
Comandos de Linux

Para ejecutar los comandos de Linux tenemos que abrir un terminal de trabajo. Si tenemos varias aplicaciones abiertas podemos pasar de una a otra con CTRL ALT.

Para ver la ayuda del comando utilizado podemos escribir:

man comando

comando -help en algunos casos debemos poner doble guión.

El formato para escribir los comandos es:

comando [-opciones]

Veamos algunos de los comandos relacionados con los procesos:

- **ps** (process status), nos da la información sobre los procesos. Por defecto sale el proceso bash (que es el Shell) y el comando ejecutado. Y la información referente a

```
admin@ubuntu:~$ ps
  PID TTY          TIME CMD
 2165 pts/0        00:00:00 bash
 2434 pts/0        00:00:00 _ps
```

PID, es el identificador de proceso.

TTY, es el terminal asociado del que lee y al que escribe. Si no tiene asignado pone una interrogación. Y los enumera del 0 en adelante.

TIME, el tiempo de ejecución.

CMD, nombre del proceso.

A este comando le podemos añadir parámetros como:

- ✓ **f**, nos añade la siguiente información:

```
admin@ubuntu:~$ ps -f
  UID          PID  PPID  C STIME TTY          TIME CMD
admin     2165    2156  0 12:36 pts/0        00:00:00 bash
admin     2607    2165  4 12:53 pts/0        00:00:00 ps -f
```

UID, que es el nombre del usuario.

PPID, número de identificador del proceso padre.

C, porcentaje de recursos utilizados.

STIME, hora en que comenzó el proceso.

SZ, tamaño virtual de la imagen del proceso.

RSS, tamaño de la parte residente en memoria en kilobytes.

PSR, procesador que el proceso tiene actualmente asignado.

- ✓ **u**, muestra los procesos del usuario y de todas las terminales que estén abiertas.
- ✓ **A**, muestra los procesos de todas las terminales.
- ✓ **AF**, muestra todos los procesos activos de cualquier usuario y con mucha más información.
- ✓ **e**, muestra la información de todos los procesos pero la elemental (pid, tty, time, cmd). Si le añadimos la letra o al parámetro podemos listar los campos a visualizar separados por comas incluso podemos indicar el nombre de cualquier campo. El campo nlwp nos indica las hebras de un proceso.
- ✓ **L**, vemos las hebras de cada proceso.

Comandos de Linux

- ✓ **I**, vemos la prioridad del proceso. Que es Nice, en la columna NI muestra el valor de la prioridad va -20 que es la más alta a 19 que es la más baja, por defecto se ejecuta con 0 el proceso.

Los estados de un proceso son:

- ❖ **S** (sleep), el proceso está dormido espera que se le asigne la cpu, esperando algún evento.
- ❖ **R** (Running), está siendo ejecutado en el procesador.
- ❖ **T** (Stopped), el proceso está detenido por alguna señal.
- ❖ **Z** (Zombie), el proceso padre del que depende aún no ha terminado por lo que no ha sido liberado del todo. Sigue estando en la tabla de procesos.

Otras letras que pueden añadirse son:

- ❖ **l**, multihebra.
- ❖ **+**, proceso en primer plano.
- ❖ **<**, proceso con prioridad alta.
- ❖ **N**, proceso con prioridad baja.
- **ps tree**, visualiza los procesos en forma de árbol. Si le añadimos **-p** nos indica el número de PID.
- **top**, muestra los recursos utilizados en tu sistema, va cambiando la información en tiempo de ejecución, cada 3 segundos se actualiza. Si pulsamos la letra **h** nos muestra la ayuda. Para salir pulsaremos la letra **q**. Para matar un proceso la letra **k** y para modificar la prioridad de un proceso será la letra **r**, nos pedirá alguna información más cuando queramos hacer algo sobre los procesos para saber cuál es.
- **kill**, para matar un proceso. Indicaremos el número de proceso o procesos separados por un espacio en blanco.

Veamos algunos parámetros:

- ✓ **I**, muestra todas las señales que podemos mandarle a un proceso y esta pueden ser utilizadas con el número o con el nombre.
- ✓ **1** o **SIGHUP**, cierra la terminal asociada a uno o más procesos.
- ✓ **2**, pausa un proceso.
- ✓ **9** o **SIGKILL**, el proceso debe finalizar de forma inmediata, sin tiempo de liberar los recursos que están utilizando.
- ✓ **15** o **SIGTERM**, termina el proceso normalmente, dándole tiempo a liberar los recursos utilizados. En alguna ocasión pueden ser ignorados por un proceso y no terminan.
- ✓ **18**, continua la ejecución de un proceso.
- ✓ **19 y 20**, detienen momentáneamente la ejecución de un proceso.
- kill -9 2543**
- kill -SIGKILL 2543**
- **pkill**, matamos un proceso con parte de su nombre, esto puede hacer que matemos algún proceso de nombre parecido.
- **killall**, mata un proceso por su nombre.
- **xkill**, mata un proceso que tenga un entrono gráfico. Al ejecutar el comando debemos ir a la ventana del programa abierto y pulsar la tecla **1**, que es cuando lo mata.
- **pgrep**, nos da el número de PID del nombre del proceso indicado.

Comandos de Linux

pgrep nombre

- **pidof**, con el podemos ver el número de PID. Su sintaxis es comando y el nombre del proceso.

Pidof firefox

- **nice ó renice**, lo utilizamos para cambiar la prioridad de un proceso. Por defecto toma el valor 0 y las prioridades van del -20 la prioridad más alta a la 19 que es la más baja. Solo root puede dar prioridades negativas. La segunda pueda alterar la prioridad en tiempo de ejecución.

renice número pid

- **htop**, es una mejora del top. Por defecto no está instalado (sudo apt-get install htop).
- **atop**, es un monitor de procesos en tiempo real. Si no está instalado sudo apt-get install atop.
- **nohup o &**, ejecuta los procesos en un segundo plano.
- **jps**, nos muestra los procesos en java que se están ejecutando en ese momento.

v, salida de argumentos que se le pasan a la jvm.

l, salida el nombre de paquete completo para la clase principal de la aplicación o el nombre de la ruta completa al archivo JAR de la aplicación.

q, suprime la salida del nombre de la clase, el nombre del archivo JAR, y los argumentos se pasa al método principal, produciendo sólo una lista de identificadores locales VM.

m, salida de los argumentos que se pasan al método principal. La salida puede ser nulo para las JVM incrustados.

- **jstack**, vemos los hilos y demonios que se están ejecutando en ese proceso que indicamos. Debemos indicar el pid del proceso

jstack pid

- **free**, nos permite visualizar la memoria del equipo. Si ponemos como parámetro la letra **b**, **g**, **k**, **m** nos daría la memoria en esas unidades (bytes, gigabyte, kilobyte y megabyte) por defecto la da en kilobyte. Si ponemos **t** nos da el total de memoria.

free -m -t

- **vmstat**, nos da información de la memoria virtual del equipo. Con el parámetro **a** da la activa e inactiva y con **f** el número de forks (procesos)en el sistema.

vmstat -a

- **uptime**, el tiempo de conexión del usuario.
- **Grep o egrep**, con este comando podemos hacer búsqueda de información. Podemos indicar el nombre del proceso o programa o su pid. Como es un filtro debe ir precedido del comando que deseamos ejecutar y a continuación el con un pipe.

grep nombre o grep pid

ej: ps -AF | grep 1234

grep 'cadena' fichero formato del comando

Busca coincidencias de todas las cadenas indicadas, no tienen que ser completas las cadenas.

grep 'cadena1|cadena2|cadena3' fichero

grep [0-9] Indicamos un rango de caracteres.

grep[0,1,2] Solo de los caracteres indicados

Comandos de Linux

grep **[[:digit]]** que tiene un dígito.
grep **[[:lower]]** está en minúscula.
grep **[[:upper]]** está en minúscula.
^ que empiece por ^fire
\$ que termine por \$fox
? para sustituir un solo carácter.
* para sustituir una cadena de caracteres.
-i, no diferencia entre mayúsculas y minúsculas.
-w búsqueda en palabras.
-v excluye la cadena
-n indica en que línea está la coincidencia.
-c nos dice cuántas veces aparece repetida.

Podemos escribir más de un pipe para el filtro.

En algunas instrucciones vamos a tener que tener permiso de administrador. Para ello ejecutamos la instrucción correspondiente pero irá precedida de la palabra **sudo**. Nos pedirá la contraseña para ejecutar lo pedido.

Ctrl C, aborta la ejecución de una pantalla abierta.

- **clear**, limpiar pantalla en una terminal.
- **gedit**, editor de texto para crear nuestros pequeños documentos.
- **Monitor del sistema**, vemos de forma gráfica la información de los procesos.
- **ls**, para ver los ficheros y carpetas, también admite **dir**. El parámetro **l** nos muestra los permisos de los ficheros o carpetas. Además por colores vemos que es cada cosa. Si está en blanco es un fichero, en verde es un ejecutable, en rojo es un fichero comprimido y en azul es una carpeta.

El primer carácter que nos muestra **ls** nos indica que es: **d** es una carpeta, **-** es un fichero, **l** es un link y **c** es una terminal (tty).

- **Chmod**, da permisos a los ficheros de lectura, escritura y ejecución. Para el usuario, el grupo y otros resto. Están en binario con 3 bits. Los permisos totales serían 777.

sudo chmod [u|g|o] [=|+|-] [r|w|x] fichero o carpeta

sudo chmod +x fichero da el permiso de ejecución a todos.

sudo chmod 744 fichero damos permisos de lectura, escritura y ejecución la usuario, lectura al grupo y lectura a otros

- **mv**, para mover o renombrar un fichero.
- **rm**, borra un fichero o directorio.
- **cp**, para copiar ficheros.
- **cat**, visualiza el contenido de un fichero.
- **touch**, crea un archivo vacío. Si existe actualiza la hora.

http://www.tutorialspoint.com/unix_commands/

Comandos de Linux

INSTALAR UN PROGRAMA EN LINUX

- Para instalar un programa podemos hacerlos desde el centro de software de Ubuntu que es una forma mucho más cómoda. Si queremos instalarlo desde una terminal variara en función de cómo sea el paquete (<http://www.ubuntu-guia.com>). Veamos algunos casos:
 - `sudo apt-get install paquete [paquete...]` Instalar paquetes.
 - `sudo apt-get remove paquete[paquete...]` Desinstalar paquetes
 - `sudo apt-get remove --purge paquete[paquete...]` Desinstalar paquetes incluyendo los archivos de configuración.
 - `sudo apt-get update` Actualiza la lista de paquetes disponibles.
 - `sudo apt-get upgrade` Actualiza el sistema con los paquetes.
 - `sudo apt-get help` Es la ayuda del comando
 - `sudo dpkg -i nombrepaquete.deb` Instala el paquete deb. Programas para ser instalados.
 - `sudo dpkg -r nombrepaquete` Desinstala el paquete deb.
 - `sudo alien -i nombrepaquete.rpm` Convierte un paquete rpm a deb. Son archivos de Red-Hat, Mandriva
 - `sudo ./nombre.bin` Son ficheros en binario que son ejecutables. Tenemos que tener permisos de ejecución.
 - `bash nombre.sh` Instala un paquete que es un script.
 - `sudo ./ nombre.sh` Instala un paquete que es un script.
 - `sh nombre.run` Son asistentes gráficos que ayudan a la instalación.
 - `sudo ./ nombre.run`

CONFIGURAR EL TECLADO

Vamos a configurar el teclado en castellano, debemos ejecutar las siguientes sentencias:

```
sudo apt-get install language-pack-es
sudo apt-get install language-pack-es-base
sudo apt-get install language-support-es
```

reconfigurar

```
sudo dpkg-reconfigure locales
```

diccionarios

```
sudo apt-get install aspell-es
sudo apt-get install myspell-es
```

Después ir al entorno gráfico y buscar Language Support, seleccionar el castellano y ponerlo en la primera opción. Si con todo esto no está el teclado bien configurado buscar Distribución del teclado en el entorno gráfico y elegir un teclado de 105 teclas.