigue la recomendación y selecciona la ejecución como demonios independientes.

También se nos pregunta si vamos a utilizar contraseñas cifradas, lo que es más que recomendable; además hay que tener en cuenta que los usuarios de Samba son distintos de los usuarios del sistema (aunque los nombres pueden coincidir); lo más inteligente es responder afirmativamente, lo que creará un archivo para almacenar los usuarios y contraseñas de Samba.

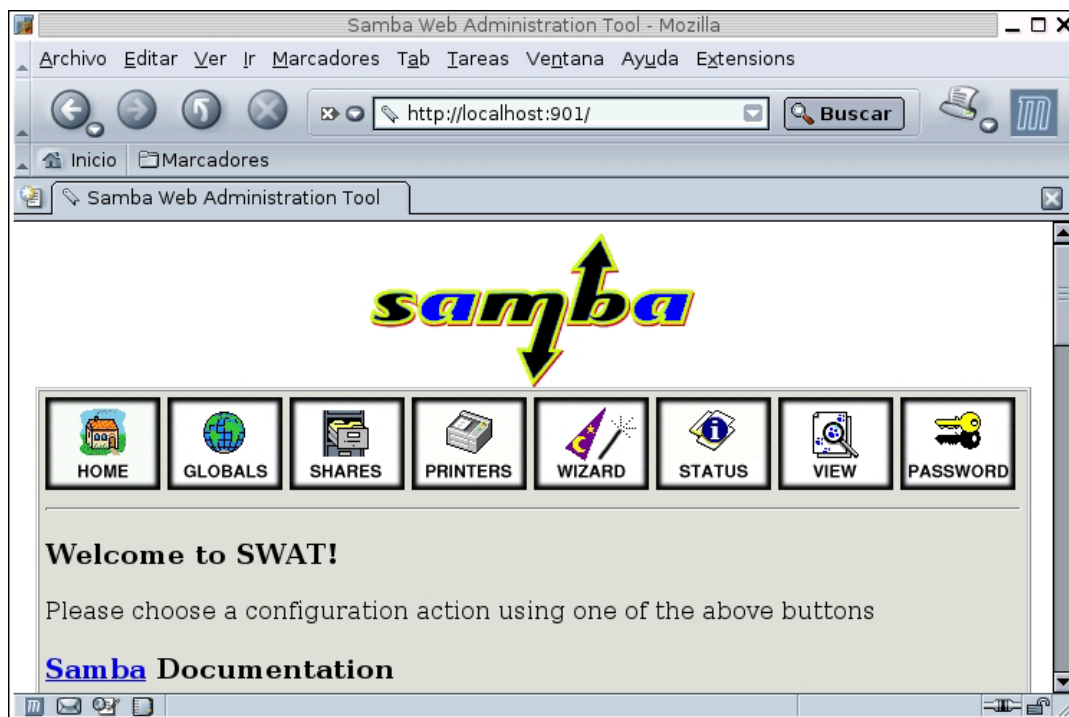
Puedes ejecutar la aplicación de “Red” e introducir una descripción y un grupo de trabajo o dominio de la red. El servidor WINS no es necesario que lo actives, a no ser que tengas uno funcionando en tu red, claro está (si no estás seguro es que no lo necesitas).



El directorio donde residen los archivos de configuración es `/etc/Samba`; en él se encuentran los tres archivos que configuran Samba; el principal es `smb.conf`. La configuración de este archivo es tediosa realizarla a mano, ya que a diferencia por ejemplo del `httpd.conf` de Apache el archivo está casi vacío, por lo que habría que introducir las diferentes opciones, en lugar de eliminar el carácter “#”.

Para solventar este problema vamos a utilizar un programa gráfico llamado “swat”; para instalarlo puedes usar `apt` (`apt-get install swat`), que resolverá las dependencias de forma adecuada; la única ventana que nos muestra el programa de instalación es una advertencia de que el archivo `smb.conf` va a ser modificado y sobrescrito, para que realicemos una copia de seguridad del archivo con las modificaciones que tengamos ya hechos; tan sólo acepta.

Este programa tiene una forma curiosa de ejecutarse, ya que funciona como si fuera un servicio dependiente de `inetd`; tras la instalación está desactivado, por lo que edita el archivo `inetd.conf` y elimina el “`#<off>#`” que hay en la última línea. Una vez hecho reinicia el demonio `inetd` y abre tu navegador de Internet favorito; “swat” se ejecuta como un servidor local en el puerto 901.



Debes introducir el nombre de usuario “root” y su contraseña; la pantalla principal (“home”) muestra el menú disponible, y está visible en todo momento; el resto de la página muestra los enlaces a la documentación existente referente a los demonios, los archivos de configuración, utilidades de administración, herramientas clientes, utilidades de diagnóstico, ... aunque está en inglés, puede servirte como primera referencia ante una duda puntual.

La primera sección es “globals”, y permite establecer las opciones generales de Samba; podemos elegir la configuración básica (recomendada) o la avanzada. Junto a cada opción aparece un enlace a una pequeña descripción de esa opción. Ya debes tener configuradas algunas, como “workgroup”, el nombre de NetBios y la descripción (“server string”). El grupo de trabajo se utiliza para permitir que Samba emule el funcionamiento de un servidor windows, permitiendo que los usuarios se autentifiquen con su nombre de usuario y contraseña.

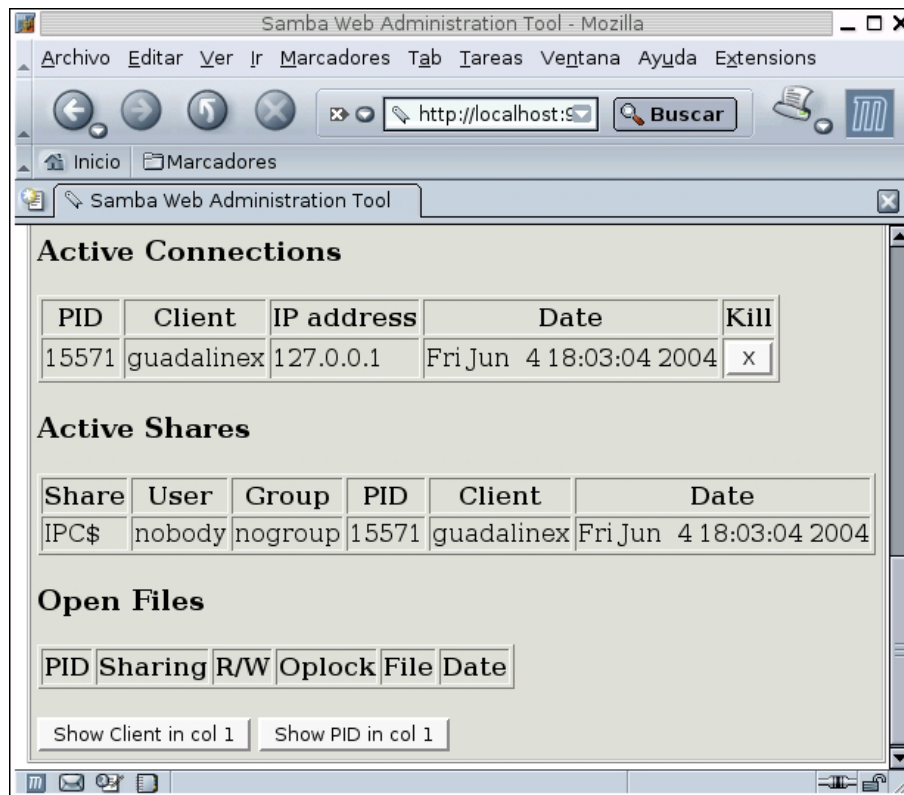
En las opciones de seguridad podemos elegir entre el nivel usuario (cada usuario debe validar su acceso mediante su login y password de Linux) que es el nivel seleccionado por defecto (y recomendado), y nivel compartido o “share” (para accesos sin contraseña); “guest account” es el nombre de usuario para accesos anónimos. El resto de opciones se usan raramente.

La pantalla “Shares” contiene la página para gestionar los directorios compartidos.; para crear uno debes introducir un nombre para identificarlo y pulsar en “Create Share”, podrás configurar varios elementos; el primero y más importante es “path”, que es la ruta a compartir; pueden especificarse usuarios a los que se les permite el acceso (“valid users”) o se les deniega (“invalid users”). Una vez hechas las modificaciones oportunas debes pulsar “Commit Changes”. Puedes crear todos los directorios compartidos que quieras.

El menú “Printer” contiene la página de gestión de impresoras compartidas; sus opciones son similares a la anterior; la variable “path” debe ponerse con el valor “/tmp”.

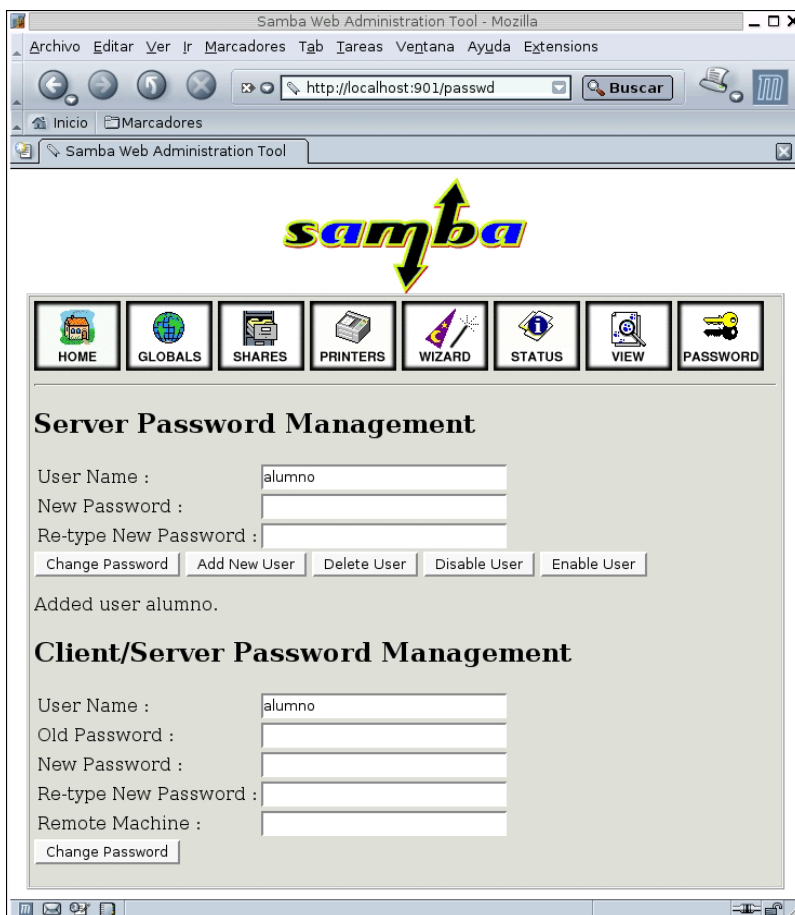
“Wizard” es un asistente de configuración rápido, donde puede elegirse la forma de ejecutar los demonios, la utilización de WINS y si el directorio local del usuario es navegable o no. Una vez hechos los cambios, debes pulsar “Commit”.

Con el botón “Status” se muestra el estado del servidor y permite manipular los demonios (activarlos, detenerlos o reiniciarlos); se muestran las conexiones activas, los elementos compartidos y los archivos abiertos.



“View” muestra el fichero de configuración utilizado, mostrando las opciones más relevantes.

“Password” permite cambiar la clave de acceso y dar de alta nuevos usuarios; estos valores se almacenarán en el archivo smbpasswd. Las opciones para los usuarios son, de izquierda a derecha: cambiar contraseña, añadir usuario, borrar usuario, deshabilitar usuario y habilitar usuario. La parte inferior permite modificar un usuario ya creado.



Para poder utilizar recursos exportados con Samba debes utilizar el comando: `"smbclient //nombre_máquina/recurso -U usuario"`, siendo este último el nombre de un usuario Samba y no el login del sistema. Esta herramienta funciona igual que un cliente ftp, aunque puedes utilizar el explorador nautilus mediante la dirección `"smb://nombre_máquina/recurso"`; si no pones ningún recurso verás todos los disponibles.

Se puede montar al estilo de `"mount"` un directorio compartido en red (ya sea de Linux o de Windows) mediante la orden `"smbmount //nombre_máquina/directorio ruta_local -o nombre_usuario"`.

La conexión desde un sistema Windows se realiza como si fuera una máquina Windows más.

Ejercicios

- 1) Crea un grupo de trabajo llamado “grupo_red”, y a tu máquina asígnale el nombre miLinux.
- 2) Crea un usuario de Samba llamado “alumno”, con la misma contraseña.
- 3) Comparte el directorio /var/cache/apt/archives con el nombre “red_apt”.
- 4) Si tienes una impresora conectada a tu Linux compártela con el nombre “red_impres”

Soluciones

- 4) Tu archivo smbpasswd y smb.conf debe parecerse mucho a éste.

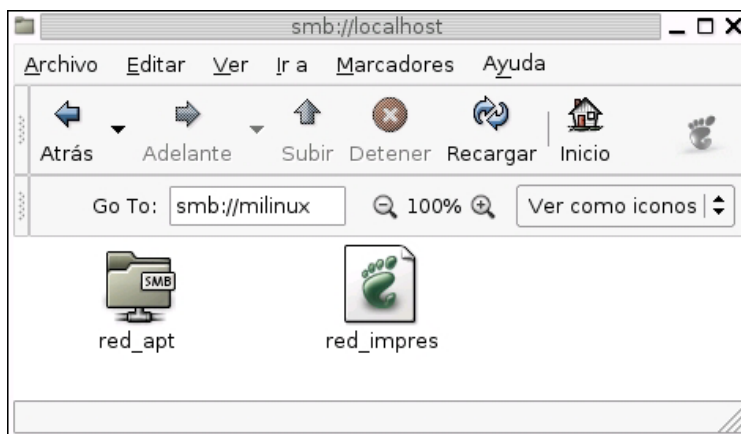
```
guadalinux:/etc/samba# cat smbpasswd
alumno:1001:XXXXXXXXXXXXXXXXXXXXXXXXXXXXX:XXXXXXXXXXXXXXXXXXXXXXXXXXXXX:[U] :LCT-00000000:
guadalinux:/etc/samba# cat smb.conf
# Samba config file created using SWAT
# from 127.0.0.1 (127.0.0.1)
# Date: 2004/06/04 19:18:21

# Global parameters
[global]
    workgroup = GRUPO_RED
    netbios name = MILINUX
    server string = Ordenador con GuadaLinux
    panic action = /usr/share/samba/panic-action %d

[red_apt]
    path = /var/cache/apt/archives
    valid users = alumno

[red_impres]
    printable = Yes
guadalinux:/etc/samba#
```

Desde nautilus deberías ver:



CAPÍTULO 14

Configuración de un Firewall

Una red nunca es segura; bajo esta premisa, si poseemos una red (o incluso si tenemos una única máquina conectada a Internet), debemos tomar todas las precauciones a nuestro alcance para impedir que los paquetes con código malicioso o no deseado circulen por ella.

Un cortafuegos o “firewall” es un dispositivo que filtra el tráfico entre dos o más redes; puede ser un dispositivo físico o un software que se ejecuta en un sistema operativo. Su funcionamiento se basa en una serie de reglas de filtrado, de manera que si se cumple una determinada regla se ejecute una determinada acción (por lo general aceptar el paquete o rechazarlo).

En Linux existe un cortafuegos incluido en todos los núcleos de la serie 2.4, denominado “netfilter”; la herramienta para manejarlo se denomina “iptables”, que es la que se utiliza para utilizar las reglas de filtrado. Todos los paquetes que pasen por nuestro Linux (ya sean creados internamente o provengan del exterior) serán comprobados por el cortafuegos (si está activado, evidentemente). Las acciones que puede tomar con los paquetes que pasan por él son DROP (descartar el paquete, con lo que éste desaparece) o ACCEPT (dejarle pasar y permitirle alcanzar el destino).

Hay que tener en cuenta que este filtrado de paquetes se realiza antes del control de acceso que ya hemos visto y para todos los paquetes de red (no sólo los servicios ejecutados a través de inetd con los archivos hosts.allow y hosts.deny), por lo que es una herramienta con una potencia y eficacia mayor al constituir la primera línea de defensa.

Otra utilidad del cortafuegos es que realiza operaciones de “ip_masquerading” o enmascaramiento IP, que es una técnica que permite la navegación a toda una red local con una única conexión a Internet sin necesidad de utilizar un servidor proxy; es una de las posibilidades que nos ofrece el sistema

NAT (Network Address Translation, o Traducción de Direcciones de Red.

Configurar las reglas de un cortafuegos correctamente no es una tarea fácil, ya que requiere de amplios y profundos conocimientos del funcionamiento de la familia de protocolos TCP/IP. Además estas reglas desaparecen al reiniciar la máquina, lo que obliga a tener que volver a introducirlas de nuevo; la solución para esto es escribir todas las reglas en un archivo y hacer que se ejecute en el arranque (a modo de servicio).

Afortunadamente tenemos una herramienta gráfica que hará nuestro trabajo más fácil; se trata de la “Configuración cortafuegos” en el menú Aplicaciones -> Herramientas del Sistema; esta herramienta se llama “lokkit” y se encarga de configurar las “iptables” sin tener que escribir ni una línea en la consola, tan sólo hay que responder una serie de preguntas. El programa está en castellano, así que es fácil realizar el proceso si tienes las ideas claras de cómo quieres configurar tu red o tu ordenador. Observa que en la primera pantalla te advierte que “No es un diseñador de cortafuegos para expertos”; esto no debe confundirte y hacerte creer que no es útil, sino que un experto preferirá introducir las reglas a mano; de todas formas verás que se pueden modificar las reglas configuradas por “lokkit” posteriormente.



La primera pregunta es la más difícil: hay que decidir si queremos una seguridad alta o baja (o no utilizar ninguna, pero ésta no es recomendable). La baja

es recomendable en la mayoría de los casos ya que bloqueará la mayoría de los servicios y te permitirá utilizar muchas aplicaciones de Internet, mientras que la alta bloquea todos los accesos (tanto entrantes como salientes) salvo los que le especifiquemos; ésta es recomendable si piensas utilizar tu máquina como servidor de cara a Internet (si éste es tu caso deberías plantearte introducir las reglas a mano).

Selecciona entonces la “Seguridad Baja”; lo primero que te pregunta es si las interfaces de red encontradas son fiables o no, es decir, si pertenecen a la red local (que se supone libre de problemas) o a una conexión de Internet. Sigue el consejo de la pantalla y para la conexión a Internet selecciona que no; para el resto depende de si quieres filtrarles el acceso a través de tu máquina o permitirles el paso.

En la siguiente debes responder si tu máquina configura su dirección IP de forma dinámica, para que puedas comunicarte con el servidor y obtenerla (porque si la bloqueas no podrás utilizar la interfaz); la mayoría (no todos) de los proveedores de Internet proporcionan IPs dinámicas mediante DHCP, así que para la conexión a Internet responde que sí.

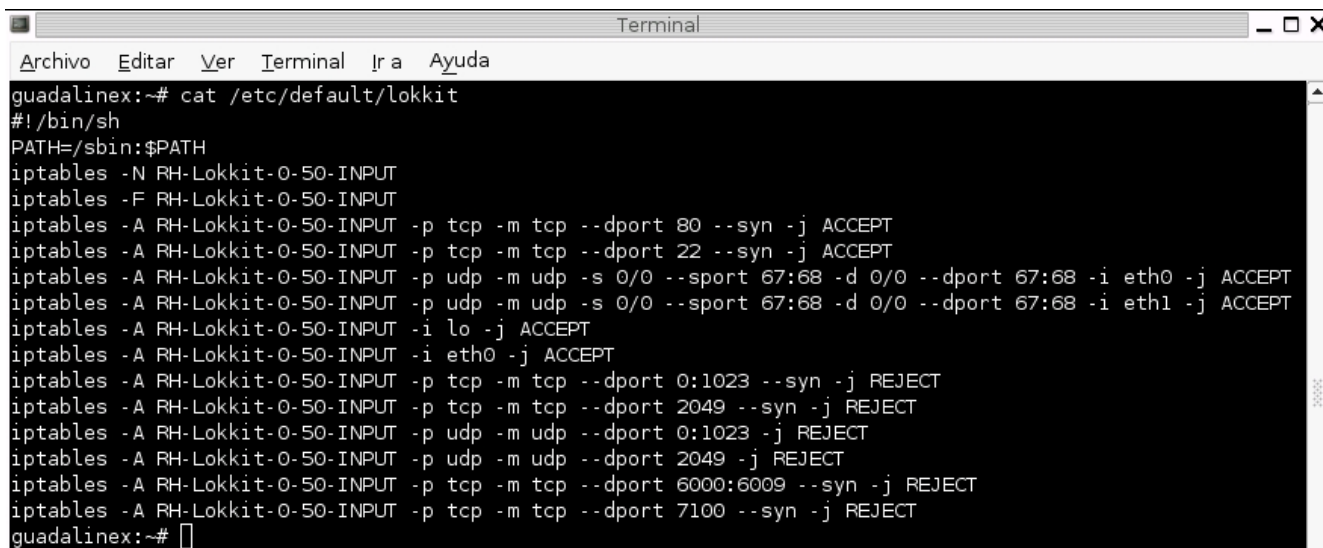
Ahora vas a tener que decidir si se podrá acceder a tu máquina desde el exterior, para seleccionar los servicios uno a uno. Ten en cuenta que hay servicios que puede ser buena idea permitirles acceso para usarlos tú mismo, así que lo más sensato es responder que sí.

Seguidamente “lokkit” va a reconocer los servicios de que dispones en tu máquina y te va a preguntar uno por uno si quieres permitir el acceso o no desde el exterior (desde tu propia red siempre podrás acceder a ellos). Por ejemplo puedes activar el servidor Web (si lo tienes configurado y listo para funcionar) y el ssh, pero no actives ni el SMTP ni el telnet.

La última pantalla advierte que debes tener cuidado, porque si activas el cortafuegos y estás accediendo de forma remota a tu máquina te puedes quedar sin acceso inmediato a ella; se supone que estás sentado delante del ordenador del que estás configurando el cortafuegos, así que pulsa “Terminar” para que se activen las reglas de filtrado y comenzar a utilizar el cortafuegos.

Otra de las ventajas de utilizar “lokkit”, a parte de lo fácil que ha sido hacerlo de forma gráfica, es que almacena las reglas en el archivo /etc/default/lokkit, para que estén disponibles en el caso de que el servicio

“lokkit” sea ejecutado al iniciar la máquina; para el caso que hemos realizado aquí, puedes ver en la imagen las órdenes necesarias para crear las reglas.

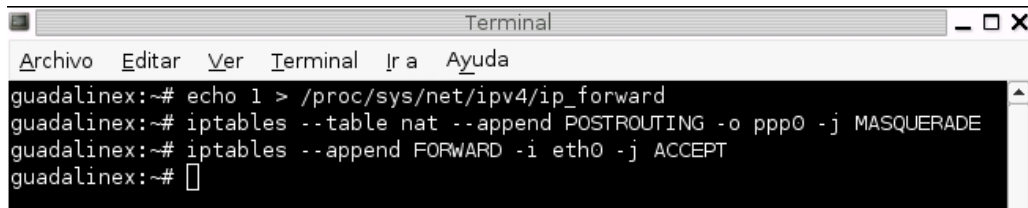


```
Terminal
Archivo Editar Ver Terminal Ir a Ayuda
guadalinux:~# cat /etc/default/lokkit
#!/bin/sh
PATH=/sbin:$PATH
iptables -N RH-Lokkit-0-50-INPUT
iptables -F RH-Lokkit-0-50-INPUT
iptables -A RH-Lokkit-0-50-INPUT -p tcp -m tcp --dport 80 --syn -j ACCEPT
iptables -A RH-Lokkit-0-50-INPUT -p tcp -m tcp --dport 22 --syn -j ACCEPT
iptables -A RH-Lokkit-0-50-INPUT -p udp -m udp -s 0/0 --sport 67:68 -d 0/0 --dport 67:68 -i eth0 -j ACCEPT
iptables -A RH-Lokkit-0-50-INPUT -p udp -m udp -s 0/0 --sport 67:68 -d 0/0 --dport 67:68 -i eth1 -j ACCEPT
iptables -A RH-Lokkit-0-50-INPUT -i lo -j ACCEPT
iptables -A RH-Lokkit-0-50-INPUT -i eth0 -j ACCEPT
iptables -A RH-Lokkit-0-50-INPUT -p tcp -m tcp --dport 0:1023 --syn -j REJECT
iptables -A RH-Lokkit-0-50-INPUT -p tcp -m tcp --dport 2049 --syn -j REJECT
iptables -A RH-Lokkit-0-50-INPUT -p udp -m udp --dport 0:1023 -j REJECT
iptables -A RH-Lokkit-0-50-INPUT -p udp -m udp --dport 2049 -j REJECT
iptables -A RH-Lokkit-0-50-INPUT -p tcp -m tcp --dport 6000:6009 --syn -j REJECT
iptables -A RH-Lokkit-0-50-INPUT -p tcp -m tcp --dport 7100 --syn -j REJECT
guadalinux:~#
```

Aquí puedes ver la complejidad de la que hablaba antes; además el orden en el que se introducen las reglas es importante, y de cambiarlas podría hacer que algún paquete se “colara” en nuestro sistema. Hay quien dice que la configuración de un cortafuegos es un “arte” reservado a unos pocos privilegiados; de cualquier forma, para la mayoría de la gente no suele ser necesaria una configuración tan exhaustiva.

Para activar el “ip-masquerading” desgraciadamente debemos hacerlo a mano, aunque se pueden añadir las reglas al archivo /etc/default/lokkit (ten en cuenta que si ejecutar la herramienta de configuración deberás introducirlas de nuevo). En este caso además de crear un par de reglas debemos habilitar el reenvío de paquetes, lo que se realiza escribiendo un “uno” en el archivo apropiado (que puedes ver en la imagen); si este archivo no existe, es necesario recompilar el núcleo y activar la opción “ip_forwarding”.

Las dos reglas que necesitamos sirven la primera para enmascarar todos los paquetes y los encamine hacia la interfaz ppp0, mientras que la segunda sirve para aceptar los paquetes que vengan de la interfaz eth0 para que puedan ser reenviarlos.

A terminal window titled "Terminal" with a menu bar containing "Archivo", "Editar", "Ver", "Terminal", "Ir a", and "Ayuda". The terminal shows the following commands and output:

```
guadalinux:~# echo 1 > /proc/sys/net/ipv4/ip_forward
guadalinux:~# iptables --table nat --append POSTROUTING -o ppp0 -j MASQUERADE
guadalinux:~# iptables --append FORWARD -i eth0 -j ACCEPT
guadalinux:~#
```

He supuesto que te conectas a Internet a través de un módem (ya sea módem convencional, ADSL o módem cable) que utiliza la interfaz ppp0 para acceder a Internet, y además tienes una tarjeta de red (eth0) para acceder a una red local. De esta forma todos los paquetes que lleguen a la máquina con este cortafuegos serán encaminados a través de Internet pero con la IP del cortafuegos, de manera que de cara al exterior no se tiene conocimiento de la red local; cuando los paquetes se reciben como respuesta a alguna petición, el sistema los reenvía a su destinatario.

Ejercicios

- 1) Configura el cortafuegos con un nivel seguridad alta
- 2) Saca una copia del archivo de configuración y llámalo /root/seg_alta
- 3) Configura el cortafuegos con un nivel de seguridad bajo
- 4) Saca una copia del archivo de configuración y llámalo /root/seg_baja
- 5) Observa las diferencias entre los dos archivos

Soluciones

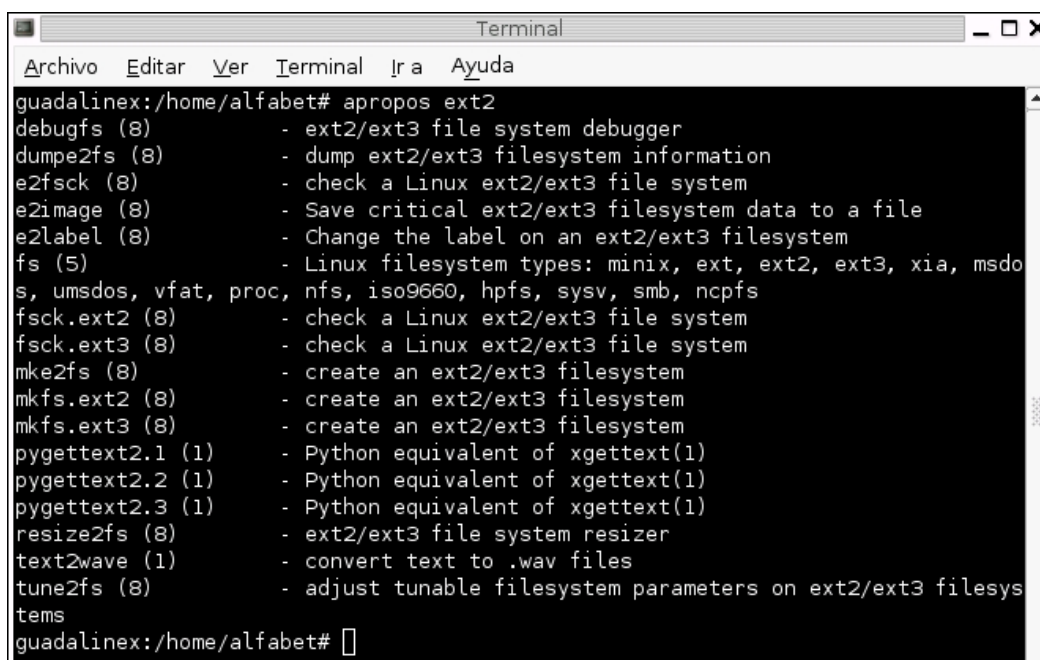
- 2) `cp /etc/default/lokkit /root/seg_alta`
- 4) `cp /etc/default/lokkit /root/seg_baja`
- 5) Podrás observar cómo ambas configuraciones parecen muy complejas; si has respondido del mismo modo en ambos casos verá como en el caso del nivel alto las reglas son más restrictivas que en el otro caso.

CAPÍTULO 15

Fuentes de Información sobre administración de Linux

La mayoría de la información se encuentra en el propio Linux, mediante las páginas de manual o páginas “man”, que debe ser siempre tu primera opción a la hora de resolver alguna cuestión. Para acceder a ellas hay que teclear “man” seguido del comando del que queramos información, por ejemplo “man ls”, que nos mostrará la información completa; si sólo estamos interesados en la descripción, podemos emplear el comando “whatis” en su lugar (whatis ls).

Otro comando útil es “apropos”, que busca en todas las página de manual por ocurrencias de la cadena escrita a continuación; por ejemplo con “apropos ext2” veremos los comandos en los que aparece de una u otra forma el sistema de ficheros ext2.



```

Terminal
Archivo  Editar  Ver  Terminal  Ir a  Ayuda
guadalinux:/home/alfabet# apropos ext2
debugfs (8)          - ext2/ext3 file system debugger
dumpe2fs (8)         - dump ext2/ext3 filesystem information
e2fsck (8)           - check a Linux ext2/ext3 file system
e2image (8)          - Save critical ext2/ext3 filesystem data to a file
e2label (8)          - Change the label on an ext2/ext3 filesystem
fs (5)               - Linux filesystem types: minix, ext, ext2, ext3, xia, msdos,
umsdos, vfat, proc, nfs, iso9660, hpfs, sysv, smb, ncpfs
fsck.ext2 (8)        - check a Linux ext2/ext3 file system
fsck.ext3 (8)        - check a Linux ext2/ext3 file system
mkfs (8)             - create an ext2/ext3 filesystem
mkfs.ext2 (8)        - create an ext2/ext3 filesystem
mkfs.ext3 (8)        - create an ext2/ext3 filesystem
pygettext2.1 (1)     - Python equivalent of xgettext(1)
pygettext2.2 (1)     - Python equivalent of xgettext(1)
pygettext2.3 (1)     - Python equivalent of xgettext(1)
resize2fs (8)        - ext2/ext3 file system resizer
text2wave (1)        - convert text to .wav files
tune2fs (8)          - adjust tunable filesystem parameters on ext2/ext3 filesystems
guadalinux:/home/alfabet#

```

Pero la principal fuente de información es la web; en ella puedes encontrar manuales para casi todo. Los siguientes enlaces no sólo tienen información, solución de problemas e incluso foros, sino que a menudo también tienen noticias sobre el desarrollo de algún programa, e incluso enlaces a la página de descargas o al código fuente.

www.guadalinex.org

Página oficial del proyecto GuadaLinux; desde ella puedes acceder a la zona de descargas, que dispone entre otras cosas de ayuda y manuales. También posee un foro donde exponer tus dudas o consultar la de otras personas.

<http://es.tldp.org>

El proyecto LuCAS (LinUx en CASTellano) pretende ofrecer documentación en español referente a Linux, encargándose de la traducción de documentos y manuales que en su origen están escritos en inglés.

<http://www.insflug.org>

Página en castellano del INSFLUG, que realiza las traducciones de documentos breves como las preguntas de uso frecuentes (FAQ) o los manuales “como” (How-To).

<http://www.gacetadelinux.com>

Edición en español de la revista más famosa sobre Linux; además de interesantes artículos tiene una sección de trucos.

<http://www.linuxlots.com/~barreiro/spanish/gnome-es>

Página del grupo dedicado a la traducción de los documentos relacionados entre otros con el proyecto Gnome.

<http://quark.fe.up.pt/ApachES/manual-es>

Manual oficial del servidor web Apache en su versión 1.3

<http://www2.idesoft.com/squid/index.html#top>

Información sobre el proxy Squid.

<http://www.proftpd.org>

Página oficial sobre el servidor proftpd; está en inglés.

<http://www.esdebian.org>

Aquí reside una comunidad española de usuarios de Linux.

<http://www.insflug.org/COMOs/Samba-Como/Samba-Como.html>

Manual en línea sobre la configuración de Samba.

A parte de estos, siempre puedes buscar algún manual o ayuda en la multitud de foros que existen sobre cualquier tema.

Glosario

Bash: ver shell

Cuenta: Pareja de palabras formada por un nombre de usuario y su contraseña

Daemon/Demonio: Programa residente en memoria y que está a la espera de que alguien necesite de sus servicios para realizar una acción determinada.

Debian: Una de las distribuciones más usadas de Linux; sirve como base a otras distribuciones.

Device: Dispositivo; representan entre otros a cualquier elemento de hardware; residen en el directorio /dev.

DHCP: Protocolo de Control de Huésped Dinámico; es un mecanismo para asignar las direcciones IP sobre la marcha, según se vayan conectado máquinas. Un servidor se encarga de gestionar la entrega de las IP.

DNS: Servidor de Nombres de Dominio; es un servicio que permite obtener la IP de una máquina a partir de un nombre (y viceversa).

Firewall: Cortafuegos; es un método para filtrar las conexiones entrantes y salientes de una red para protegerla de accesos no autorizados.

FTP: Protocolo de Transferencia de Ficheros; permite enviar y recibir archivos de forma directa a través de una red.

GID: Identificador de grupo; número único que identifica a un grupo en el sistema.

Grupo: Asociación de usuarios sobre los que se realiza la misma acción

administrativa.

Host: Huésped; nombre genérico que recibe cualquier máquina de una red.

HTTP: Protocolo de Transferencia de HiperTexto; protocolo utilizado principalmente en páginas web para enviar documentos de texto.

Ide: Modo de conexión interno para unidades como discos duros, cd-roms y grabadoras. En los ordenadores personales puede haber hasta cuatro de ellos, distribuidos en dos canales (primario y secundario).

IP: Protocolo entre redes. Es el protocolo básico para conectar dos redes de distintas. Se basa en la asignación de direcciones únicas a cada máquina.

Gnome: Entorno de modelos de objetos de red GNU; uno de los entornos de escritorio más utilizados; posee sus propias aplicaciones para las tareas más comunes.

KDE: Entorno de escritorio K; el otro entorno de escritorio más utilizado junto con Gnome; al igual que éste, dispone de muchas aplicaciones.

Localhost: Nombre que recibe la máquina propia o local; es una referencia a la dirección de red 127.0.0.1.

Login: Nombre que identifica a una persona ante el sistema.

Módulo: Código compilado que complementa al núcleo o kernel y que añade alguna funcionalidad que éste no presenta.

MTA: Agente de transporte de correo; es el encargado de enviar un mensaje a la dirección adecuada.

NFS: Sistema de ficheros en red; es un sistema para compartir un sistema de ficheros a través de la red.

Paquete: Archivo comprimido que se utiliza para distribuir programas para instalar en una distribución Linux.

Password: Contraseña; necesaria para que un usuario pueda acceder al sistema. Cada contraseña está asignada a un nombre de usuario.

Pop3: Protocolo de oficina de correos versión 3. Es el encargado de coger el correo de un usuario de un servidor de correo y almacenarlo en su máquina.

Proxy: Representante; es el encargado de sustituir (mediante su IP) a una o más máquinas y hacerse pasar por ellas.

Puerto: Zona de acceso a una máquina que permite diferenciar información destinada para varias aplicaciones.

Redhat: Una de las distribuciones más utilizadas, controlada por la empresa del mismo nombre.

Repositorio: Lugar de almacenamiento de paquetes para que los usuarios puedan instalar nuevos programas y actualizar sus sistemas con versiones más nuevas.

Root: Administrador del sistema; es un usuario con control total sobre el sistema, por lo que es conveniente utilizarlo únicamente cuando es estrictamente necesario.

Router: Dispositivo hardware o software que realiza la interconexión entre dos redes diferentes.

Script: Archivo de texto que guarda órdenes para ser ejecutadas una tras otra.

Samba: Implementación en Linux del protocolo SMB.

SMB: Bloque de mensajes de servidor. Protocolo utilizado en redes para compartir archivos e impresoras entre máquinas con diferentes sistemas operativos.

Shell: Intérprete de comandos, que permite introducir órdenes o comandos.

SMTP: Protocolo encargado del envío de mensajes de correo entre servidores.

Swap: Memoria de intercambio, utilizado como expansión de la memoria principal; puede residir en una partición independiente o puede ser un archivo convencional en un sistema de ficheros.

TCP: Protocolo de control de transporte; complementa a IP y se encarga de asegurar que la conexión se realiza correctamente de extremo a extremo.

UDP: Protocolo de datagrama de usuario; protocolo similar a TCP pero más ligero.

No asegura que los datos lleguen correctamente e incluso tolera alguno se pierda.

UID: Identificador de Usuario; número único que identifica a un usuario en el sistema.

Usuario: Nombre de una persona física (o programa) para identificarse en el sistema.

X: ver X-Window.

XFree86: ver X-Window.

X-Window: Sistema cliente/servidor que permite comunicar un hardware con una aplicación, y que define los métodos para redibujar la pantalla, mover ventanas, recoger pulsaciones de ratón, ...