

# Curso práctico del Linux

por: Claus Denk

---

Diciembre 1995

Universidad de Sevilla - Facultad de Física

## Índice General

<b>1</b>	<b>Introducción</b>	<b>6</b>
	¿ Qué podemos hacer con el Linux ? . . . . .	6
	Ventajas del Linux frente a otros sistemas operativos . . . . .	7
	¿ Y las inconveniencias ? . . . . .	7
	Pequeña historia del Linux . . . . .	8
	Asuntos legales . . . . .	8
	¿ Qué máquina hace falta para el Linux ? . . . . .	9
	Compatibilidad del hardware . . . . .	10
	Información sobre el Linux . . . . .	12
	Por la red: . . . . .	12
	Libros: . . . . .	12
	Obtención del Linux . . . . .	13
<b>2</b>	<b>Instalación del Linux</b>	<b>14</b>
	Organización del SLACKWARE 3.0 . . . . .	15
	Crear los disquetes de arranque . . . . .	17
	Creación del bootdisk . . . . .	18
	Creación del rootdisk . . . . .	19
	Preparar espacio para el Linux (DOS) . . . . .	20
	Uso del FDISK (DOS 6.2): . . . . .	21
	Uso del programa FIPS . . . . .	22
	Instalación del Linux . . . . .	23
	El fdisk del Linux . . . . .	24
	HELP y QUICK/VERBOSE . . . . .	25
	KEYMAP . . . . .	25
	SWAP . . . . .	25

<i>ÍNDICE GENERAL</i>	3
TARGET . . . . .	26
SOURCE . . . . .	26
DISK SETS . . . . .	27
INSTALL . . . . .	27
CONFIGURE YOUR SYSTEM . . . . .	28
<b>3 Usar el Linux</b>	<b>30</b>
Shells . . . . .	30
Procesos . . . . .	31
Ficheros en UNIX . . . . .	32
Ficheros especiales . . . . .	33
Consolas virtuales . . . . .	33
Redirección y Pipes . . . . .	34
Comandos del UNIX/Linux . . . . .	35
Trabajar con ficheros y directorios . . . . .	36
El editor <b>vi</b> . . . . .	39
Compresión y Archivador . . . . .	40
Control de procesos . . . . .	41
Información y entorno . . . . .	42
Comandos útiles en shell-scripts . . . . .	43
Trabajar con disquetes del MS-DOS . . . . .	44
Comandos para el <b>root</b> . . . . .	45
Comandos avanzados . . . . .	46
<b>4 Configuración del sistema</b>	<b>47</b>
¿ Dónde están los ficheros ? . . . . .	48
Configuración de la red . . . . .	50
Terminología . . . . .	51

La utilidad netconfig . . . . .	52
Instalación del LOADLIN . . . . .	53
Las utilidades para disquetes en formato MS-DOS . . . . .	55
La Configuración del X-Windows . . . . .	56
Elegir el servidor de X . . . . .	57
El fichero XF86Config . . . . .	58
Probar la configuración . . . . .	62
El programa xf86config . . . . .	63



Linus Torvalds después de terminar la versión 1.0 del “kernel”

**Advertencia:**

En este curso se explicarán métodos para cambiar la configuración de un ordenador, que pueden provocar pérdidas de datos. No asumo ninguna responsabilidad por los daños sufridos como consecuencia de los posibles errores contenidos en este documento.

C. Denk

## 1 Introducción

El Linux es un sistema operativo para ordenadores con procesadores de la familia de Intel (386/486/Pentium/..), compatible con UNIX. Linux es de libre distribución (free software) y ya está disponible una gran variedad de aplicaciones (programas) para Linux. En este curso no vamos a distinguir entre el sistema operativo y las aplicaciones, así que:

$$\text{Linux} = \text{sistema operativo} + \text{aplicaciones}$$

Antes de entrar en el tema vamos a ver qué se puede hacer con el Linux y si merece la pena invertir algo de tiempo (dinero no hace falta) en instalarlo y aprenderlo. He intentado orientar el contenido de este curso a las necesidades de un investigador. Esto se refleja en los temas elegidos. Para una empresa o un programador serían más interesantes otros aspectos (bases de datos, entornos de programación) del Linux. De estos temas vamos a tratar sólo lo imprescindible.

### ¿ Qué podemos hacer con el Linux ?

- Trabajar en un entorno gráfico o de texto.
- Escribir (problemas, artículos, tesis, etc.): Latex ,Wordperfect(\$).
- Dibujar (diagramas) y pintar (obras de arte): Xfig y Xpaint.
- Programar: FORTRAN, C, C++, Lisp, Pascal, ....
- Cálculo simbólico: MuPad, Mathematica(\$), Maple(\$).
- Analizar datos gráficamente: gnuplot, plotmtv, xplot.
- Acceso al Internet: mail, Mosaic (WWW), tin, ftp, telnet y mucho más.
- Bases de datos, .....

Notas:

## Ventajas del Linux frente a otros sistemas operativos

- Muy barato: 0 pesetas.
- Hardware barato y ampliable.
- Compatibilidad (X - UNIX - C - Internet).
- Multitasking - multiuser - acceso remoto vía X.

## ¿ Y las inconveniencias ?

- Support.
- Estabilidad.
- No es tan sencillo como MS-Windows o Mac.

## Pequeña historia del Linux

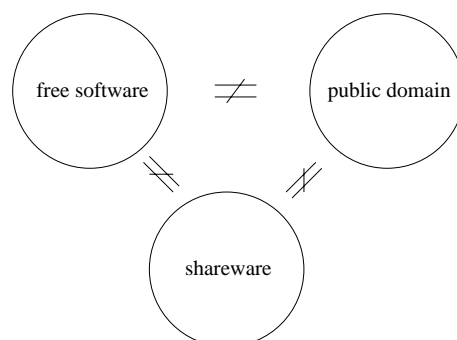
1969	AT&T UNIX, Ken Thompson
1978	BSD (= Berkley Software Distribution) UNIX
1991	Agosto: Versión 0.01 del Linux (Linus Torvalds)
1993/1994	Versión 1.0 del Linux (sin X)
hoy	Versión 1.2.13 con X-Windows y muchas aplicaciones, GNU

Notas:

## Asuntos legales

El Linux es de libre distribución (free software). La GNU (Gnu's Not Unix) General Public License (GPL) define el concepto de free software. Brevemente:

- Es legal copiar, cambiar y redistribuir todos los ficheros de Linux . Las versiones copiadas o modificadas de Linux también deben ser protegidas por la GPL.
- Se puede vender el Linux (para beneficiarse) pero no se puede restringir el derecho a redistribución.



Notas:



## ¿ Qué máquina hace falta para el Linux ?

A la hora de comprar hardware para Linux hay que analizar para qué se va a usar el ordenador. En algunos casos se podrá usar una máquina ya existente si se amplía un poco. En la siguiente tabla he resumido tres sistemas como ejemplo, pero hay que tener en cuenta que esta colección es muy subjetiva y puede cambiar rápidamente con el desarrollo en hard- y software.

sistema	mínimo	estándar	lujo
uso	?	Terminal X, Latex, Internet	multi-usuario, ...
CPU	386 SX/DX	486 DX66 o Pentium 90	Pentium 120/133/..
memoria RAM	4 MB	8/16 MB	32-.. MB Ram
disco duro	80 MB	540 MB	$\geq 1$ Gb
monitor	14 "	17"	17"/21" o sin

En un grupo de investigación hay que tener en cuenta las necesidades de todos los miembros del grupo para poder configurar una red de estaciones de trabajo óptima. Con el presupuesto, como condición de contorno, se debe intentar sacar un alto rendimiento de cada componente de la red. Algunas reglas generales para la configuración de una red son:

- Si se va a realizar mucho trabajo numérico, hay que considerar un ordenador que esté solamente dedicado a esta tarea. Un sistema “lujo” o bien una estación de trabajo más potente (Dec-Alpha, HP, etc.) puede ser usado por todos los miembros del grupo.
- En muchos casos sólo hace falta una impresora que esté conectada a un ordenador de la red. Esto también se puede aplicar a otros tipos de periféricos caros (como scanner, etc.)
- Un servidor de ficheros (sistema “estándar” con un disco duro grande) facilita cambiar de máquina a máquina. ¡ Una unidad de cinta (para copias de seguridad) y un power-supply (riesgo de apagón) aseguran su trabajo !
- Si las estaciones de trabajo que sirven como terminal son capaces de realizar un trabajo local que requiere mucho RAM (Latex, Mathematica, etc.), estos procesos no se tienen que correr en el servidor numérico.

Notas:

## Compatibilidad del hardware

El mercado de los ordenadores PC basados en procesadores Intel es muy amplio. La arquitectura abierta ha hecho posible un gran variedad de periféricos a bajo precio (discos duros, tarjetas gráficas, monitores, tarjetas de comunicación, CD-ROM, etc.) para estos sistemas. El Linux trabaja bien con la gran mayoría de estos componentes, pero hay periféricos (y ordenadores) que no son compatibles con el Linux. A la hora de comprar un equipo nuevo, hay que asegurarse de que todos los componentes sean compatibles con el Linux. No es suficiente si el vendedor le dice que su hardware es compatible con todos los sistemas operativos (muchas veces se refieren con esto a MS-DOS y Windows).

**Pida un presupuesto detallado y compruebe todos los componentes en el**

**→ `HARDWARE-HOWTO`**

He resumido un `→ SHORT_HARDWARE_HOWTO` que contiene los capítulos más importantes del `HARDWARE_HOWTO`. Para los periféricos especiales, como unidades de cinta, tarjetas de sonido u ordenadores portátiles hay que consultar el documento entero.

Algunos consejos generales para la compra:

- Una CPU con alta velocidad no significa necesariamente un ordenador rápido. Todos los componentes deben ser óptimos para conseguir un alto rendimiento.
- ¡ No ahorre en el monitor !
- Normalmente no hace falta una tarjeta gráfica demasiado avanzada (4MB VRAM ..), pero sí una tarjeta que tenga un “accelerated chipset”.
- El ordenador debe ser ampliable.

Notas:

Modos de Video:

Modo	Monitor	Resolución	Freq. horiz.(kHz)	Freq. vert.(Hz)
VGA	14"	640 x 480	31.5	60
VESA/72	14"/15"	800 x 600	48.1	72
VESA/70	17"	1024 x 768	56.5	70
?	21"	1280 x 1024	78.85	74

El modo VESA/56 (800 x 600 con 56 Hz) no es aconsejable, ya que provoca cansancio en los ojos. Podemos averiguar si somos sensibles al efecto parpadeo del monitor, mirando de reojo a la pantalla (preferentemente con fondo blanco).

Notas:

## Información sobre el Linux

### Por la red:

- WorldWideWeb (WWW). Textos en forma de “hypertext”.  
<http://www.cs.us.es/archive/linux.html> - The Linux Documentation Project
- mail to `linux@cs.us.es`, ayuda de gente que usa el Linux aquí en Sevilla.
- newsgroups on tin: `comp.os.linux.xxxxx` - Aquí se pueden hacer preguntas y normalmente se reciben muchas respuestas rápidamente.

### Libros:

- Linux Installation and Getting Started by Matt Welsh. *Libro accesible por la red en forma .html o en postscript.*
- Running Linux by Matt Welsh and Lar Kaufman, O'Reilly & Associates, Inc., 1995, ISBN 1-56592-100-3. *cerca de 600 páginas, de fácil lectura, 25 \$.*
- DrX. Linux, The Linux Documentation Project, 1994, ISBN 1-885329-02-4. *Colección de toda la información del Linux Documentation Project (todo accesible por la red), más de 1000 páginas. Incluye Linux Installation and Getting Started y temas más avanzados, 50 \$.*
- Using Linux, QUE, 1995, ISBN 0-7897.0100-6. *829 páginas, contiene una referencia muy completa sobre el Linux, incluye CD-ROM con Linux 50 \$.*
- X Window System Users's Guide by Valerie Quercia and Tim O'Reilly, O'Reilly & Associates, Inc., 1988..1993, ISBN 1-56592-014-7. *Todo sobre el X-Windows a nivel de usuario, 835 páginas, 35 \$.*

### Notas:

## Obtención del Linux

- CD-ROM de varios distribuidores. Existen varios tipos de distribuciones (SLACKWARE, YGGDRASIL, ..). Éstas se diferencian en el método de instalación, y también en la colección de programas. La distribución SLACKWARE es la más común (?) y es muy completa. Puede ser un poco difícil que el Linux detecte el CD-ROM, pero una vez configurado la instalación es muy cómoda.
- Disquetes de alguien que ya tenga el Linux. Es la manera más fácil, pero también conlleva todos los inconvenientes de los disquetes.
- `ftp` de un servidor a la partición del DOS del disco duro y, posteriormente, instalación desde estos ficheros. Si no hay ni CD-ROM ni disquetes, esto es una manera sencilla y segura de instalar el Linux. Hace falta bastante sitio en la partición del DOS y el ordenador ya debe estar conectado a la red. Para evitar complicaciones hay que preparar la partición del Linux antes de traerse los ficheros (siguiente capítulo)
- Instalación por la red. En este caso el CD-ROM está conectado a un ordenador remoto y hay que “montar” el directorio que contiene los ficheros. Para entendidos en el tema.

Notas:

## 2 Instalación del Linux

La instalación del Linux no es tan sencilla como, por ejemplo, la instalación de Windows 3.1. Hay que plantearse cada paso de la instalación y debemos enterarnos de lo que hacemos. Los pasos que hay que seguir dependen de la configuración de la máquina en la que queremos instalar el Linux. En una máquina (incluso en un disco duro) pueden coexistir varios sistemas operativos, y no podemos tratar todas las posibilidades. Vamos a suponer que la máquina en la que se va a instalar el Linux tiene un disco duro de bastante capacidad, digamos 540 MB, y que el MS-DOS está instalado en esa máquina. Además, supongamos que el ordenador dispone de 8 MB de RAM o más. Los pasos que hay que dar son:

- Crear los disquetes de instalación.
- Liberar espacio del DOS en el disco duro para poder meter el Linux (FDISK del DOS).
- Definir el espacio liberado como utilizable por el Linux (`fdisk` del Linux).
- Instalar el Linux.
- Configurar el sistema.

Podemos seguir las instrucciones del

→ **INSTALLATION-HOWTO**

paso a paso.

Notas:

## Organización del SLACKWARE 3.0

Los directorios de la distribución de SLACKWARE tienen la forma siguiente:

BOOTING.TXT	COPYING	ChangeLog	Copyright
FAQ.TXT	INSTALL.TXT	LOWMEM.TXT	MIRRORS.TXT
README30.TXT	UPGRADE.TXT	bootdsk.12/	bootdsk.144/
contents/	contrib/	docs/	install/
kernels/	non-commercial-use/	patches/	rootdsk/
slaktest/	slakware/	source/	

Notas:

En el directorio **slakware/** encontramos varios subdirectorios que contienen los “disk sets”:

A1-A5	Sistema básico
AP1-AP5	Aplicaciones (sin X)
D1-D10	Varios compiladores (gcc)
E1-E6	Emacs editor
F1-F2	FAQ(=Frequently Asked Questions)
K1-K5	Código fuente del kernel de Linux
N1-N4	Networking (programas para la red)
Q1-Q15	Kernels precompilados para configuraciones especiales
T1-T9	TeX (LateX, BibTeX)
TCL1	Desarrollo de aplicaciones X
X1-X16	Sistema de X-Windows
XAP1-XAP4	Aplicaciones para X
XD1-XD3	Desarrollo de servidores X
XV1-XV3	XView, manager de ventanas
Y1-Y4	Juegos

Notas:



## Crear los disquetes de arranque

Hace falta:

- Algunos disquetes (4-6) formateados para el DOS, distribución del Linux (disquetes, CD-ROM o conexión con la red)
- “Installation HOWTO”, FAQ.TXT
- Programas GZIP.EXE, RAWRITE.EXE y, si queremos usar el programa FIPS, los ficheros FIPS.EXE, RESTORRB.EXE, FIPS.DOC (todo en el directorio `install/`).
- Uno o dos “diskimages” del directorio `bootdsk.144/`, como, por ejemplo, `bare,idecd` o `scsi`. (En algunas distribuciones en forma `.gz`)
- “Diskimages” del directorio `rootdisk/`, por ejemplo, `color.gz` y `rescue.gz`

Aunque tengamos un CD-ROM, tenemos que preparar dos disquetes para poder instalar el Linux:

- El “bootdisk”. Con este disquete vamos a poder botar el ordenador con el “kernel” del Linux.
- El “rootdisk”. Tenemos que insertar este disquete después de botar con el “bootdisk”. El “rootdisk” contiene un sistema de ficheros que nos permite correr los programas que necesitamos para la instalación.

Copiamos estos ficheros al disco duro e imprimimos los documentos necesarios.

Notas:

### Creación del bootdisk

El “bootdisk” se crea a partir de un fichero del tipo “diskimage” (imagen del disco). En el directorio `bootsdsk.144` encontramos varios ficheros para crear un “bootdisk”. El “bootdisk” contiene el “kernel” del Linux, y debemos elegirlo de acuerdo con el hardware que tengamos. Por ejemplo:

<code>bare</code>	Disco duro IDE, no CD-ROM
<code>scsi</code>	Disco duro IDE o SCSI, CD-ROM SCSI
<code>idecd</code>	Disco duro IDE o SCSI, CD-ROM IDE/ATAPI

Para otras configuraciones podemos encontrar el “bootdisk” adecuado en el directorio `bootsdsk.144`. Si el fichero del cual queremos crear el “bootdisk” está comprimido (termina en `.GZ`) tenemos que descomprimirlo:

```
GZIP -D BARE.GZ
```

En la distribución SLACKWARE 3.0 no hay que descomprimir estos ficheros. Ahora usamos el programa RAWRITE para pasar el fichero al disquete.

```
RAWRITE
```

RAWRITE pregunta por el nombre del fichero y por la unidad de la disquetera. Al pasar el fichero al disquete, el resultado es un disquete no reconocible por el MS-DOS.

Notas:

### Creación del rootdisk

El “rootdisk” se crea de la misma manera que el “bootdisk”. En el directorio **rootdsk**s encontramos los ficheros para crear un “rootdisk”. El “rootdisk” contiene un sistema de ficheros (filesystem) básico. Aconsejo crear estos dos “rootdisks”:

color.gz     Instalación estándar.

rescue.gz    Disco de rescate, por si algo va mal.

En SLACKWARE 3.0 no hay que descomprimir estos ficheros, simplemente hay que pasarlos a un disquete, usando el programa RAWRITE.

Si el ordenador dispone solamente de 4 MB RAM hay que seguir un método diferente. En el fichero LOWMEM.TXT encontramos información sobre este caso.

Notas:

## Preparar espacio para el Linux (DOS)

En un PC, el disco duro normalmente está preparado solamente para el MS-DOS (todo el espacio está dedicado al MS-DOS). En el disco duro está guardada la “partition table” (tabla de partición). En esa tabla figura la siguiente información: qué parte del disco duro está dedicada al MS-DOS (y los demás sistemas operativos instalados) y con qué disco lógico (letra C: o D:) se relaciona. Podemos:

- Editar esta tabla con FDISK, provocando una pérdida de todos los datos en el disco duro.
- Cambiar esta tabla, con un programa como FIPS, de manera “suave”

---

En cualquier caso, hay que hacer un **BACKUP** de todos los ficheros importantes (o del disco duro entero) antes de usar un programa como **FDISK** o **FIPS**.

Antes y después de realizar el cambio en la tabla de partición hay que correr el **SCANDISK** para asegurarse de que todo está bien.

Hay que preparar un disco de utilidades. Para copiar los ficheros de arranque formatearemos el disquete con el comando “**FORMAT A: /S**”. Después, copiaremos en ese disquete programas como **FDISK**, **FORMAT**, **EDIT**, **CHKDSK**, **SCANDISK**, **MSAV**, **BACKUP**, **RESTORE**, etc.

¡ Hay que probar si el ordenador bota con ese disco !

---

**Uso del FDISK (DOS 6.2):**

En caso de que el disco duro tenga una única partición, para reducir su tamaño, podemos realizar los siguientes pasos (consulte el manual del DOS antes de usar el FDISK, puesto que se van a perder todos los ficheros).

- FDISK /STATUS
- FDISK
- 4. Mostrar información sobre la partición
- 3. Eliminar partición o unidad lógica
- 1. Eliminar partición primaria
- Qué partición desea eliminar ? -> [n] (véase en tabla, 1 en la mayoría de los casos)
- Escriba la etiqueta del volumen -> ..... (véase en tabla)
- Está seguro ? [S/N] -> Si respondemos con S borramos todos los ficheros.
- 4. Mostrar información sobre la partición
- 1. Crear partición DOS
- 1. Crear partición primaria DOS
- ¿ Desea utilizar tamaño máximo disponible y ..... (S/N) -> N
- Tamaño ? -> [nn] (por ejemplo, 80)
- 2. Establecer partición activa
- Escriba el número de partición que desee activar .. [n] (será 1)
- ESC
- El sistema se reiniciará (tener el disquete de utilidades DOS en la ranura del floppy)
- A:FORMAT C: /S
- Continuar ? [S]

Ahora podemos teclear una etiqueta y el ordenador debería botar de nuevo del disco duro. Podemos instalar de nuevo el DOS y hacer un RESTORE de nuestros ficheros.

Notas:

### Uso del programa FIPS

FIPS permite cambiar la tabla de partición sin que se produzcan pérdidas de datos (si tenemos suerte). Antes de usar este programa, estudie detalladamente la documentación del FIPS. Hay que preparar un disquete de utilidades con los ficheros de sistema y con los programas FIPS y RESTORRB.

Si el MS-Windows está instalado en la máquina, desactivaremos el fichero del “swap” (memoria virtual). Esto se consigue en el “Panel de Control”, después “386 extendido” y “Memoria virtual”. Con “Cambiar >>” podemos elegir que el tipo sea “Ninguno” y con “Aceptar” realizamos los cambios.

Ahora botaremos con el disquete de utilidades y cambiaremos al disco duro. Después, seguiremos los siguientes pasos:

- SCANDISK
- DEFRAG
- SCANDISK
- Botar del disquete de utilidades, no cambiar al disco duro.
- A:FIPS
- Botar de nuevo del disquete
- A:CHKDSK C: o mejor A:SCANDISK C:
- Botar del disco duro si no hay errores.

Si todo va bien, podemos reducir el tamaño de la partición de DOS sin perder los ficheros del DOS. Si SCANDISK detecta errores, tenemos que decidir si queremos intentar a arreglarlos. Si arreglamos algo, siempre usaremos la utilidad de SCANDISK para poder deshacer los cambios. En cualquier caso podemos usar RESTORRB para deshacer los cambios que ha realizado el FIPS.

Notas:

## Instalación del Linux

Ahora estamos preparados para instalar el Linux. Insertaremos el “bootdisk” y botaremos el ordenador desde ese disquete. En ese momento, el “kernel” del Linux se cargará. Leeremos detenidamente todos los mensajes. Cuando aparezca el

```
boot:
```

sólo será necesario introducir parámetros en caso de que la configuración sea especial (disco SCSI, etc.). Normalmente basta con pulsar INTRO. Cuando aparezca

```
VFS: Insert ramdisk floppy and press ENTER
```

insertaremos el “rootdisk” y pulsaremos INTRO. Si todo va bien, llegamos al

```
slackware login:
```

Nos identificamos como **root** (sin password) y ya nos encontramos en un entorno Linux, aunque todavía no se puede hacer mucho con el sistema. El primer paso consiste en preparar las particiones del Linux en el disco duro con el comando **fdisk**.

Notas:

## El fdisk del Linux

Con el comando

```
fdisk /dev/hda
```

entramos en el programa **fdisk** del Linux. **fdisk** está descrito muy detalladamente en el “Installation-HOWTO”. Por ejemplo:

- p - print the partition table (aquí aparece la partición del DOS).
- n - add a new partition.
- p - primary.
- 2 (número de la partición, 1 será del DOS).
- First cylinder: siguiente cilindro después del DOS.
- Last cylinder or + size: por ejemplo “+300M” o número de un cilindro (+ = Shift ;).
- p - print the partition table

Hay que dejar bastante sitio para el “swap”, que será un espacio en el disco duro dedicado solamente a la memoria virtual. Normalmente se usa una partición del “swap” de un tamaño equivalente al doble de la memoria RAM instalada en la máquina (8 MB RAM → 16 MB swap). La partición del “swap” se crea de la misma manera que la del Linux, pero hay que usar la función “(t)ype” del **fdisk** para cambiar el tipo de la partición después de crearla. El tipo “Linux swap” viene representado por el código “82”.

Si estamos seguros de nuestros cambios, usaremos “(w)rite” para escribir la tabla de partición en el disco duro. Para estar seguro de que el sistema ha reconocido las nuevas particiones, rebootaremos la máquina (**reboot** o Ctrl-Alt-Del) con el “bootdisk” y miraremos de nuevo la tabla de partición.

Notas:



Ahora podemos entrar en el programa de instalación del Linux:

```
setup
```

Este programa maneja los pasos más importantes de la instalación.

## **HELP y QUICK/VERBOSE**

Debemos acostumbrarnos a utilizar el menú y leer las notas del “HELP”. Podemos elegir los modos “QUICK” o “VERBOSE” (el último es mejor para los principiantes). Si entramos en la sección “KEYMAP”, el `setup` nos conduce paso a paso por las secciones siguientes.

## **KEYMAP**

Aquí podemos elegir un fichero para definir el teclado. El fichero `es.map.gz` se elige para teclados españoles.

## **SWAP**

Aquí activamos la partición del swap. El programa `setup` ejecuta los programas `mkswap` y `swapon`. El último comando tarda un poco.

## TARGET

En este menú elegimos la “partición de destino” para el Linux. A continuación, hay que formatear esta partición. ¡ Formatear con “check” tarda bastante tiempo ! Una buena elección para el número de bytes por “inode” es 2048 (un “inode” por cada 2048 bytes).

El programa `setup` detectará la partición del DOS y nos ofrece la posibilidad de hacer esta partición visible para el Linux. Podemos “montar” la partición

```
/dev/hda1
```

en el directorio

```
/dos
```

Después de haber añadido el directorio del DOS, es posible añadir otra partición del DOS. Con “q” llegamos a la siguiente sección.

## SOURCE

Aquí podemos elegir como (de donde) vamos a instalar los ficheros del Linux:

- 1) Install from hard drive partition
- 2) Install from floppy disks
- 3) Install via NFS
- 4) Install from a pre-mounted directory
- 5) Install from CD-ROM

En este curso los vamos a instalar de disquetes y, por lo tanto, elegiremos 2. Hay que especificar la disquetera, el disco A: con 1,44 MB equivale al “device” `/dev/fd0H1440`.

## **DISK SETS**

En esta sección elijiremos los “disk sets” que queremos instalar. Podemos instalar todos los ficheros de una vez, o bien, instalar sólo el “disk set A” .

## **INSTALL**

En el modo “QUICK” cada “disk set” ofrece un menú, de donde podemos elegir los ficheros que queremos instalar. En el modo “VERBOSE” el programa `setup` explica el contenido de cada fichero, pudiéndose elegir si lo queremos instalar o no. Conviene seleccionar el “Prompting Mode” como “Normal”.

## CONFIGURE YOUR SYSTEM

Aquí `setup` nos ayuda a configurar el sistema. Realizaremos los siguientes pasos:

- Crear un “bootdisk”. Podemos formatear un disquete y, después, usarlo para crear un disquete que va a botar nuestro sistema. Elegimos, por ejemplo, la opción “`make simple vmlinuz > /dev/fd0 bootdisk`”.
- Podemos configurar un modem (conectar el “device” con un puerto de comunicación, por ejemplo COM2)
- Tenemos que elegir el tipo de ratón que vamos a usar, y conectarlo con el puerto de comunicación, por ejemplo COM1.
- Hay que especificar el tipo de CD-ROM, si es que lo tenemos.
- Podemos elegir un “font” para la pantalla.
- LILO INSTALLATION: Esta sección nos ofrece la posibilidad de instalar un “bootloader”. Éste es un programa que se ejecuta antes de que ningún sistema operativo que se encuentre en el disco duro sea activo. Para los principiantes aconsejo la opción

`Skip LILO installation and exit this menu`

- Instalar **gpm**, que es un programa muy útil para seleccionar y copiar texto en la pantalla.
- Como último paso, hay que elegir la “timezone”, aquí en Sevilla “MET” (Middle European Time)

Salimos del **setup** con “EXIT” y podemos botar nuestro sistema con el “boot-disk” que hemos creado en el primer paso de **Configure your System**. ¡ Este disquete es diferente del “bootdisk” de la instalación ! Si el sistema está bien configurado, la combinación de teclas Ctrl-Alt-Del (Control-Alt-Supr) inicia el proceso de rebotar. Si todo va bien, aparecerá

**darkstar login:**

y ya tendremos un sistema Linux ! Todavía nos queda trabajo por hacer:

- El primer paso debería ser entrar como **root** (todavía no está puesto ningún password). Crearemos un password (comando **passwd**) para **root**.
- Debemos abrir una cuenta para un usuario con el comando **adduser**. Esta cuenta nos va a servir para “explorar” el sistema. Realizaremos un login de este usuario en el terminal virtual 2 (Alt-F2) para comprobar si el sistema acepta el login.

**Usaremos la cuenta root sólo para trabajos que un usuario normal no puede realizar !**

En el siguiente capítulo veremos las diferentes configuraciones que tenemos que hacer hasta que el sistema “pite” ...

## 3 Usar el Linux

Para configurar el Linux hace falta entender los conceptos básicos (y algunos avanzados) del UNIX. No podemos alcanzar un entendimiento profundo del UNIX en pocas horas, pero al configurar nuestra máquina aprenderemos mucho sobre el funcionamiento de este sistema operativo.

En lo siguiente vamos a presentar una pequeña introducción a los conceptos más importantes de UNIX. No creo que sea posible entender claramente todo lo que vamos a ver en este capítulo, pero al menos espero que nos acordemos de lo aquí expuesto en caso de necesidad.

### Shells

Después de realizar el login, el sistema arranca un programa (“shell”) que interpreta nuestros comandos y los transmite al “kernel” del Linux. Hay varios tipos de “shells”, por ejemplo

<b>csh</b>	C-shell, con sintaxis parecida al C
<b>bash</b>	Bourne again shell, shell por defecto del Linux
<b>zsh</b>	Z-shell, permite caracteres como ñ en nombres de ficheros

También existen otros “shells”, como el Bourne shell, el Korn shell, el T-shell y el A-shell. En este curso vamos a usar solamente el **bash**. Antes de presentarnos el “prompt”, el shell se prepara un entorno (“environment”), que consiste en un conjunto de variables (“shell variables”). Estas variables contienen información sobre

- El nombre del shell que usamos (\$SHELL='/bin/bash')
- El camino de búsqueda (\$PATH='/bin:/usr/bin:....')
- El terminal que usamos (\$TERM='console')
- Y otras constantes que definen nuestro entorno

Notas:

El fichero `/etc/profile` contiene todos los comandos necesarios para crear el entorno de cada usuario. El shell ejecuta este fichero como parte del proceso del “login”. Cada usuario puede configurar su entorno con un fichero `.profile` en su directorio raíz (su “\$home”). Este fichero se ejecuta después del fichero `/etc/profile`. Vamos a ver como podemos configurar nuestro entorno modificando el fichero `.profile`.

Dentro del shell introducimos nuestros comandos, incluso podemos correr pequeños programas escritos en el lenguaje del shell (“shell scripts”). Muchos de los comandos del UNIX son realmente “shell scripts”.

Notas:

## Procesos

El programa que nos permite trabajar con el sistema (“shell”) es uno de los procesos (“processes”) activos en este momento. Cada usuario puede tener muchos procesos activos al mismo tiempo. El sistema operativo se preocupa de proporcionarles acceso a la CPU (y a los demás componentes del ordenador) a todos los procesos activos. Un proceso puede trabajar de forma interactiva (“foreground”), y sin necesidad de interacciones (“background”). Tenemos que aprender cómo controlar nuestros procesos, esto es, cómo crear procesos nuevos, cómo cambiar un proceso del “foreground” al “background” y cómo “matar” a un proceso innecesario. Vamos a ver cuales son los procesos que necesita el sistema operativo para poder funcionar.

Notas:

## Ficheros en UNIX

En UNIX, el concepto de fichero es más amplio que en otros sistemas operativos. Por ejemplo: Un programa que necesita leer datos del teclado no necesita saber como funciona éste, sino que simplemente accede al teclado a través de un fichero. El “keyboard device driver” del kernel se preocupa de interaccionar con el teclado y de facilitar al sistema los datos recibidos en forma de un fichero (“device file”). En lo siguiente vamos a hablar de ficheros en el sentido común de la palabra, pero veremos que el concepto de fichero se aplica también a directorios, periféricos y a la manera de interactuar con el sistema.

En sistemas operativos multi-usuarios es importante tener medios para proteger los ficheros de cada usuario. Los ficheros de sistema requieren una protección especial para garantizar la seguridad del sistema. Cada fichero conlleva, además del nombre, información sobre:

- Quién es el “dueño” del fichero
- Qué pueden hacer los demás usuarios con este fichero: leer(read), escribir(write), ejecutar(execute))

Llamaremos a toda esta información “derechos de acceso”. En UNIX se pueden especificar los siguientes derechos de acceso:

User			Group			Other		
r	w	x	r	w	x	r	w	x
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
400	200	100	40	20	10	4	2	1

En la representación octal sumamos los números que representan cada permiso. En el ejemplo propuesto obtendremos  $400 + 200 + 40 = 640$ . Podemos memorizar esta regla como

UGO-4-2-1

Más tarde, veremos cómo se pueden modificar los derechos de acceso de nuestros ficheros con el comando `chmod`.

Notas:



## Ficheros especiales

Si un comando de UNIX necesita acceder a un “device” del sistema (como el disco duro, el terminal o un puerto de comunicación), lo hace a través de un fichero (“device file”). Estos ficheros no son ficheros reales, más bien representan los “devices” a los que se pueden acceder. En el directorio `/dev` encontramos estos ficheros. Algunos ejemplos son:

- `/dev/hda0` - primera partición del primer disco duro IDE
- `/dev/fd0`, `/dev/fd0H1440` - primer floppy
- `/dev/cdrom` → `/dev/scd0` - CD-ROM
- `/dev/ttyS0` - puerto de comunicación (COM1)
- `/dev/null` - bit bucket (“cubo de basura”)

Los derechos de acceso a estos ficheros definen los derechos de acceso a los “devices” correspondientes.

Notas:

## Consolas virtuales

El Linux ofrece varios terminales virtuales en un mismo terminal físico: usando la combinación de teclas Alt-F1 .. Alt-F8 podemos cambiarnos a “otro” terminal. De esta manera podemos tener varias sesiones abiertas a la vez, y podemos saltar de sesión a sesión cómodamente.

Notas:

## Redirección y Pipes

El UNIX ofrece la posibilidad de cambiar la fuente de entrada (por defecto el teclado) y el destino de salida (por defecto la pantalla) de los comandos. Un comando normalmente comunica con nosotros a través de los tres canales siguientes:

- standard input - teclado
- standard output - pantalla
- standard error output - pantalla

En UNIX podemos “conectar” cada uno de estos canales con un fichero, la salida de un programa puede ser utilizada por otro programa. Por ejemplo, podemos:

- Mandar la salida de un comando a un fichero:

```
comando > fichero.out
```

- Mandar solamente los mensajes de error a un fichero:

```
comando 2> fichero.err
```

- Mandar la salida y los mensajes de error de un comando a un fichero:

```
comando >& fichero.err
```

- Añadir la salida de un comando a un fichero:

```
comando >> fichero.out
```

- Especificar a un comando que lea los datos de entrada de un fichero en lugar de hacerlo desde el terminal:

```
comando < fichero.in
```

- Usar la salida de un comando como datos de entrada para otro comando:

```
comando1 | comando2
```

- Combinar estas posibilidades:

```
comando1 < fichero.in | comando2 > fichero.out
```

Notas:

## Comandos del UNIX/Linux

A continuación, presentaremos una selección de todos los comandos del UNIX. Son mis “cuarenta principales”, es decir los comandos que más a menudo utilizo. Cada comando está representado por una tabla de la forma:

Comando	Descripción
<hr/>	
Forma de invocar	
Opciones	

Las listas de las opciones NO son completas, he elegido las opciones más útiles. Puede resultar muy conveniente prepararse unas tablas similares con los comandos nuevos que aprendemos. Si buscamos un comando para una tarea especial, usaremos el `man -k`, por ejemplo, `man -k delete` nos lleva a:

```

groupdel (1)    - Delete a group
mdel (1)        - Delete an MSDOS file
rename (n)      - Rename or delete a command
rmdir (2)       - Delete a directory
tr (1)          - Translate or delete characters
unlink (2)      - Delete a name and possibly the file
                  it refers to
unset (n)       - Delete variables
userdel (1)     - Delete a user account and related files

```

La información acerca de las opciones de un comando se obtiene con:

```
man comando
```

Aquí “comando” representa el comando acerca del cual queremos obtener información.

**Trabajar con ficheros y directorios**

**ls**      mostrar el contenido de directorios

---

`ls [opciones] [directorio]`

- a    todos los ficheros, incluyendo los invisibles (.)
- l    formato largo (información detallada)
- S    ordenado respecto al tamaño
- F    clasifica los tipos de ficheros
- R    recursivo → todos los subdirectorios
- 1    una columna

**pwd**    imprimir el nombre del directorio actual

---

`pwd`

**cd**      cambiar el directorio actual a otro directorio o a \$home

---

`cd [directorio]`

**mkdir**   crear un directorio nuevo

---

`mkdir [opciones] directorio`

**rmdir**   borrar un directorio vacío. El directorio debe estar vacío.

---

`rmdir [opciones] directorio`

Notas:

**cp** copiar fichero(s) a otro fichero o a un directorio

---

```
cp [opciones] fichero otro_fichero
cp [opciones] fichero directorio
```

- v verbose, dice lo que hace
- i interactivo, pregunta si otro fichero con el mismo nombre existe
- r recursivo, copia subdirectorios y ficheros

**mv** cambiar de nombre un fichero/directorio, o moverlo a otro directorio

---

```
mv [opciones] nombre_antiguo nombre_nuevo
mv [opciones] fichero directorio
```

- v verbose, dice lo que hace
- i interactivo, pregunta si existe otro fichero con el mismo nombre

**rm** borrar un fichero

---

```
rm [opciones] fichero
rm [opciones] -- fichero
```

- v verbose, dice lo que hace
- i interactivo, pregunta si existe otro fichero con el mismo nombre
- r borrar directorios recursivamente ☹️

**ln** crear un “link”

---

```
ln [opciones] fichero link
```

- s link simbólico
- f borra link si ya existe

Notas:

**more**    mirar un fichero de texto

---

`more [opciones] fichero`

**less**    mirar un fichero de texto

---

`less [opciones] fichero`

`+num`    empieza con línea num

**cat**    transmitir varios ficheros a la salida estandar

---

`cat [opciones] lista_de_ficheros`

cat manda la salida por defecto a standard output, por esta razón, hay que usar la redirección para la concatenación de ficheros:

`cat lista_de_ficheros > fichero`

Podemos usar `cat > fichero` para crear un fichero de texto sin usar un editor. Ctrl-D señala el fin del fichero.

**chmod**    cambiar derechos de acceso de un fichero

---

`chmod [opciones] modo fichero`

`-v`    verbose, dice lo que hace

`-R`    recursivo ☹

El modo se puede dar absoluto (=) o relativo (+-) y es de la forma [ugoa][+=[rwx]. También se puede usar la forma octal:

`chmod octal fichero`

Notas:

## El editor vi

Para editar un fichero podemos usar varios editores. El **vi** es un editor muy sencillo, y es imprescindible conocer sus comandos más básicos. El **vi** se invoca con

```
vi fichero
```

Una vez abierto el fichero nos podemos encontrar en dos modos:

- Modo de comandos
- Modo de editar

Los comandos más útiles son:

i	insert, lleva al modo de editar
a	append, lleva al modo de editar
ESC	escape, termina el modo de editar y lleva al modo de comandos
x	delete, borra el carácter actual
dd	line delete, borra la línea actual
yy	yank, copia la línea actual en el buffer
p	put, inserta el buffer en la posición actual
/string	search, busca “string” en el texto
:nnn	goto line, posiciona el cursor en la linea nnn
:q	quit, salir del vi si no se han realizado cambios
:q!	quit, salir del vi sin guardar los cambios
:wq	write and quit, guardar los cambios y salir del vi

Podemos introducir un número antes de usar un comando, repitiendo así el comando varias veces. Por ejemplo

```
5yy  
p
```

guarda 5 líneas en el buffer y después lo inserta en la posición actual.

Notas:

## Compresión y Archivador

**gzip**      comprimir un fichero

---

```
gzip [opciones] fichero
```

El fichero comprimido termina en .gz

**gunzip**    descomprimir un fichero .gz

---

```
gunzip [opciones] fichero.gz
```

La descompresión “quita” el .gz

**tar**        archivador

---

```
tar [opciones] -f fichero.tar directorio
```

- x    extract, sacar del archivo
- c    create, crear archivo nuevo
- r    append, añadir al archivo
- t    type, mostrar el contenido del archivo
- v    verbose, dice lo que hace

Hay muchas más opciones (unidades de cinta etc.), pero las tres formas más comunes son:

```
tar -cvf tarfile.tar .    crear archivo del dir actual
                           y de todos los subdirectorios
tar -tf tarfile.tar        mostrar el contenido del archivo
tar -xvf tarfile.tar       extraer ficheros del archivo
                           al directorio actual
```

Notas:



## Control de procesos

**ps** estado de procesos

---

**ps** [opciones]

- u** formato de “ usuarios”
- a** también procesos de otros usuarios
- x** también procesos sin terminal

**fg** continuar un comando parado interactivamente (bash)

---

**fg** [%n]

**bg** continuar un comando parado en el background (bash)

---

**bg** [%n]

**kill** mandarle un “signal” a un proceso

---

**kill -s signal pid**

- 15** SIGTERM, matarlo si el programa lo detecta
- 9** SIGKILL, matarlo de todas formas
- 19** SIGSTOP, pararlo
- 18** SIGCONT, continuar

**kill -l** da una lista de los posibles señales

En el shell podemos usar la combinación de teclas Ctrl-Z para parar un comando interactivo. La combinación de teclas Ctrl-C termina un comando interactivo.

Notas:

**Información y entorno**

<b>date</b>	mostrar o cambiar fecha y hora del sistema
<hr/>	
date [opciones] [fecha/hora]	
-s set, cambiar fecha y/u hora (sólo root)	
<b>df</b>	mostrar sitio disponible en los discos duros
<hr/>	
df [opciones] [directorio]	
<b>du</b>	mostrar información del sitio ocupado en un directorio
<hr/>	
du [opciones] [directorio]	
<b>env</b>	mostrar el entorno o ejecutar un comando en un entorno especial
<hr/>	
env [opciones] [comando]	
<b>who</b>	información sobre usuarios activos
<hr/>	
who [opciones] [am i]	
<b>passwd</b>	cambiar password
<hr/>	
passwd [usuario]	
Sólo el root puede cambiar el password de otros usuarios	

Notas:

## Comandos útiles en shell-scripts

**echo**      mostrar una línea de texto o una variable de entorno

---

```
echo [opciones] [texto o $VARIABLE]
```

```
ejemplo: echo $PATH
```

**alias**      asignar una equivalencia a un nombre

---

```
alias nombre=equivalencia
```

El alias se puede usar para redefinir comandos, por ejemplo:

```
alias ls='ls -F'
```

**export**    asignar un valor a una variable y hacerla visible

---

```
export variable=valor
```

```
ejemplo: export MYBIN=/home/denk/_bin
```

En un shell script se pueden realizar bucles y cálculos aritméticos, como, por ejemplo, en el bash:

```
let ab=1
echo $ab
while test $ab -le 10
do
    echo $ab
    cp fil fil$ab
    let ab=ab+1
done
```

Notas:

## Trabajar con disquetes del MS-DOS

<code>mmdir</code>	mostrar el contenido de un disquete
<hr/>	
<code>mmdir [-w] [directorio_del_disquete]</code>	
<code>-w</code> sólo los nombres de los ficheros	
<code>mcd</code>	cambiar el directorio actual del disquete
<hr/>	
<code>mcd [directorio_del_disquete]</code>	
<code>mdel</code>	borrar ficheros del disquete
<hr/>	
<code>mdel [-v] [ficheros_del_disquete]</code>	
<code>-v</code> verbose, dice lo que hace	
<code>mcopy</code>	copiar ficheros del/al disquete
<hr/>	
<code>mdel [opciones] fichero_fuente fichero_destino</code>	
<code>-t</code> texto, cambia CR/LF a LF y viceversa	
<code>-v</code> verbose, dice lo que hace	
<code>mformat</code>	formatea un disquete ya preparado con fdformat
<hr/>	
<code>mformat [opciones] disquetera:</code>	
<code>-l</code> permite añadir una etiqueta	
hay que usar, por ejemplo, el comando	
<code>fdformat /dev/fd0H1440</code>	
para preparar (low level format) el floppy antes de usar <code>mformat</code> .	

Notas:

## Comandos para el root

<b>adduser</b>	crear una cuenta para un usuario nuevo
<hr/>	
<b>adduser</b>	
<b>adduser</b> es interactivo. Pregunta por todos los datos necesarios. ¡ Evite la tecla BACKSPACE !	
<b>userdel</b>	eliminar una cuenta de un usuario
<hr/>	
<b>userdel usuario</b>	
<b>-r</b> borra los ficheros del usuario en su \$home	
<b>shutdown</b>	tirar el sistema
<hr/>	
<b>shutdown [opciones] hora [mensaje]</b>	
<b>-r</b> reboot, rebotar después del shutdown	
<b>-h</b> halt, solamente tirar (para apagar)	
Ejemplos:	
<b>shutdown -h now</b> (tirar ahora)	
<b>shutdown -r +10</b> (tirar en diez minutos y rebotar)	
<b>mount</b>	montar un sistema de ficheros
<hr/>	
<b>mount [opciones] device directorio</b>	
Hay varias formas de invocar el comando <b>mount</b> . Explicaremos este comando mas adelante.	
<b>umount</b>	desmontar un sistema de ficheros
<hr/>	
<b>umount directorio</b>	
Este comando se explicará con el comando <b>mount</b> .	

Notas:

## Comandos avanzados

**find** encontrar ficheros y realizar acciones con estos ficheros

---

`find [directorio_inicial] [expresión]`

Expresión:

<code>-name pattern</code>	nombre del fichero
<code>-iname pattern</code>	nombre del fichero, case-insensitive
<code>-size [+_]n</code>	tamaño
<code>-gid n</code>	group id
<code>-uid n</code>	user id
<code>-perm [+_]mode</code>	derecho de acceso

Operadores:

<code>-not expr</code>	verdadero si expr es falso
<code>expr1 -and expr2</code>	verdadero si expr1 y expr2 son verdaderas
<code>expr1 -or expr2</code>	verdadero si expr1 o expr2 es verdadera

Acción:

<code>-print</code>	mostrar ficheros encontrados
<code>-exec comando;</code>	ejecutar comando con fichero encontrado

**grep** encontrar texto en ficheros

---

`grep [opciones] [pattern] ficheros`

<code>-G</code>	pattern=basic reg. expression
<code>-E</code>	pattern=extended reg. expression
<code>-F</code>	pattern=list of strings
<code>-i</code>	case insensitive

pattern - regular expression:

<code>[01234abc]</code>	cualquier carácter de la lista
<code>[a-z]</code>	rango de caracteres
<code>.</code>	cualquier carácter

## 4 Configuración del sistema

Una vez que hemos instalado el Linux y nos sentimos algo seguro con los comandos del UNIX, podemos desear realizar algunas configuraciones de nuestro sistema. La necesidad de cambiar algo en el sistema normalmente surge por los siguientes motivos:

- Un programa necesita ficheros de configuración
- Algún programa/comando no funciona
- Queremos realizar una tarea mas cómodamente

Muchas veces puede resultar bastante difícil averiguar qué tenemos que modificar/configurar. Una receta podría ser la siguiente:

- Leemos detenidamente los mensajes de error en el caso de que algo no funcione.
- Consultamos el `man` de todos los programas relacionados con el problema.
- Buscamos los ficheros que podrían contener información acerca del programa de que se trata (`usr/doc`, `usr/doc/faq` o el directorio donde se ha instalado el programa). Estos ficheros normalmente llevan nombres como `README*`, `readme*`, `INSTALL*`, `install*`, `*.doc`
- Si tenemos acceso a otra máquina ya configurada, podemos comparar todos los ficheros relacionados. Siempre debemos tener en cuenta el hardware y la versión del Linux de ambas máquinas.
- Antes de realizar cambios en cualquier fichero de configuración, creamos una copia de seguridad de este fichero (por ejemplo `passwd` → `passwd.old`, `passwd.OLD`, etc.)
- Debemos tener una idea de lo que hacemos, probar “a ver si funciona” puede provocar una ☹

En este capítulo he resumido algunas configuraciones que he realizado en nuestras máquinas, aquí en la Facultad. ¡ Los pasos siguientes dependen de la distribución del Linux y del hardware que tengamos !

Notas:

## ¿ Dónde están los ficheros ?

La estructura de directorios en una máquina UNIX/Linux es bastante más compleja que en otros sistemas operativos. Hay que conocer el contenido de los directorios más importantes para poder encontrar un fichero rápidamente. El directorio raíz contiene:

<code>bin/</code>	comandos más básicos del UNIX
<code>boot/</code>	ficheros para botar
<code>cdrom/</code>	para poder montar un CD-ROM
<code>dev/</code>	“device files”
<code>dosc/</code>	partición del DOS
<code>etc/</code>	ficheros de configuración
<code>home/</code>	directorios de los usuarios
<code>lib/</code>	bibliotecas para programas
<code>lost+found/</code>	ficheros que se han “perdido”
<code>mnt/</code>	para poder montar otros sistemas de ficheros
<code>proc/</code>	información sobre procesos
<code>root/</code>	“home” del root
<code>sbin/</code>	comandos para el root
<code>tmp/</code>	para ficheros temporales
<code>usr/</code>	véase directorio <code>usr/</code>
<code>var/</code>	véase directorio <code>var/</code>

En el directorio `/usr` encontramos gran parte de la instalación del Linux, los directorios más importantes son:

<code>X11R6/</code>	programas, bibliotecas y manuales del X-Windows
<code>bin/</code>	resto de comandos del UNIX/Linux
<code>doc/</code>	documentación sobre programas instalados
<code>include/</code>	ficheros de “include” (*.h) para código C
<code>info/</code>	ficheros para el sistema “info”
<code>lib/</code>	bibliotecas
<code>local/</code>	binarios, bibliotecas, manuales, etc. para instalaciones “locales”
<code>man/</code>	páginas del <code>man</code>
<code>sbin/</code>	demonios y utilidades para el root
<code>src/</code>	código fuente

En el directorio `/var` hay más ficheros del sistema, los directorios más importantes son:

<code>X11R6/</code>	configuración del X-Windows
<code>adm/</code>	log-files, administración
<code>spool/</code>	ficheros temporales de “spool” (impresora, mail, news, etc.)



El directorio `/etc` contiene varios ficheros importantes para la configuración del sistema. El kernel lee estos ficheros a la hora de botar y monta las particiones, arranca demonios, etc. Algunos ficheros importantes son:

<code>HOSTNAME</code>	nombre de la máquina
<code>exports</code>	directorios a los que otras máquinas pueden acceder
<code>fstab</code>	sistemas de ficheros que se montan a la hora de botar
<code>ftpaccess</code>	controla el acceso vía ftp
<code>hosts</code>	lista de direcciones de máquinas de la red local
<code>inittab</code>	controla la inicialización del sistema (runlevels)
<code>issue</code>	Welcome to Linux ..... login prompt
<code>motd</code>	message of the day (después del login)
<code>organization</code>	nombre de la institución
<code>passwd</code>	passwords de todos los usuarios
<code>printcap</code>	definición de las impresoras
<code>profile</code>	profile por defecto para el bash
<code>rc.d/</code>	directorio que contiene varios “startup-scripts”
<code>skel/</code>	directorio esqueleto para los usuarios

¡No todos los ficheros se editan directamente, algunos son configurados por `netconfig`, `passwd`, `visudo`, etc.!

Notas:

## Configuración de la red

Si nuestro PC está conectado a la red necesitamos pedir un número IP (por ejemplo 150.214.138.93) para el ordenador. El número IP nos permite formar parte del Internet, ya que nos da una dirección única en esta red mundial. Para poder pedir este número hace falta:

- Un nombre único para la máquina dentro de nuestro “domain” (véase abajo).
- Una descripción del hardware.
- El número de la tarjeta ethernet.

El kernel que usamos debe detectar la tarjeta ethernet, al botar podría aparecer, por ejemplo, el siguiente mensaje:

```
eth0:  SMC Ultra at 0x280, 00 00 C0 63 4F AF, IRQ 3
memory 0xd0000-0xd3fff.
```

Los seis dígitos 00 00 C0 63 4F AF componen el número de la tarjeta ethernet. ¡ Cada tarjeta ethernet tiene un número diferente ! Si no aparece ningún mensaje acerca de la tarjeta ethernet, debemos usar otro kernel que la soporte. Muchas tarjetas ethernet traen utilidades para el MS-DOS que permiten averiguar el número de la misma.

Al instalar los disquetes “N” del Linux, podemos correr una utilidad llamada **netconfig** para configurar los programas que acceden a la red. Si a la hora de instalar aún no tenemos el número de IP, podemos usar el **netconfig** posteriormente. Para informarnos sobre el estado de la tarjeta de comunicación, podemos invocar el comando **ifconfig**.

Notas:

## Terminología

La terminología de redes es muy amplia. Paso a enumerar los términos más importantes:

- TCP/IP - Transmission Control Protocol/Internet Protocol, este es el “lenguaje” que hablan los ordenadores entre sí.
- IP-address, es el número que define la dirección de una máquina, por ejemplo, “150.214.138.93” (servix.us.es)
- DNS - Domain Name Service, convención sobre cómo se compone el “nombre” de nuestra máquina, por ejemplo “servix.us.es”
- host name - el nombre de una máquina, por ejemplo “servix” o “servix.us.es”
- domain name - el nombre de la red local a la que estamos conectado. Aquí en la Facultad de Física es “us.es”.
- name server - un ordenador que realiza la traducción DNS  $\rightarrow$  IP-address. Aquí, en la Facultad de Física, el “servidor de nombres” tiene el número IP “150.214.130.10” (opalo).
- gateway - un ordenador que conecta dos redes locales.

Para poder configurar un ordenador que está conectado a la red no hace falta entender todos estos conceptos, la utilidad `netconfig` nos ayuda en los pasos más importantes.

Notas:

### La utilidad `netconfig`

Podemos configurar nuestro sistema de dos formas:

- local loopback mode - si no hay tarjeta ethernet
- conexión real - con tarjeta ethernet y conexión a la red (hace falta el número IP)

El shell script `netconfig` configura ficheros que son necesarios para el funcionamiento de programas que necesitan acceso a la red. Al correr el `netconfig` tenemos que introducir la siguiente información (tomaremos como ejemplo el “servix”):

- hostname: “servix”
- domain name: “us.es”
- ... ONLY use loopback: “No”. Si respondemos con “Yes”, el `netconfig` no realizará las siguientes configuraciones.
- IP address: “150.214.138.93”
- gateway address: “150.214.138.1”
- netmask: “255.255.255.0”
- ... access a nameserver: “Yes”
- Name server for domain us.es: “150.214.130.10”

Una vez configurada la red, rebootaremos el ordenador y probaremos algunos programas que accedan a la red (por ejemplo `telnet`, `ftp`, `mail`). Podemos añadir algunas máquinas locales en el fichero `/etc/hosts`:

```
127.0.0.1      localhost
150.214.138.93 servix.us.es servix
150.214.138.71 idfix1.us.es idfix1
150.214.138.72 idfix2.us.es idfix2
150.214.1.10   obelix.cica.es obelix
150.214.1.6    sevax1.cica.es sevax1
150.214.1.14   sevax2.cica.es sevax2
```

Notas:

## Instalación del LOADLIN

Si hemos instalado el LOADLIN, encontraremos el fichero `lodlin15.zip` en el directorio `/root`. LOADLIN permite botar el Linux desde el MS-DOS. Para instalar el LOADLIN hay que seguir los siguientes pasos:

- Copiar el fichero `lodlin15.zip` a la partición del DOS. Si durante la instalación, hemos hecho “visible” esta partición para el Linux, podemos copiar el fichero con `cp`. También podemos usar el `mcopy` para copiar este fichero a un disquete y posteriormente, al disco duro desde el DOS.

```
cp lodlin15.zip /dosd/lodlin15.zip
```

o

```
mcopy lodlin15.zip a:
```

- Copiar el kernel a la partición del DOS. El kernel normalmente se encuentra en el directorio raíz con el nombre `vmlinuz`. Podemos utilizar el mismo método que indicamos arriba para copiar el kernel a la partición del DOS.
- Salimos del Linux (rebotando) y entramos en el DOS. Creamos un directorio LOADLIN y descomprimos el fichero LODLIN15.ZIP en este directorio con el programa PKUNZIP:

```
PKUNZIP LODLIN15.ZIP
```

- Editamos el fichero LINUX.BAT, por ejemplo:

```
c:\loadlin\loadlin c:\vmlinuz root=/dev/hda2 ro
```

Si ejecutamos el fichero LINUX.BAT desde el MS-DOS, el Linux se bota con el kernel `c:\vmlinuz`, montando la partición `/dev/hda2` como directorio raíz. Podemos especificar otros parámetros para el LOADLIN. En el directorio donde hemos descomprimido el LODLIN15.ZIP encontraremos ficheros que contienen más información acerca de este programa.

Notas:

También podemos incluir un menú de arranque en el fichero CONFIG.SYS, por ejemplo:

```
[menu]
menuitem=Linux, Linux
menuitem=Msdos, MS-DOS
menudefault=Linux, 10
[Common]
[Msdos]
DEVICE=C:\DOS\SETVER.EXE
DEVICE=C:\DOS\HIMEM.SYS
DOS=HIGH
COUNTRY=034,,C:\DOS\COUNTRY.SYS
DEVICE=C:\DOS\DISPLAY.SYS CON=(EGA,,1)
FILES=30
STACKS=9,256
[Linux]
```

En este ejemplo de CONFIG.SYS, el MS-DOS nos presenta un menú de arranque que permite elegir si queremos arrancar el Linux o el MS-DOS. Si no elegimos nada en 10 segundos, automáticamente se arrancará el Linux. El fichero AUTOEXEC.BAT tendrá la siguiente forma:

```
@ECHO OFF
goto %config%

:Msdos
C:\DOS\SMARTDRV.EXE /X
PROMPT $p$g
SET TEMP=C:\TEMP
MODE CON CODEPAGE PREPARE=((850) C:\DOS\EGA.CPI)
MODE CON CODEPAGE SELECT=850
KEYB SP,,C:\DOS\KEYBOARD.SYS
C:\MOUSE\MOUSE
PATH=C:\;C:\WINDOWS;C:\DOS;C:\PKZIP
goto end

:Linux
C:\LOADLIN\LINUX.BAT
goto end

:end
```

## Las utilidades para disquetes en formato MS-DOS

Los programas `mcopy`, `mcd`, `mdir`, `mdel`, `mformat`, etc. permiten acceder a disquetes formateados para el MS-DOS. Como ya hemos visto, el Linux accede a la disquetera a través de un fichero especial que se encuentra en el directorio `/dev`. Tal como viene instalado el Linux, solamente el usuario `root` tiene acceso a estos dispositivos. Para que todos los usuarios puedan usar la disquetera hay que cambiar los permisos de estos ficheros:

```
chmod a+r,a+w /dev/fd0
chmod a+r,a+w /dev/fd0H1440
```

De esta manera podemos incluso formatear un disquete para el MS-DOS desde el Linux:

```
fdformat /dev/fd0H1440
mformat a:
```

De todas formas, hay que tener en cuenta que el Linux es un sistema operativo multi-usuario: configurado de esta forma, todos los usuarios pueden modificar la información de un disquete que se encuentre en la disquetera. ¡ Si insertamos un disquete con datos importantes, otro usuario (conectado con la misma máquina por la red) puede formatearlo ! Hay que advertirles a todos los usuarios que sólo se debe formatear un disquete cuando se está seguro de que se formatea en la máquina local. No dejaremos disquetes en la disquetera.

Notas:

## La Configuración del X-Windows

La configuración del X-Windows es el paso más difícil de la instalación. Debemos actuar con cuidado, ya que, si no definimos los parámetros correctamente, podemos dañar al hardware (monitor). Hay que dar dos pasos:

- Elegir el servidor que vamos a usar. Durante la instalación hemos podido instalar varios servidores de X. Hay que definir cuál se va a usar.
- Crear el fichero `XF86Config`. Este fichero contiene toda la información que el X necesita acerca del hardware que tenemos.

Podemos realizar estos dos pasos manualmente o con la ayuda del programa `xf86config`. Los mensajes que imprime este programa son difíciles de entender para un principiante. Por esta razón vamos a explicar en primer lugar el método manual. Una vez que hemos entendido los conceptos básicos del hardware involucrado (modos de Monitor, tarjetas gráficas), podemos usar el `xf86config` como un “experto”, obteniendo una configuración mejor del X-Windows.

Notas:



### Elegir el servidor de X

El servidor de X se define mediante un enlace simbólico (symbolic link) que vincula el fichero `/var/X11R6/bin/X` con el servidor de X, como por ejemplo (`ls -l`):

```
/var/X11R6/bin/X -> /usr/X11R6/bin/XF86_S3*
```

Si arrancamos el sistema X-Windows mediante el comando `startx`, se ejecutará el fichero `/var/X11R6/bin/X`. Si este fichero apunta al servidor que queremos usar, este servidor de X arrancará. Podemos cambiar el servidor de X mediante el comando `ln`:

```
ln -fs /usr/X11R6/bin/servidor /var/X11R6/bin/X
```

Aquí “servidor” determina el servidor que queremos usar, por ejemplo:

- XF86\_Mach64, tarjetas Mach 64, acelerado
- XF86\_Mono, servidor monocolor
- XF86\_S3, tarjetas S3, acelerado
- XF86\_VGA16, modo estándar VGA con 16 colores
- ...

Con el comando

```
ls /usr/X11R6/bin/XF86_*
```

podemos averiguar qué servidores se han instalado. Si no aparece el servidor que necesitamos para nuestra tarjeta, tendremos que instalarlo de nuevo. El programa de instalación actualiza el enlace simbólico del fichero `/var/X11R6/bin/X` al servidor.

Notas:

## El fichero XF86Config

En el fichero `XF86Config` definimos el hardware que estamos usando. Hay dos lugares donde podemos colocarlo:

- En el directorio `/etc` con los demás ficheros de configuración.
- En el directorio `/usr/X11R6/lib/X11`

Resulta más conveniente trabajar en el directorio `/usr/X11R6/lib/X11`, ya que todos los ficheros de documentación se encuentran en el directorio `/usr/X11R6/lib/X11/doc`. Podemos consultar estos ficheros:

- `README.Config` - Cómo configurar el fichero `XF86Config`
- `README.tarjeta` (`tarjeta=S3`, `P9000`, `Video7`, etc.) - información especial para nuestra tarjeta
- `Monitors` - configuraciones para algunos monitores
- `Device` - configuraciones (sección “Device”) para muchas tarjetas gráficas
- `modeDB.txt` - más configuraciones para tarjetas gráficas y monitores
- `AccelCards` - configuraciones para tarjetas aceleradas
- `VideoModes.doc` - explicación detallada de cómo funciona el monitor y la tarjeta gráfica. Para una configuración avanzada.

Si nuestra tarjeta y monitor aparecen especificados en estos ficheros, no surgirán mayores dificultades. En este caso podemos transferir las especificaciones al fichero `XF86Config`.

Notas:

Podemos usar el fichero `/usr/X11R6/lib/X11/XF86Config.eg` como fichero de partida:

```
cd /usr/X11R6/lib/X11
cp XF86Config.eg XF86Config
chmod u+w XF86Config
```

Debemos asegurarnos de que sólo haya un fichero `XF86Config` en todo el sistema:

```
find / -name XF86Config
```

Ahora editamos el fichero `XF86Config`. Este fichero contiene varias secciones:

- Files - aquí se define dónde se encuentran el base de datos RGB y los directorios que contienen las fuentes del X.
- Keyboard - aquí se puede modificar la definición del teclado.
- Pointer - en esta sección se define el tipo de ratón que usamos.
- Monitor - definición del monitor
- Mode - definición de varios modos de vídeo
- Device - definición de la tarjeta gráfica que usamos
- Screen - combinación de un “monitor”, un “device” y un “driver”(servidor) con varios modos de vídeo.

En el fichero `README.Config` se explican más detalladamente estas secciones. ¡ Hay que estudiar este documento antes de editar el fichero `XF86Config` ! En las dos páginas siguientes veremos dos ejemplos para el fichero `XF86Config`:

- `XF86Config.vga16` - configuración para el servidor `XF86_VGA16`, esta configuración debería funcionar con la mayoría del hardware disponible en el mercado. Con este fichero no se aprovechan las posibilidades de una tarjeta gráfica avanzada (solo 16 colores, resolución estándar VGA, ningún soporte de funciones aceleradas de la tarjeta).
- `XF86Config.pvga14` - configuración para una tarjeta S3-864 con “RamDac” y “ClockChip” en combinación con un monitor Philips 14C. Este fichero es un ejemplo para una configuración especial. ¡ No use este fichero para su equipo, ya que probablemente su configuración será distinta !

Notas:

Dec 17 1995 15:40	XF86Config.vga16	Page 1
<pre># ***** # Files section. This allows default font and rgb paths to be set # ***** Section "Files" # The location of the RGB database.     RgbPath      "/usr/X11R6/lib/X11/rgb" # Multiple FontPath entries are allowed (which are concatenated together),     FontPath     "/usr/X11R6/lib/X11/fonts/misc/"     FontPath     "/usr/X11R6/lib/X11/fonts/Type1/"     FontPath     "/usr/X11R6/lib/X11/fonts/Speedo/"     FontPath     "/usr/X11R6/lib/X11/fonts/75dpi/"     FontPath     "/usr/X11R6/lib/X11/fonts/100dpi/" EndSection # ***** # Server flags section. # ***** Section "ServerFlags" EndSection # ***** # Keyboard section # ***** Section "Keyboard"     Protocol     "Standard"     AutoRepeat   500 5 # To set the LeftAlt to Meta, RightAlt key to ModeShift, # RightCtl key to Compose, and ScrollLock key to ModeLock: # # DENK: activated these #   LeftAlt  Meta #   RightAlt ModeShift #   RightCtl Compose #   ScrollLock ModeLock EndSection # ***** # Pointer section # ***** Section "Pointer"     Protocol     "Microsoft"     Device       "/dev/mouse" # DENK: Activated 3 buttons #   Emulate3Buttons EndSection # ***** # Monitor section # ***** # Any number of monitor sections may be present Section "Monitor"     Identifier   "Generic Monitor"     VendorName   "Unknown"</pre>		

Dec 17 1995 15:40	XF86Config.vga16	Page 2
<pre>    ModelName    "Unknown" # HorizSync is in kHz unless units are specified.     HorizSync    31.5 # typical for a single frequency fixed-sync monitor # # HorizSync     30-64 # multisync # HorizSync     31.5, 35.2 # multiple fixed sync frequencies # VertRefresh is in Hz unless units are specified.     VertRefresh  60 # typical for a single frequency fixed-sync monitor # # VertRefresh   50-100 # multisync # Modes can be specified in two formats. A compact one-line format, or # a multi-line format. # A generic VGA 640x480 mode (hsync = 31.5kHz, refresh = 60Hz) # These two are equivalent # # ModeLine "640x480" 25.175 640 664 760 800 480 491 493 525 # # Mode "640x480" #   DotClock      25.175 #   HTimings       640 664 760 800 #   VTimings       480 491 493 525 #   EndMode EndSection # ***** # Graphics device section # ***** # Any number of graphics device sections may be present Section "Device"     Identifier   "Generic VGA"     VendorName   "Unknown"     BoardName    "Unknown"     Chipset      "generic" EndSection # ***** # Screen sections # ***** # The 16-colour VGA server Section "Screen"     Driver       "vga16"     Device       "Generic VGA"     Monitor      "Generic Monitor"     Subsection   "Display"         Modes     "640x480"         ViewPort  0 0         Virtual    800 600     EndSubsection EndSection</pre>		

4 CONFIGURACIÓN DEL SISTEMA

Dec 17 1995 15:43	XF86Config.pvga14	Page 2
<pre># ***** # Files section. This allows default font and rgb paths to be set # *****  Section "Files"     RgbPath      "/usr/X11R6/lib/X11/rgb"     FontPath     "/usr/X11R6/lib/X11/fonts/misc/"     FontPath     "/usr/X11R6/lib/X11/fonts/Type1/"     FontPath     "/usr/X11R6/lib/X11/fonts/Speedo/"     FontPath     "/usr/X11R6/lib/X11/fonts/75dpi/"     FontPath     "/usr/X11R6/lib/X11/fonts/100dpi/" EndSection  # ***** # Server flags section. # *****  Section "ServerFlags" EndSection  # ***** # Keyboard section # *****  Section "Keyboard"     Protocol     "Standard"     AutoRepeat   500 5     # DENK: activated these     LeftAlt      Meta     RightAlt     ModeShift     RightCtl     Compose     ScrollLock    Modelock EndSection  # ***** # Pointer section # *****  Section "Pointer"     Protocol     "Microsoft"     Device       "/dev/mouse"     # DENK: Activated 3 buttons     Emulate3Buttons EndSection  # ***** # Monitor section # *****  # Any number of monitor sections may be present  Section "Monitor"     Identifier    "My Monitor"     VendorName    "Philips"     ModelName     "14C"      HorizSync    31.5, 35.16, 35.5 # Philips 14C     VertRefresh   50-100           # multisync, Philips 14C      # A generic VGA 640x480 mode (hsync = 31.5kHz, refresh = 60Hz)</pre>		

Dec 17 1995 15:43	XF86Config.pvga14	Page 2
<pre>Mode "640x480"     DotClock      25.175     HTimings      640 664 760 800     VTimings      480 491 493 525 EndMode  # DENK:The 800x600 VESA/56 mode, as copied from README.Config. # This mode is compatible with the Philips 14C, but results in a # very flickery display Mode "800x600"     DotClock      36     HTimings      800 824 896 1024     VTimings      600 601 603 625 EndMode  EndSection  # ***** # Graphics device section # ***** # Any number of graphics device sections may be present  Section "Device"     Identifier    "Generic VGA"     VendorName    "Unknown"     BoardName     "Unknown"     Chipset       "generic" EndSection  Section "Device"     Identifier    "PVGA-S3"     VendorName    "unknown"     BoardName     "PVGA-S3"     Chipset       "s3_generic"     VideoRam      1024     ClockChip     "s3_sdac"     Rambac        "s3_sdac" EndSection  # ***** # Screen sections # *****  # The accelerated servers (S3, Mach32, Mach8, 8514, P9000, AGX, W32)  Section "Screen"     Driver        "accel"     Device        "PVGA-S3"     Monitor       "My Monitor"     Subsection    "Display"         Depth     8         Modes      "800x600"         ViewPort   0 0         Virtual    1000 800     EndSubsection     Subsection    "Display"         Depth     8         Modes      "640x480"         ViewPort   0 0         Virtual    1000 800     EndSubsection EndSection</pre>		

## Probar la configuración

Antes de probar la configuración hay que asegurarse de que las especificaciones del monitor son las correctas. ¡ Una tarjeta gráfica moderna puede producir señales con una frecuencia muy alta y una frecuencia demasiado alta puede dañar al monitor (especialmente monitores que no sean “multi-sync”) ! Podemos arrancar el X-Windows con el comando

```
startx
```

Si no funciona, podemos usar la opción **-probeonly** del servidor de X para averiguar los parámetros que faltan en la configuración (por ejemplo CLOCK-rates):

```
X -probeonly >& /tmp/xoutput
```

¡ Si el X-Windows no arranca bien, hay que leer toda la documentación para nuestra tarjeta !

Notas:

## El programa `xf86config`

Una alternativa a la edición manual del fichero `XF86Config` es el uso del programa

`xf86config`

Esta utilidad nos ayuda a crear el fichero `XF86Config`. En varios diálogos tenemos que especificar los componentes del hardware que usamos. El `xf86config` accede a una base de datos de tarjetas gráficas y, si nuestra tarjeta figura en ella, simplifica considerablemente la configuración del X-Windows. De todas formas es aconsejable leer el fichero `README.Config` antes de usar este programa. Debemos tener preparada toda la información que tengamos sobre nuestro monitor, ratón y tarjeta gráfica.

El resultado del `xf86config` es el fichero `/etc/XF86Config`. También es posible crear el enlace del fichero `/var/X11R6/bin/X` con el servidor de X con la ayuda del programa `xf86config`.

Notas: