



Nombre: .....

Fecha: / 11 /2010

Grupo: 1  2  3  4 

## PRÁCTICA 15

### SISTEMAS DE DIRECTORIOS EN LINUX Y WINDOWS (I)

Los sistemas de directorios de un sistema operativo tienen como misión fundamental organizar la información que tenemos en el mismo. El sistema de directorios forma parte del sistema de ficheros (FAT, FAT32, NTFS, ext3). Tal y como aprendimos en el Tema 1, los discos duros por lo general tienen la información organizada en sectores, que se repartían en cilindros que pertenecían a discos, que su vez podían tener una o múltiples cabezas. Esta estructura no guarda ninguna relación con la forma en que un usuario la puede observar a través de un sistema operativo (al menos, tanto en Linux como en Windows). El usuario final encuentra la información almacenada en ficheros que se encuentran a su vez en un sistema de directorios y subdirectorios propio del sistema operativo.

La estructura de un sistema de directorios suele ser la de un árbol. Dentro de ese árbol, los directorios se ordenan de forma jerárquica. Cada directorio puede tener uno o más subdirectorios. De este modo, en cada subdirectorio podemos encontrar ficheros con idéntico nombre (aunque parezca un hecho trivial, si no fuese por los directorios esto no sería posible). A partir de ahí, las diferencias entre los sistemas de directorios empiezan a aflorar. En algunos sistemas de directorios está permitido que un mismo fichero esté en varios directorios. En otros, como Windows, cada unidad de almacenamiento del sistema contiene su propio sistema de directorios; en Linux, todos los dispositivos conectados al ordenador se integran en una única jerarquía de directorios. Trataremos de ilustrar esas características y diferencias a lo largo de esta práctica.

1. Inicia Ubuntu. En primer lugar, vamos a instalar las "Guest Additions" (menú Dispositivos de la máquina virtual), que luego nos permitirán montar unidades externas como si fueran discos duros. Reinicia la máquina virtual para que los cambios surtan efecto.

Cuando la máquina haya arrancado de nuevo, teclea el siguiente mandato:

```
$pwd
```

Anota en tu informe el resultado del mandato. Ejecuta también en la terminal "man pwd" y apunta el significado del mandato. Quédate con la idea de que "pwd" son las siglas de "print working directory".

2. Aparte de saber en qué directorio estamos, es importante saber también a qué directorios nos podemos dirigir desde nuestro directorio actual. Ejecuta el mandato que ya conocemos:

```
$ls -l
```

De la lista que se muestra, todas las líneas que comienzan con "d" corresponden a directorios. Esto quiere decir que nos podemos "dirigir" a ellos por medio de "cd nombre\_directorio".

3. Comprueba el significado del mandato "cd" ("change directory") por medio de:

```
$help cd
```

4. Dirígete ahora al directorio "Escritorio" por medio del mandato:

```
$cd Escritorio
```

5. Comprueba la ruta completa del directorio en el que te encuentras ahora por medio del mandato:

```
$pwd
```

6. Vamos a comprobar ahora los contenidos del directorio en que nos encontramos:

```
$ls
```

7. Al parecer la carpeta está vacía. En la estructura de árbol propia del sistema de directorios, podría decirse que hemos llegado a una de las hojas (o de los fines) del árbol (y por tanto no podemos seguir "bajando" en la estructura de directorios). Sin embargo, aún estando "vacía", todas las carpetas en los sistemas basados en Linux contienen dos directorios. Teclea el mandato:

```
$ls -la
```

En primer lugar, debes recordar que la opción "-l" se usa para mostrar directorios y ficheros en "long format" (es decir, con su tipo, permisos, número de enlaces o de contenidos, propietario, grupo, tamaño y fecha; lo puedes recordar en <http://freeengineer.org/learnUNIXin10minutes.html#Listing>). Comprueba en <http://www.tuxfiles.org/linuxhelp/aboutfiles.html> la sección correspondiente a "hidden files and directories". Responde a las siguientes preguntas en tu informe ¿Para qué sirve la opción "-a" del mandato "ls"? ¿Qué caracteriza a los ficheros y directorios ocultos? ¿Cuál suele ser su utilidad? ¿Cómo puedes convertir un fichero o directorio "convencional" en un fichero oculto?

¿Qué directorios ocultos han aparecido en tu carpeta "Escritorio" al teclear "ls -la"? Vamos a comprobar "qué directorios" son. Ejecuta el mandato:

```
$cd .
```

Ejecuta ahora:

```
$pwd
```

¿Dónde estabas antes de hacer "cd ."? ¿Adónde te has dirigido? El directorio "." es una abreviatura que siempre equivale a "mi directorio actual" (y se encuentra en cualquier directorio de Linux). Aunque el directorio "." pueda parecer de poca utilidad por el momento, resultará práctico más adelante para hacer operaciones como "copiar ficheros de un directorio dado a mi directorio actual".

Veamos ahora el significado del directorio "..". Ejecuta el mandato.

```
$cd ..
```

Ejecuta "pwd".

¿Dónde estabas? ¿Adónde te ha dirigido? El directorio ".." es una abreviatura que significa "sube un nivel en el árbol de directorios". También se encuentra en todos los directorios del sistema. Compruébalo por medio de:

```
$ls -la
```

8. Vamos seguir subiendo por el árbol de directorios de nuestro sistema. Repite el mandato:

```
$cd ..
```

9. Ahora deberías encontrarte en el directorio "/home". Lee en <http://www.tuxfiles.org/linuxhelp/linuxdir.html> y apunta en tu informe la finalidad de este directorio. ¿Qué carpetas aparecen en este directorio en tu ordenador? ¿Qué usuarios "propios" (es decir, no para uso interno del sistema) hay en tu ordenador? (Recuerda que lo puedes comprobar por medio de "less /etc/passwd"). Observa que el usuario "root" no dispone de un directorio en la carpeta "home".

10. Seguimos ascendiendo por el árbol de directorios. Ejecuta de nuevo:

```
$cd ..
```

Ejecuta después:

```
$pwd
```

¿En qué directorio te encuentras? Comprueba en <http://www.tuxfiles.org/linuxhelp/linuxdir.html> el significado del directorio "/". Observa que el directorio "/", también conocido como "raíz", no tiene nada que ver con el usuario "root". El directorio "raíz" hace referencia a que el mismo es el origen del árbol de directorios, y por tanto todos los contenidos de tu ordenador (ficheros, discos duros que puedas manejar, unidades USB...) estarán "por debajo" de este directorio. ¿Podemos seguir subiendo por el árbol de directorios? Ejecuta los mandatos:

```
$cd .. ; pwd
```

El carácter ";" concatena los dos mandatos (primero se ejecuta "cd .." y a continuación "pwd").

¿Has ascendido un nivel en el árbol de directorios?

11. Vamos ahora a hacer un breve repaso de lo que nos podemos encontrar en el directorio "raíz" de un sistema Linux (puede que haya alguna pequeña diferencia entre distribuciones, pero en general todas las distribuciones mantienen una estructura muy similar). Ejecuta:

```
$ls -la
```

Observa el propietario de todos los directorios que se encuentran en el directorio raíz. ¿Tendremos permisos de escritura en el directorio raíz (sin usar "sudo")? Ejecuta el mandato:

```
$mkdir hola
```

(Puedes comprobar la función de "mkdir" por medio de "man mkdir"). ¿Qué ha sucedido?

Vamos a ver brevemente qué contiene cada uno de los directorios que "cuelgan" del directorio raíz. Ejecuta de nuevo:

```
$ls -la
```

En <http://www.linux-es.org/node/112> puedes encontrar una descripción detallada de la utilidad de cada uno de ellos. En <http://www.pinoytux.com/linux/linux-file-directory-structure> tienes una representación gráfica que también debe serte útil. Apunta en tu informe su significado. Es importante que retengas algunas ideas. ¿En qué directorio encontrarías los dispositivos de almacenamiento externos (CDROM, USB...)? ¿En qué directorio(s) encontrarías los mandatos que se ejecutan desde la consola? ¿En qué directorio encontrarías los directorios de inicio de los usuarios?

Hay una opción del mandato "ls" que permite mostrar el contenido de cada directorio de forma recursiva (explorando de este modo toda la estructura de directorios de la máquina Linux que se encuentre por debajo de nuestro directorio actual). Ejecuta el mandato (usamos "sudo" para que pueda listar el contenido de todos los directorios):

```
$sudo ls -R
```

Evidentemente la salida del mandato no ha sido de gran ayuda. Sin embargo, sí puede serlo si quieres explorar todos los directorios que cuelgan de uno dado (y no todos los de la máquina, como acabamos de hacer).

Antes de seguir trabajando con directorios, es importante distinguir la diferencia entre rutas absolutas y rutas relativas (problema que ya tratamos a la hora de hablar de los enlaces en html). Desde el directorio raíz dirígete al directorio "home" de alumno:

```
$cd /home/alumno
```

Ahora dirígete a su "Escritorio"; comprueba que para llevar a cabo tal tarea, los dos mandatos siguientes serían correctos:

```
$cd Escritorio
```

```
$pwd
```

```
$cd /home/alumno/Escritorio  
$pwd
```

La primera ruta que hemos especificado (que no empezaba por "/") es una ruta relativa al directorio en que nos encontrábamos cuando hemos ejecutado el mandato "cd Escritorio" (es decir, "/home/alumno"). Por eso, el resultado final es "/home/alumno/Escritorio".

En el segundo caso (que empezaba por "/") la ruta es absoluta. Empezando desde el directorio raíz, "/" nos dirigimos al directorio "home", dentro de éste a "alumno" y después a "Escritorio".

12. Vamos a recuperar ahora las variables de entorno que ya vimos en una práctica anterior. Ejecuta el mandato:

```
$set | less
```

Apunta el valor de la variable \$HOME. Ejecuta ahora:

```
$cd $HOME; pwd
```

¿A qué directorio te has dirigido?

13. Los usuarios también pueden definir sus propias variables de entorno por medio del mandato "export". Ejecuta el mandato:

```
$export mi_escritorio="/home/alumno/Escritorio"  
$set | less
```

¿Encuentras la variable definida entre las variables del entorno? Ejecuta ahora el mandato:

```
$cd $mi_escritorio; pwd
```

Las variables definidas por medio de "export" (o de "set") sólo son válidas para la sesión de la shell que se encuentre abierta. Si quieres hacerlas permanentes (para futuras sesiones) debes introducirlas (con precaución), por ejemplo, en el fichero ".bashrc" (que se encuentra en tu directorio "\$HOME"). El uso de variables puede ser cómodo para crear "atajos" a carpetas. Otro atajo para la variable "\$HOME" de cada uno de los usuarios viene dado por "cd ~". Comprueba el resultado de ejecutar:

```
$cd ~; pwd
```

14. Vamos a ver ahora cómo se puede montar una unidad externa o carpeta en nuestra máquina Linux. Como ya hemos aprendido, en los sistemas Linux todos los directorios de la máquina "cuelgan" o descienden del directorio raíz. Sin embargo, si queremos añadir una unidad externa (de red o simplemente de almacenamiento, como un USB o un CD), e independientemente de que el sistema de ficheros (fat32, ntfs) de la unidad a montar sea distinto del de la máquina origen (posiblemente ext3 ó ext4), podemos hacerlo a través del mandato "mount". Trata de encontrar en <http://www.tuxfiles.org/linuxhelp/mounting.html> el motivo por el cual en Linux es necesario montar dispositivos, mientras que en Windows no lo es.

En primer lugar, vamos a comprobar el manual de ayuda del mandato "mount":

```
$man mount
```

De las distintas opciones de uso que ofrece, nosotros haremos uso de la siguiente estructura:

```
mount -t vfstype device directory
```

Comprueba el significado de la opción "-t".

Para poder montar una carpeta o directorio de nuestra máquina Windows en Linux, haremos uso de la interfaz que nos facilita VirtualBox para ello. Dentro de la opción "Dispositivos" del menú de la máquina virtual, en la opción "Carpetas Compartidas", pulsa el icono  (Agregar carpeta compartida). En el menú "Ruta carpeta" debes identificar la carpeta de tu sistema Windows que pretendes montar. Eso incluye, por supuesto, tu unidad de USB. Elige una carpeta de la máquina Windows (el Escritorio, tu USB, el directorio C:) y asígnale un nombre en el campo "Nombre carpeta" (a ser posible un nombre sin espacios en blanco; apúntalo porque luego lo necesitarás para poder montar la unidad). Pulsa "Aceptar".

Recupera ahora la "shell" Linux. En primer lugar, crea la carpeta donde quieres que se monte el directorio o unidad externa (por ejemplo, en el escritorio de Linux /home/alumno/Escritorio) por medio del mandato:

```
$mkdir Carpeta_Windows
```

Comprueba que la operación se ha completado con éxito ("ls -l").

Ahora ejecuta el mandato:

```
$mount -t vboxsf Nombre_Carpeta Carpeta_Windows
```

(donde Nombre\_Carpeta debes sustituirlo por el nombre que asignaste en el menú de VirtualBox a la unidad o directorio de Windows). ¿Qué ha sucedido? Soluciona el error obtenido.

Cambia de directorio ahora:

```
$cd Carpeta_Windows
```

Comprueba los contenidos de la carpeta:

```
$ls -la
```

¿Se corresponden con los que tienes en la carpeta montada en Windows? El hecho de haber usado "sudo" para montar la carpeta hace que las operaciones de escritura y borrado sobre la misma deban hacerse por medio de "sudo".

De igual modo a como hemos hecho entre la máquina virtual y la máquina Windows en este caso, podemos montar en una máquina Linux cualquier otra partición que se encuentre en su mismo disco duro (éste sería el caso en que no tuvieras Linux en una máquina virtual sino en una partición). Puedes encontrar los pasos a seguir (similares a los que hemos mostrado, aunque cambiando alguna de las opciones) en [http://www.guia-ubuntu.org/index.php?title=Montar\\_particiones](http://www.guia-ubuntu.org/index.php?title=Montar_particiones).

Suponiendo que hayas terminado de operar con la carpeta montada, puedes "desmontar" la misma por medio de:

```
$sudo umount Carpeta_Windows
```

15. Recupera los mandatos que has usado hasta ahora en la sesión por medio de "history". Redirígelos a un fichero de nombre "mandatos\_practica\_15". Al final de la práctica deberás subirlo a tu página de inicio junto al informe de la práctica.

Pasamos ahora a ver la estructura de directorios propia de los sistemas Windows, con la que por lo general estamos más familiarizados.

16. Abre un intérprete de mandatos de Windows ("cmd"). Vamos a repetir algunos de los ejercicios que hemos hecho en Linux y ver lo que sucede. Para conocer el directorio en que te encuentras en Windows puedes usar el mandato:

```
>cd
```

Apunta el nombre del directorio en que te encuentras. Observa que en Windows, las rutas de carpetas y directorios se especifican por medio de la barra "\", en lugar de usar "/".

17. Vamos a ver ahora el mandato que nos permite listar el contenido de un directorio (ls en Linux):

```
>dir
```

Observa las dos primeras entradas del directorio. ¿Las reconoces de Linux?

18. El mandato "dir" nos permite usar algunas opciones adicionales. Puedes comprobarlas por medio de:

```
>help dir
```

Vamos a observar ahora quién es el propietario de cada uno de los ficheros de la carpeta en que nos encontramos. Para ello debes usar:

```
>dir /Q
```

19. Vamos a ascender ahora por el árbol de directorios de nuestra máquina Windows. Ejecuta el mandato:

```
>cd ..
```

20. Comprueba en qué directorio te encuentras por medio de "cd".

```
>cd
```

21. Vamos a comprobar también el valor del directorio ".":

```
>cd .
```

```
>cd
```

22. Subimos un nuevo nivel:

```
>cd ..
```

Ahora deberías encontrarte en la unidad "C:". La unidad "C:" corresponde con una de las particiones de nuestro disco duro. Cada partición del mismo, o cada unidad externa (CDROM, disquetera, USB) tienen una letra propia, y dan lugar a un árbol de directorios propio. Observa y apunta en un informe la diferencia de esta organización con la propia de Linux.

23. Si tienes alguna otra unidad habilitada en tu disco duro (por ejemplo, un USB introducido en F:, o un cd introducido en D:), puedes acceder a ellos por medio de:

```
>F:
```

24. Comprueba que estás en C: (por medio de "cd"). Si es así, ejecuta el mandato:

```
>dir
```

Observa los contenidos del directorio "C:". Trata de identificar algunas de las carpetas más relevantes en el mismo (posiblemente sea más difícil que en Linux). Observa en particular las carpetas "Windows", "Program Files" y "Users". ¿Cuál será su utilidad?

25. En Windows, al igual que en Linux, podemos trabajar con rutas relativas y rutas absolutas. Las rutas absolutas siempre serán de la forma "Unidad:\ruta\archivo", mientras que las relativas serán simplemente "ruta\_relativa\archivo".

Desde el directorio en que te encuentras ahora, puedes explorar todo el árbol de directorios de la unidad C: (que no de todo el sistema, como en Linux). Ejecuta el mandato:

```
>tree
```

La información obtenida quizá no sea de gran utilidad, pero sí que debería servir para comprobar la estructura de árbol del sistema de directorios, y cómo unas carpetas se encuentran dentro de otras.

Ejecuta de nuevo el mandato sobre la carpeta "Users":

```
>tree Users
```

Observa cómo se distribuye la información en dicha carpeta.

26. En Windows también podemos hacer uso de las variables de entorno para dirigirnos de forma más sencilla a directorios. Ejecuta el mandato:

```
>set
```

Apunta el valor de la variable HOMEPATH. Ejecuta ahora:

```
>cd %HOMEPATH%
```

¿En qué directorio te encuentras? También podemos definir nuevas variables de entorno por medio del propio mandato "set":

```
>set escritorio=C:\Users\Alumno\Desktop
```

Comprueba ahora el resultado de ejecutar:

```
>cd %escritorio%
```

(Las variables de entorno que se definen por medio de "set" en el intérprete, al igual que en Linux, tampoco son permanentes; si quieres hacerlas permanentes tienes que hacerlo en Equipo -> Propiedades -> Configuración Avanzada del sistema -> Opciones avanzadas -> Variables de Entorno).

27. Ejecuta el mandato:

```
>DOSKEY /h
```

¿Qué ha sucedido? Redirige la salida del mismo a un fichero "mandatos\_windows\_practica\_15".

28. Comprueba que el fichero contiene la información que necesitas por medio del mandato:

```
>type mandatos_windows_practica_15
```

29. Sube a tu página de inicio el informe de la práctica, así como los ficheros mandatos\_practica\_15 y mandatos\_windows\_practica\_15.