Comandos Linux

Adolfo Sanz De Diego

Octubre 2017

Table of Contents

# Acerca de

## Autor

* **Adolfo Sanz De Diego**
  + Blog: [asanzdiego.blogspot.com.es](http://asanzdiego.blogspot.com.es/)
  + Correo: [asanzdiego@gmail.com](mailto:asanzdiego@gmail.com)
  + GitHub: [github.com/asanzdiego](http://github.com/asanzdiego)
  + Twitter: [twitter.com/asanzdiego](http://twitter.com/asanzdiego)
  + LinkedIn: [in/asanzdiego](http://www.linkedin.com/in/asanzdiego)
  + SlideShare: [slideshare.net/asanzdiego](http://www.slideshare.net/asanzdiego/)

## Licencia

* **Copyright:**
  + [Antonio Sarasa Cabezuelo](mailto:asarasa@gmail.com)

## Fuente

* Las slides y sus fuentes las podéis encontrar en:
  + <https://github.com/asanzdiego/curso-intro-linux-web-sql-2017>

# Comandos básicos

## Ayuda Comandos

* El comando **man nombrecomando** muestra el manual del comando. Una vez dentro para salir hay que pulsar la tecla q.
* También podemos probar con **nombrecomando -h** o **nombrecomando --help** o con **info nombrecomando**.

## Cambio de directorio

* Para cambiar de directorio se usa el comando **cd directorio\_destino**.
* Usamos **cd ..** para ir al directorio superior.
* Usamos **cd sin argumentos** para volver a la carpeta personal.

## Situación actual

* El comando **pwd** imprime la ruta del directorio en el que nos encontramos en este momento.

## Listado

* El comando **ls** muestra los nombres de los ficheros y subdirectorios contenidos en el directorio en el que se está. Sólo se obtienen los nombres de los ficheros, sin ninguna otra información.

## Listas ocultos

* El comando **ls -a** muestra todos los ficheros incluyendo algunos que están ocultos para el usuario (aquellos que comienzan por un punto).
* Observar que el fichero punto . indica el directorio actual y el doble punto .. el directorio padre, que contiene al actual.

## Listado ordenados

* **ls -c** muestra ordenando por día y hora de creación.
* **ls -t** muestra ordenando por día y hora de modificación.
* **ls -r** muestra el directorio y lo ordena en orden inverso.
* **ls subdir** muestra el contenido del subdirectorio subdir.

## Otros listados

* **ls --color** muestra el contenido del directorio coloreado: verde para los ejecutables,azul las carpetas, fucsia las imagenes, rojo los comprimidos, ...
* **ls -l** muestra toda la información de cada fichero incluyendo: protecciones, tamaño y fecha de creación o del último cambio introducido,...

## Combinación de opciones

* Las opciones anteriores pueden combinarse como por ejemplo **ls -cr** que muestra el directorio ordenando inversamente por fechas.

## Filtrado

* Muchos comandos admite los caracteres de sustitución:
  + **\* que representa cualquier conjunto o secuencia de caracteres**
  + y **? que representa cualquier carácter pero sólo uno**.
* Por ejemplo **ls \*.gif** muestra todos los nombres de ficheros que acaben en .gif y **ls file?** muestra todos los ficheros cuyos nombres empiecen por file y tengan un nombre de cinco caracteres.

## Creación de subdirectorios

* El comando **mkdir** permite crear un nuevo subdirectorio. La sintaxis es mkdir subdir1 donde subdir es el nombre del directorio que se va a crear.

## Borrado de subdirectorios

* El comando **rmdir** borra uno o más directorios del sistema siempre que estos subdirectorios estén vacíos. La sintaxis es rmdir subdir1 donde subdir es el nombre del directorio que se va a eliminar.

## Copia de ficheros

* El comando **cp** permite hacer la copia de un fichero. La sintaxis del comando es cp file1 file2 que indica que hace una copia de file1 y le llama file2. Si file2 no existía, lo crea con los mismos atributos de file1, y en caso de existir su contenido es sustituido por el de file1. El fichero file2 estará en el mismo directorio que file1.

## Mover/renombrar ficheros

* El comando **mv** permite mover o renoimbrar un fichero. La sintaxis es mv file1 file2 y realiza la misma función que cp pero eliminando el fichero original. Desde la visión del usuario, se cambia el nombre a file1 por file2.
* Si los nombres que aparecen son de directorios entonces el comando mv namedir1 namedir2 cambia el nombre del subdirectorio namedir1 por namedir2.

## Enlaces

* Un mismo fichero puede estar repetido con más de un nombre y poder acceder a él desde más de un directorio. Esto último se denomina enlaces múltiples a un fichero, y se crean con el comando **ln**: ln file1 file2 Así el fichero file1 tiene dos nombres: file1 y file2.

## Borrado de ficheros

* El comando **rm** elimina uno o más ficheros de un directorio en el cual tengamos permiso de escritura. La sintaxis es rm file1 file2

## Permisos

* Los permisos de cada fichero se pueden ver con el comando ls -l.
* Estos permisos son:
  + **r**: de lectura (o listar en directorios)
  + **w**: de escritura (o crear y borrar ficheros en directorios)
  + **x**: de ejecución (o buscar y utilizar ficheros en directorios)

## Cambiar permisos

* Para cambiar los permisos de un fichero se emplea el comando **chmod [quien] operacion permiso file** donde:
  + quien: indica a quien afecta el permiso que se desea cambiar:
    - **u**: para el usuario propietario del archivo,
    - **g**: para el grupo del usuario propietario del archivo,
    - **o**: para los otros usuarios
    - **a**: para todos los anteriores. (valor por defecto).
  + oper: indica si el permiso se da usando un + o se quita usando un -.

## Cambiar dueño

* El comando **chown** se emplea para cambiar de propietario a un determinado conjunto de ficheros: chown newowner file1 file2 ...
* Sólo lo puede emplear el actual propietario de los mismos.
* Los nombres de propietario se encuentran almacenados en el fichero **/etc/passwd**.

## Cambio de grupo

* El comando **chgrp** se emplea para cambiar el grupo al que pertenece un fichero : chgrp newgroup file1 file2...
* Los grupos de usuarios están almacenados en el fichero **/etc/group**.

## Visualización con cat

* El comando **cat filename** permite visualizar el contenido de uno o más ficheros de forma no formateada, y copiar uno o más ficheros como apéndice de otro ya existente.

## Concatenar ficheros

* **cat file1 file2 >file3**: El contenido de los ficheros file1 y file2 es almacenado en file3.
* **cat file1 file2 >>file3**: el contenido de file1 y file2 es añadido al final de file3.
* **cat >file1**: Acepta lo que se introduce por el teclado y lo almacena en file1 (se crea file1). Para terminar se emplea d

## Visualización con more

* El comando **more filename** permite visualizar el contenido de un fichero pantalla pantalla.
* Para pasar de pantalla se utiliza la barra espaciadora o la tecla enter.
* Para salir se pulsa d o q.

## Visualización con less

* El comando **less filename** permite visualizar el contenido de un fichero pantalla pantalla.
* Para pasar de pantalla se utilizan las flechas arriba y abajo.
* Para salir se pulsa d o q.

## Búsqueda en ficheros

* El comando **grep 'conjuntocaracteres' file1 file2 file3** busca una palabra, clave o frase en un conjunto de directorios, indicando en cuáles de ellos la ha encontrado.
* Se pueden utilizar expresiones regulares de la forma: grep [-opcion] expresión\_regular [referencia...]

## Opciones grep

* **c**: escribe el número de las líneas que satisface la condición.
* **i**: no se distinguen mayúsculas y minúsculas.
* **l**: escribe los nombres de los ficheros que contienen líneas buscadas.
* **n**: cada línea es precedida por su número en el fichero.

## Mas opciones grep

* **s**: no se vuelcan los mensajes que indican que un fichero no se puede abrir.
* **v**: se muestran sólo las líneas que no satisfacen el criterio de selección.

## Ejemplos grep

* **grep ‘ˆd’ text** recupera las líneas que comienzan por d.
* **grep ‘ˆ[ˆd]’ text** recupera las líneas que no comienzan por d.
* **grep -v ‘ˆC’ file1 > file2** quita las líneas de file1 que comienzan por C y lo copia en file2.

## Redirecciones

* Se puede redirigir la salida de un comando usando los operadores:
  + (**>**) redirige la salida estándar hacia el fichero indicado y en caso de no existir se crea.
  + (**<**) redirige la entrada estándar desde un determinado fichero
  + (**>>**) redirige la salida estándar hacia otro fichero, pero añadiendo dicha salida al final de ese fichero, sin sobreescribir el contenido original.

## Ejemplos Redirecciones:

* **date >> archivo** el fichero archivo contendrá información sobre todas las veces que hemos entrado en el sistema.
* **cat file1 file2 > file3** añade al fichero file2 al final de file1 y al conjunto lo llama file3

## Tuberías

* Una tubería (|) permite comunicar la salida estándar de un comando con la entrada estándar de otro.
* Por ejemplo **ls | mail juan** envía a juan una lista de los ficheros del sistema.
* Con el operador de tubería se pueden empalmar tantos comandos como se desee.

## Ejecución segundo plano

* El carácter **& al final** permite realizar una ejecución en segundo plano recuperando inmediatamente el control del terminal. Para ello se añade el carácter & al final del comando de ejecución.

## Parar un proceso

* Para detener la ejecución de un proceso se puede utilizar el comando **kill númerodeproceso**.

## Continuar en segundo plano

* Cuando se sale del sistema si hay algún proceso ejecutándose en segundo plano se para salvo que se use el comando **nohup nombreprograma**. En este caso todas las salidas del programa se dirigen a un fichero llamado nohup.out.

## Cambiar la prioridad

* Para darle al programa la prioridad mínima habría que invocarlo con **nice -19 nombreprograma &**.
* Para darle al programa la prioridad máxima habría que invocarlo con **nice --20 nombreprograma &**.

## Listado procesos

* El comando **top** muestra una lista de los procesos que se están ejecutando.

## Agrupación de ficheros

* El comando **tar -cvf nombre\_archivo.tar fichero1 fichero2 ...** agrupa varios ficheros en uno solo “archivo” tar

## Compresión de ficheros

* El comando **gzip fichero** comprime fichero (que es borrado) y se crea un fichero con nombre fichero.gz.

## Agrupación y compresión

* El comando **tar -czfv archivo.tar.gz ficheros** empaqueta y comprime ficheros.

## Descomprimir ficheros

* El comando **tar -xzvf archivo.tar.gz** descomprime archivo.tar.gz

# Otros comandos

## Espacio carpetas

* El comando **du** permite conocer el espacio en bloques ocupado en el disco por un determinado directorio y todos los subdirectorios que cuelgan de él.
* Para obtener la información en bytes se debe emplear la opción -h: **du -h**

## Espacio particiones

* El comando **df** informa del espacio usado por las particiones del sistema que se encuentren montadas.
* Para obtener la información en bytes se debe emplear la opción -h: **df -h**

## Fecha

* El comando **date** muestra el día y la hora.

## Calendario

* El comando **cal** muestra el calendario. Tiene diversas opciones. Por ejemplo cal 1945 mostraría el calendario del año 1945.

## Usuarios conectados

* El comando **who** muestralos usuarios están conectados al ordenador en ese momento, en qué terminal están y desde qué hora.

## Quien soy

* El comando **whoami** te indica quien eres.

## Tiempo sin apagar

* El comando **uptime** muestra el tiempo que lleva encendido nuestro ordenador.

## Nombre de la máquina

* El comando **hostname** muestra el nombre de la máquina.

## Hardware

* El comando **lshw** muestra todas las características del hardware.

## Dispositivos PCI

* El comando **lspci** muestra los diferentes dispositivos PCI.

## Dispositivos USB

* El comando **lsusb** muestra los dispositivos tengo en los bus USB

## Comandos usados

* El comando **history** muestra los comandos usados por el usuario en orden cronológico.

## Permisos por defecto

* El comando **umask** muestra los permisos con los que el usuario creara sus archivos por defecto

## Versión kernel

* El comando **uname -r** muestra la version del kernel

## Cerrar programas

* El comando **xkill** permite cerrar un programa bloqueado.

## Grupos

* El comando **groups** muestra los grupos del sistema a los que pertenece un usuario.

## Limpiar terminal

* El comando **clear** limpia el terminal.

## Primeras líneas

* El comando **head** muestras las primeras líneas de un fichero

## Ultimas líneas

* El comando **tail** muestras las últimas líneas de un fichero