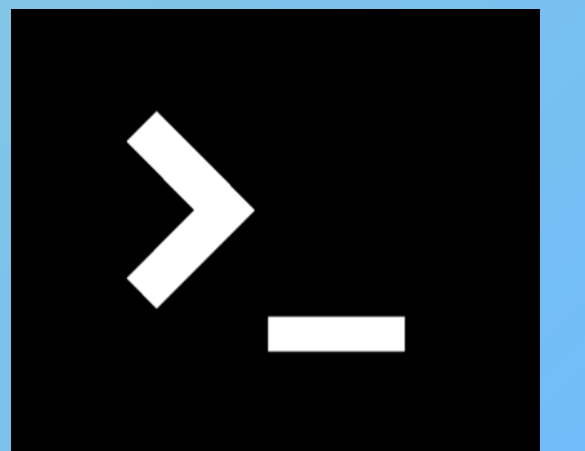


Linux y Terminal

Clase 07



◀ Índice ▶

Comandos y ficheros

¿Qué es un comando?

Partes de un comando

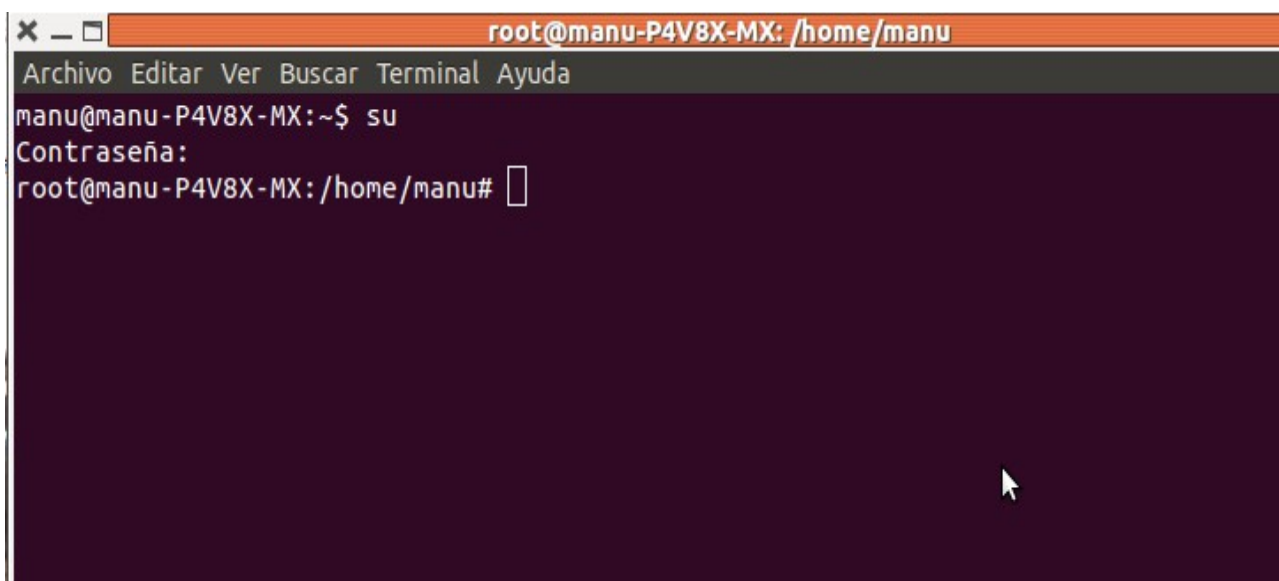
Estructura de ficheros de un sistema Linux

¿Qué es un fichero?

¿Qué es la extension de un fichero?

7 Comandos en Linux

- Vamos a ver unos comandos del sistema operativo Linux que utilizaremos cuando trabajemos en modo texto.
- Los comandos los ejecutaremos escribiéndolos en el terminal y son interpretados por la Shell.
- Los sistemas operativos que distinguen las mayúsculas de las minúsculas se les llama Case-Sensitive.
- Una vez abierto un terminal para escribir los comandos deberemos tener muy en cuenta que se distingue entre las minúsculas y las mayúsculas, no se considera por ejemplo, el mismo carácter “a” que “A”.
- Para acceder al modo texto lo podremos hacer de varias formas:
 - Pulsando a la vez la combinación de teclas CTRL+ATL+F1 o hasta el F6 y se nos abrirá una ventana en modo texto. Para volver al entorno gráfico pulsaremos la combinación CTRL+ATL+F7.
 - Desde el entorno gráfico abriremos el programa terminal mediante el menú Aplicaciones → Accesorios → Terminal (esto en Gnome)
- Para que el sistema operativo espere a que escribamos un comando, dispone de un indicador o PROMPT que generalmente termina en el símbolo \$, además de mostrarnos alguna información como en qué carpeta estamos, el nombre de la máquina, el nombre del usuario, etc.. Esta información la podemos configurar a nuestro gusto.
- Si el usuario es el 'root' o administrador del sistema, el PROMPT termina en el símbolo #



The screenshot shows a terminal window titled 'root@manu-P4V8X-MX: /home/manu'. The window has a menu bar with 'Archivo', 'Editar', 'Ver', 'Buscar', 'Terminal', and 'Ayuda'. The terminal content shows the user 'manu' at 'manu-P4V8X-MX' in the home directory (~) with a '\$' prompt. They enter the command 'su', and the prompt changes to 'root@manu-P4V8X-MX: /home/manu#' with a '#' symbol. A mouse cursor is visible at the bottom right of the terminal area.

7.1 Histórico de comandos

- Una vez que hemos escrito algún comando, si queremos volver a utilizarlo, no es necesario volver a escribirlo si no que utilizaremos las flechas del cursor para buscarlo.
- La flecha de arriba ↑ nos muestra comandos anteriores y la flecha hacia abajo ↓ nos muestra comandos posteriores.

7.2 Función autocompletar

- Si vamos a escribir el nombre de un fichero o directorio como argumento de un comando podemos escribir la primera o primeras letras del fichero o directorio y al pulsar la tecla tabulador , entonces se terminará de escribir el nombre del fichero o directorio.
- Si hay varios ficheros que coinciden en el empezar con los mismos caracteres, se rellenará hasta que tenga un carácter diferente. Si el nombre tuviera un espacio en blanco, el sistema utilizaría el carácter de barra invertida \ antes del espacio.

7.3 Sintaxis de los Comandos

- Antes de ver los comandos vamos a ver la sintaxis de los mismos.
- En general será:
 - “\$ comando [-opciones] [argumento]

NOTA:

Los corchetes significan que pueden o no aparecer en las opciones y/o en los argumentos. (que hay comandos que ni tienen opciones ni tiene argumentos)

NOTA:

Las opciones van precedidas del signo menos – y se introducen cuando queremos cambiar en un comando su funcionamiento o la información que nos muestra.

NOTA:

Los argumentos son los ficheros o directorios sobre los que queremos que se ejecute ese comando.

8 Primeros comandos

NOTA:

Para buscar información sobre los comandos existe un comando especial llamado “man”. Para buscar información sobre el comando “cal” escribiremos **man cal**

Para salir del manual le damos a la “q”.

También podemos buscar información de un comando añadiéndole la opción *--help*, como por ejemplo **cal --help**

Comando	<i>d a t e</i>	
Definición	Muestra día y hora, además de permitirnos cambiarlos	
Sintaxis	date [-opción] [+formato]	
Opciones	-s “nueva fecha”	sirve para poner la fecha y la hora que nos indique la cadena de texto. Hay que tener permisos de Administrador
Formato	+%B	Nos muestra el mes, con letras
	+%j	Número de día, en el año en curso
Ejemplos	<pre>profesor@profesorAsus15:~\$ sudo date -s "1999-03-25" jue mar 25 00:00:00 CET 1999 profesor@profesorAsus15:~\$ date +%B/%j octubre/304</pre>	

- Cambiar la fecha y la hora:
 - *date --set="2 oct 2013 13:43"*
- **Comando hwclock**
 - Muestra la fecha y hora del reloj interno hardware
 - *sudo hwclock --show*
 - Actualiza la fecha y hora del reloj de la BIOS o Hardware a la hora que tiene el sistema operativo
 - *sudo hwclock --systohc*
 - Actualiza la fecha y hora del sistema operativo a la hora que tiene el reloj de la BIOS o Hardware
 - *sudo hwclock --hctosys*

Comando	<i>c a l</i>	
Definición	Muestra el calendario del mes y el año que se le indique. Si no se le especifica nada muestra el mes actual.	
Sintaxis	cal [-opción]	
Opciones	-m NumMes	Muestra el calendario del mes indicado
	-3	Muestra el calendario del mes anterior, mes actual y mes siguiente.
Ejemplos	<pre>profesor@profesorAsusi5:~\$ cal Octubre 2012 do lu ma mi ju vi sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 profesor@profesorAsusi5:~\$ cal -m 3 Marzo 2012 do lu ma mi ju vi sa 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 profesor@profesorAsusi5:~\$ cal -3 Septiembre 2012 Octubre 2012 Noviembre 2012 do lu ma mi ju vi sa do lu ma mi ju vi sa do lu ma mi ju vi sa 1 1 2 3 4 5 6 1 2 3 2 3 4 5 6 7 8 7 8 9 10 11 12 13 4 5 6 7 8 9 10 9 10 11 12 13 14 15 14 15 16 17 18 19 20 11 12 13 14 15 16 17 16 17 18 19 20 21 22 21 22 23 24 25 26 27 18 19 20 21 22 23 24 23 24 25 26 27 28 29 28 29 30 31 25 26 27 28 29 30 30</pre>	

- Mostrar un mes en concreto
 - `cal Nov 2015`

Comando	<i>w h o</i>	
Definición	Muestra quién está conectado en el sistema	
Sintaxis	who [-opción]	
Opciones	-a	sirve para mostrar información adicional.
	-b	nos muestra a que hora arrancó el sistema.
Argumentos		

- Comando !!**
 - Sustituye al último comando ejecutado en el terminal
 - `sudo !!`
 - Ejecuta como superusuario el último comando procesado

Comando	<i>w h o a m i</i>	
Definición	Nos muestra el nombre del usuario	
Sintaxis	whoami	

Comando	<i>c l e a r</i>	
Definición	Limpia la pantalla del terminal	
Sintaxis	clear	

Comando	<i>w h e r e i s</i>	
Definición	Busca donde está dentro del sistema el comando que se le pasa como argumento	
Sintaxis	whereis <i>argumento/s</i>	
Opciones		
Argumentos	Los comandos que queremos localizar, separados por espacios	
Ampliación:	Buscar en que carpeta/s se encuentra el comando cal	

Comando	<i>u n a m e</i>	
Definición	Muestra información sobre el sistema operativo y el equipo	
Sintaxis	uname [-opción/]	
Opciones	-a	muestra toda la información
Argumentos		
Ampliación:	Buscar la opción que nos muestra únicamente la versión el nucleo.	

- **Fichero /etc/issue**
 - Muestra la versión de Ubuntu de nuestro sistema operativo
 - *cat /etc/issue*

Comando	<i>p a s s w d</i>	
Definición	Cambia la contraseña del usuario	
Sintaxis	passwd	
Opciones		
Argumentos		
Nota	Hay que tener cuidado ya que no permite que la contraseña antigua sea la misma que la nueva	

Comando	<i>h i s t o r y</i>	
Definición	Obtiene lista de los últimos comandos que hemos introducido en la consola. Por defecto te muestra los últimos 200	
Sintaxis	History [argumentos]	
Opciones		
Argumentos	Num nos muestra los últimos Num comandos introducidos	
Ejemplo		

Comando	<i>e x i t</i>
Definición	Termina la sesión de usuario o cierra el usuario.
Sintaxis	exit

1 El sistema de archivos

- Un S.O. debe administrar la información ubicada en los dispositivos de almacenamiento, para lo que utiliza un conjunto de procedimientos y estructuras de datos llamadas sistemas de archivo.
- La función del sistema de archivo es almacenar, organizar, acceder y recuperar la información almacenada en forma de archivo.
- El sistema de archivos de cualquier SO es de tipo jerárquico.



Ilustración 1: sistema de archivos jerárquico

- Cada SO suele tener un sistema de archivos propio, a continuación veremos los más utilizados:
 - **FAT**
 - Eran los más utilizados y se introdujeron a partir del MS-DOS, de FAT existen 2 tipos:
 - **FAT 16**
 - Tenía las limitaciones como que, el tamaño de las particiones no podían ser superior a 2 Gigas, el nombre del fichero no podía tener más de 8 caracteres para el nombre y 3 caracteres para la extensión.

- **FAT 32**
 - Introducido por Windows 95 pero realmente usado a partir del Windows 98, es una mejora del anterior que soporta particiones de mayor tamaño y el nombre de los archivos podía tener más caracteres.
- **NTFS (Sistema de Archivos de Nueva Tecnología)**
 - Introducido por Windows NT, es una mejora del sistema FAT que introduce mayor seguridad y un mayor tamaño de particiones y archivos.
 - Los utilizan los sistemas Windows NT, 2000, XP, Vista, Windows 7 y W8, y las versiones Server (2000, 2003, 2008 y 2012)
- **Ext2, Ext3 y Ext4.**
 - Son los sistemas de archivos soportados por la mayoría de las distribuciones Linux.
 - Ext3 es una mejora de Ext2 y la Ext4 una de la Ext3, en mayores volúmenes, manejos de ficheros más grandes, rapidez, etc... pero los 3 son compatibles entre ellos.
- **Reiser4 / ReiserFS**
 - Normalmente, para ficheros de tamaño pequeño tiene mejor rendimiento que ext2 y ext3.
- **YAFFS**
 - Es el sistema de archivos soportado por Android en sus primeras versiones (hasta la 2.3) que pasó a utilizar Ext4, ya que estaba diseñado para ser utilizado en sistemas de un solo núcleo.
- **HFS**
 - Desarrollado por Apple para el sistema operativo Mac- Os.
- **HFS+**
 - Es una mejora del HFS y además de ser utilizado por las últimas versiones de Mac- Os, es el utilizado por los dispositivos de Apple como el iPad.
- **CD FS**
 - Sistema de archivos de CD-ROM compatible con la ISO 9670.
- **UDF**
 - Es el formato utilizado por los grabadores BluRay.
- **NFS (Sistema de ficheros de red)**
 - Suele ser la opción por defecto para sistema de ficheros en red sobre GNU/Linux.
- **CIFS (Sistema de ficheros comunes de Internet)**
 - También conocido como SMB o Samba, permite el intercambio de ficheros en redes mixtas (con Windows y GNU/Linux), así como la compartición de impresora.
- **VFS (Sistema de ficheros Virtual)**
 - Se trata de un interfaz entre el núcleo (kernel) y el sistema de ficheros real.
- **SysFS**
 - Es un sistema de ficheros virtual que proporciona el kernel 2.6 de Linux.
- **SWAP**
 - Del inglés “intercambiar”, es el espacio de disco duro (puede ser un fichero o una partición) que se usa para guardar el estado de procesos que no se utilizan (o no caben) en la memoria física.
- **Fuseblk (Sistema de ficheros del usuario)**
 - Utilizado en sistemas operativos UNIX, para hacer referencia a particiones NTFS no

- montadas, y que por tanto no pueden utilizarse, hasta su montaje.
 - No es un sistema de archivos como tal.
- **VFAT**
 - En GNU/Linux utilizado para las particiones FAT.
 - No es un sistema de archivos como tal.

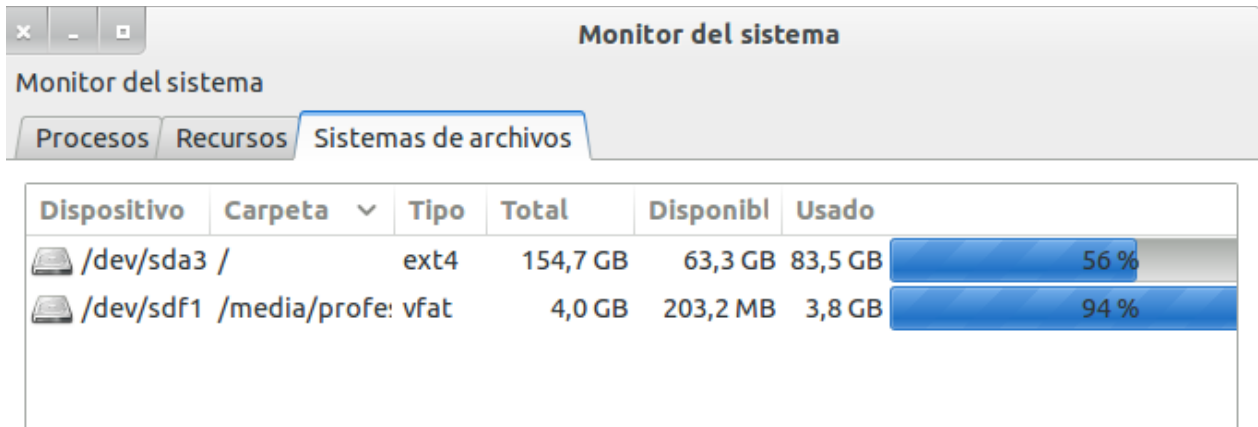
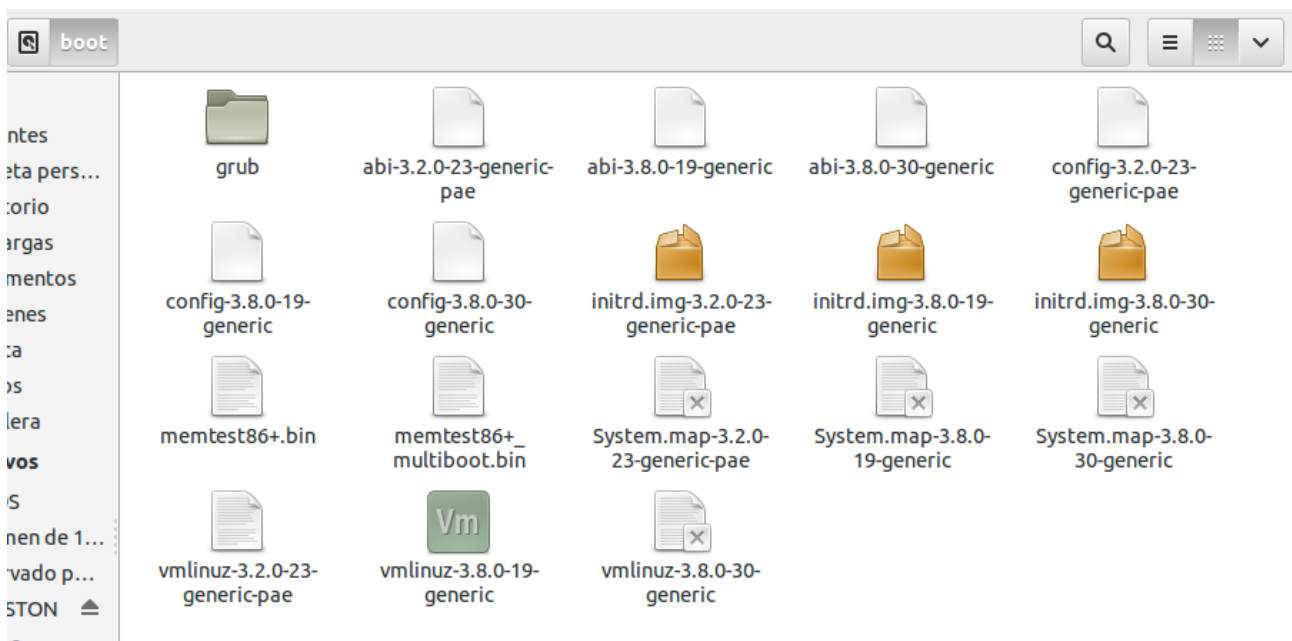


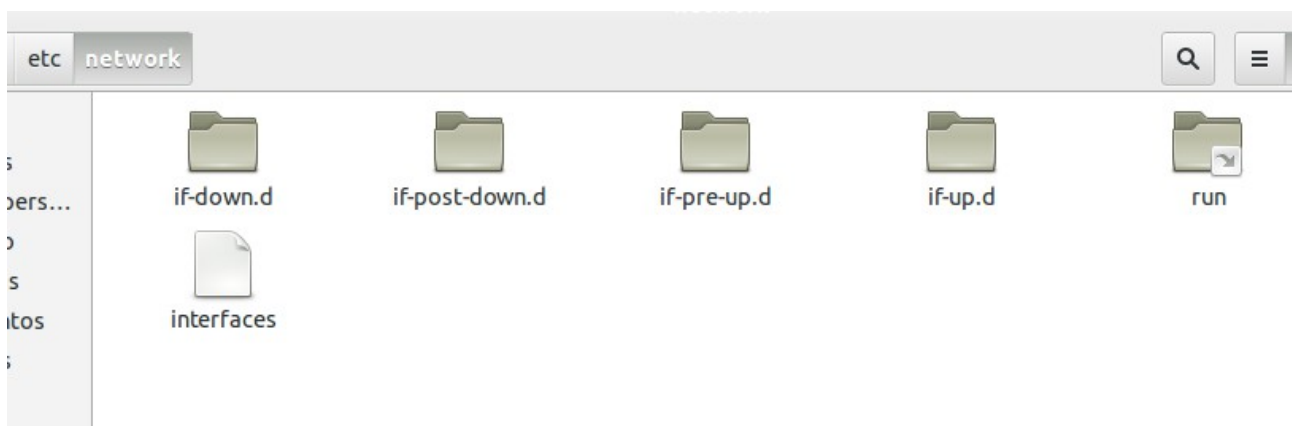
Ilustración 2: Sistema de archivos localizable en Aplicaciones -> Herramientas de Sistema -> Monitor de Sistema

2 Estructura del sistema de archivos

- El sistema de archivos en Linux está basado en el concepto de fichero ya que para Linux *todo es tratado como un fichero*, ya sea un archivo, un directorio, un dispositivo, etc.
- En Linux *cada fichero pertenece a un usuario y a un grupo* al que pertenece el usuario y pueden tener *permiso de lectura, escritura y/o ejecución*.
- El sistema de archivos de Linux tiene una estructura jerárquica o de árbol, en donde todos los ficheros y directorios cuelgan de uno llamado raíz que se representa con el carácter **/**.
- Algunos directorios de Linux son los siguientes:
 - **/home**
 - Contiene las carpetas personales de los usuarios y a través de ellas podemos acceder al escritorio.
 - **/bin**
 - Contiene los programas que puede ejecutar todos los usuarios del sistema.
 - Son archivos binarios o ejecutables.
 - **/boot**
 - Contiene los archivos de configuración de arranque del sistema.
 - Incluye /boot/grub el cual contiene el gestor de arranque **GRUB**.

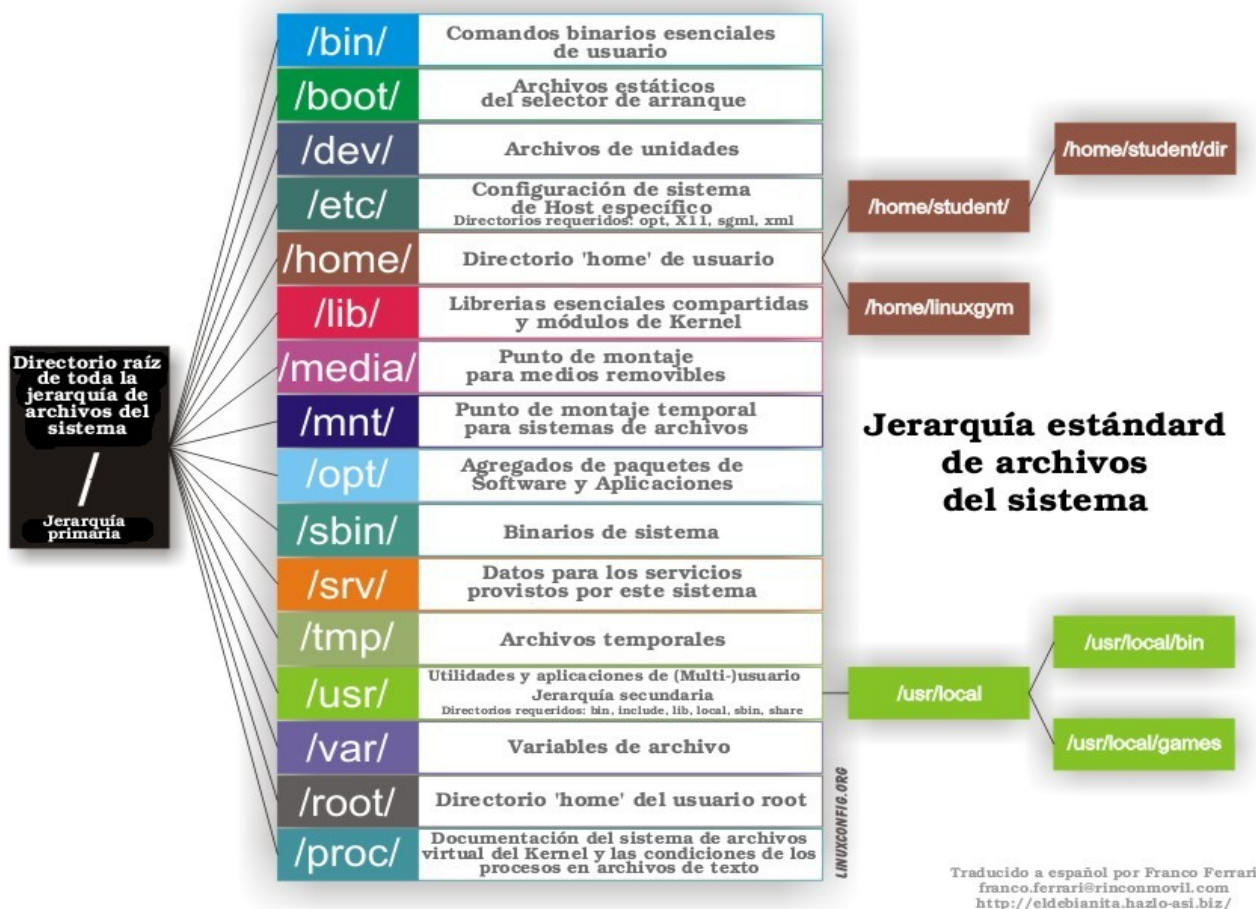


- **/dev**
 - Contiene los archivos de los dispositivos HW del sistema.
- **/etc**
 - Contiene los archivos de configuración del sistema.
 - **/etc/init.d**
 - En este subdirectorio se encuentran todos los scripts de inicio y cierre de los procesos demonios o daemons, que son un tipo especial de proceso que se ejecutan en segundo plano para dar algún tipo de servicio.
 - Estos scripts suelen tener como argumentos las palabras start, stop y restart.
 - **/etc/network**
 - Contiene los scripts necesarios para habilitar y deshabilitar las interfaces de redes del equipo.



- **/lib**
 - Contiene las bibliotecas o librerías que se ejecutan para que funcionen los programas

- que tenemos en el /bin y en el /sbin.
- **/media**
 - Contiene las unidades físicas que tenemos montadas como discos duros, pendrives, unidades de DVD, etc.
 - **/mnt**
 - Lugar donde montan sistemas de archivos temporales. En desuso.
 - **/opt**
 - Lugar donde se pueden instalar aplicaciones que no vienen en los repositorios y que por tanto no se instalan automáticamente.
 - **/proc**
 - Contiene archivos que envían o reciben información del núcleo.
 - Estos ficheros nos permiten obtener información del sistema.
 - Los archivos de este directorio tienen un tamaño de 0 kb, pero sin embargo contienen mucha información.
 - **/root**
 - Es el directorio personal del superusuario del sistema.
 - **/sbin**
 - Contiene programas que solo puede ejecutar el administrador.
 - **/usr**
 - Lugar donde se encuentran los programas instalados de forma automática a través de la red.
 - **/tmp**
 - Contiene archivos temporales.
 - **/var**
 - Contiene información variable, registros, datos de servidores, etc.
 - **/var/log**
 - Contiene los archivos logs o de sucesos en el sistema.
 - Un archivos log es un archivo de texto en el que un proceso o programa irá escribiendo las incidencias que ocurren durante su ejecución, como los errores producidos, los accesos, fecha y hora en las que se produjeron, etc.



3 Gestión de archivos y directorios.

- Un archivo es un conjunto de información relativa a un mismo concepto y que se guarda bajo un nombre que lo identifica.
- En GNU/Linux el nombre de un fichero puede tener hasta 255 caracteres exceptuando el carácter /, que se utiliza para el directorio raíz y para separar directorios.
- Cada fichero se identifica, además de por su nombre, por su número de i-nodo, que es un número entero único que identifica a cada fichero.

◀ Despedida ▶

Email

bienvenidosaez@gmail.com

Instagram

@bienvenidosaez

Youtube

youtube.com/bienvenidosaez