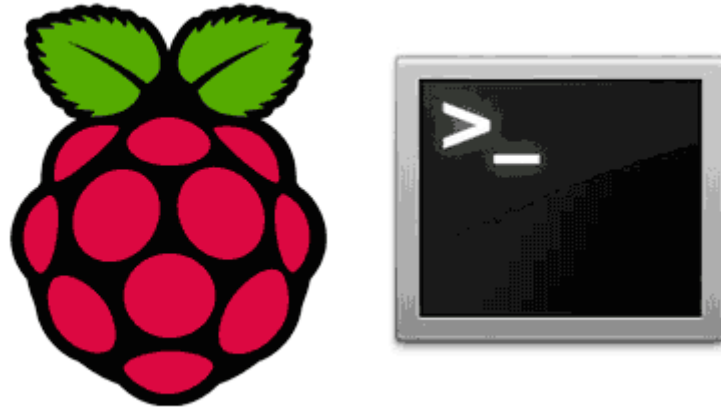


APRENDER A USAR LA CONSOLA DE COMANDOS DE RASPBERRY PI



En esta entrada vamos a aprender a **usar la consola de comandos de Raspberry Pi**, una herramienta muy útil y que debemos aprender a usar si queremos profundizar en el uso de cualquier PC con Linux.

La consola de comandos, o CLI, puede ser intimidante para los nuevos usuarios. De hecho, es una de las ideas que la mayoría de la gente tiene como referencia y más reticencias generan a la hora de pasar a un sistema como Linux.

El motivo por el que se sigue usando la consola de comandos es sencilla. **Es un método muy rápido y potente de ejecutar acciones en un ordenador.** Imaginar la velocidad de realizar cualquier acción mediante alguno de los muchos comandos disponibles, cada uno de los cuales puede tener varios parámetros, y que podemos unir en innumerables combinaciones.

Por supuesto **no consiste en saberse de memoria cientos de comandos**. Para nada. Es más, da igual cuantos años lleves usando Linux, seguramente siempre vas a poder encontrar uno nuevo. Además, hay mucha documentación al respecto, la propia consola de comandos tiene herramientas para darte ayudas, y es incluso normal tener "chuletas" o recopilatorios de comandos útiles.

Por tanto, **hay que perderle el miedo a la consola de comandos**. Además, de verdad no es tan difícil como parece. Es cuestión de empezar a usarla y poco a poco, casi sin darte cuenta, te sorprenderás de lo cómodo que encuentras con ella.

Así que para empezar a perderle el miedo, vamos a empezar viendo que es eso de la consola de comandos o CLI.

¿QUÉ ES LA CONSOLA DE COMANDOS?

El interfaz de línea de comandos (CLI, command-line interface), o simplemente consola de comandos, es **un sistema para controlar un ordenador o programa a través de instrucciones** introducidas como una línea de texto simple.

La consola de comandos es una herramienta habitual en los sistemas operativos basados en Unix, como Linux. En general, aparece como una interface sencilla, en la que **el usuario escribe órdenes y recibe información del sistema a través de mensajes de texto**.

Con el paso de los años, en una época dominada por los interfaces gráficos (GUI, graphical user interface), la consola de comandos puede parecer arcaica. Sin embargo, sigue siendo una herramienta potente, rápida, y muy habitual en los sistemas Unix.

s CLI no son exclusivos de Unix. Windows tradicionalmente ha usado cmd.exe como consola de comandos. En las últimas versiones, Microsoft ha potenciado su CLI introduciendo PowerShell, mucho más potente.

La mayoría de CLI no sólo permiten introducir órdenes de forma manual, si no que **permiten ejecutar comandos de forma automatizada desde un archivo de texto** que se denomina script (equivalente a los batch en Windows).

Adicionalmente, otro de los motivos por los que los CLI siguen estando en pleno uso es **el control remoto de un ordenador**. Emplear un CLI remoto (como ssh, que veremos en su momento) permite ejecutar acciones con mucha rapidez y rendimiento, dado que sólo enviamos texto y no tenemos que enviar/recibir la señal de video que requiere un GUI.

¿CONSOLA DE COMANDOS, TERMINAL O SHELL?

Es frecuente denominar al CLI como consola de comandos, emulador de terminal, o shell. Aunque son conceptos distintos, **en la práctica es frecuente que se usen (incorrectamente) como sinónimos** ya que están muy relacionados.

Por clarificar los términos, **un shell es cualquier interface que permite controlar un sistema operativo**. Por tanto, el CLI es un shell, pero también hay shell gráficos.

El término terminal proviene de los primeros ordenadores, en los que controlabas al ordenador desde un "dispositivo" que básicamente era una pantalla y un teclado. **El emulador de terminal es el programa que nos permite ejecutar un CLI en un sistema gráfico** y, como su propio nombre indica, "emula" estos primeros terminales.

Por último, **el CLI en sí es el propio intérprete de instrucciones**, el programa que recibe los comandos y ejecuta las acciones pertinentes en el SO.

Por resumir. En un ordenador con un entorno de ventanas, el sistema de ventanas con la captura de ratón y teclado es un shell. Cuando abrimos una consola de comandos, eso también es un shell.

El programa que abrimos es el emulador de terminal. Y lo que hay dentro de este, es el intérprete de comandos o CLI.

Por ilustrarlo, a modo de ejemplo, cuando escribimos un comando y pulsamos Enter, es el CLI el que procesa los datos. Pero, por ejemplo, la función de copiar y pegar un comando, o mostrar la salida del CLI, son propias del emulador de terminal.

¿QUÉ ES ESO DE BASH?

En Linux **existe más de un intérprete de líneas de comandos**. De hecho, hay muchas. Algunos ejemplos son Bash, Dash, KornShell (ksh), Tcsh, Fish o Z Shell (zsh), entre otros, cada uno con sus ventajas y desventajas (y fans y detractores).

GNU Bash es el CLI implementado de forma mayoritaria en las distribuciones de Linux. Ejemplos de distribuciones que emplean Bash como CLI por defecto son Debian, openSUSE, Ubuntu, Fedora, Slackware, Mandriva.

Como sabemos, Raspbian está basado en Debian. Por lo que Bash es el CLI incluido por defecto en Raspberry Pi.

Bash, Bourne-Again SHell, fue lanzado en 1989. Es el predecesor de SH, The Bourne Shell, que es uno de los CLI originales de Unix. Los scripts de Bash son compatibles con SH, pero Bash añadió funciones adicionales como autocompletado o histórico de comandos.

Por si alguno le pica la curiosidad, para comprobar que CLI estamos usando actualmente podemos usar este comando

1	<code>echo \$0</code>
---	-----------------------

Y para listar todos los CLI disponibles en un sistema

1	<code>cat /etc/shells</code>
---	------------------------------

```
pi@raspberrypi:~ $ echo $0
bash
pi@raspberrypi:~ $ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/dash
/bin/bash
/bin/rbash
pi@raspberrypi:~ $
```

¿QUÉ ES ESO DE LXTERMINAL?

En un sistema Linux, igual que existe más de un CLI disponible, **existe más de un emulador de terminal**. Algunos ejemplos de terminales son LXTerminal, Gnome Terminal, xterm, UXTerm, Lonsolem, entre otros muchos.

LXTerminal es el emulador de terminal incluido en Raspbian por defecto, ya que es el terminal por defecto de LXDE Desktop, la interface gráfica de ventanas empleado por defecto en Raspberry Pi.

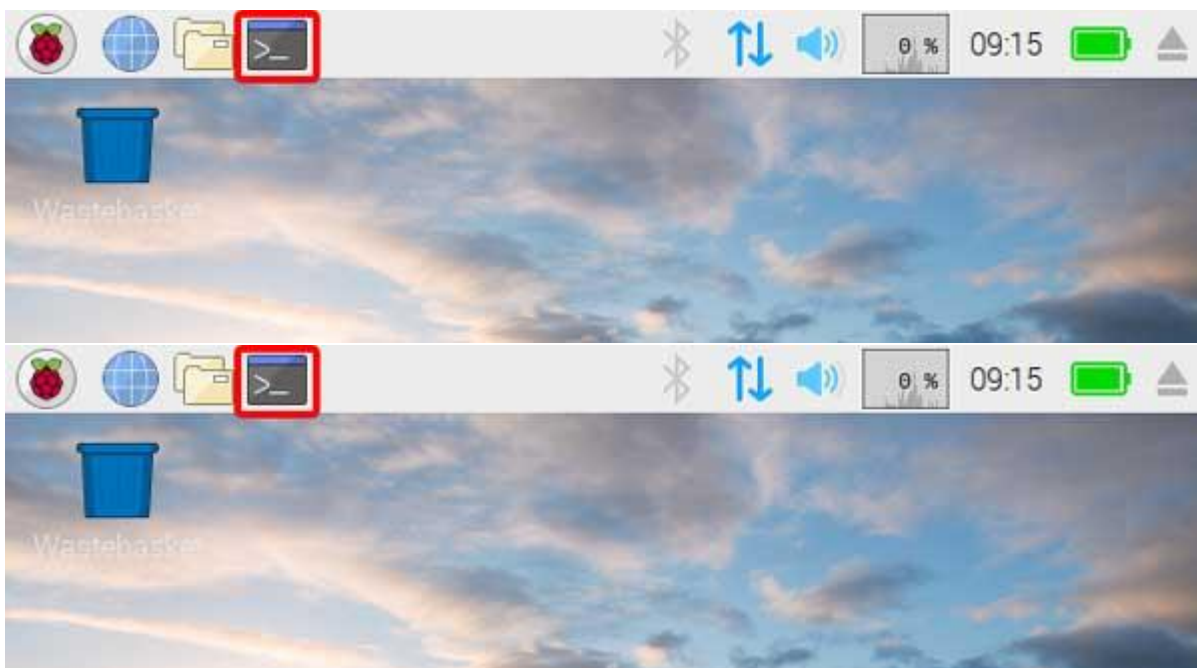
Si por curiosidad, alguno quiere comprobar que emuladores de terminal tiene disponibles en su sistema puede escribir,

```
1 sudo update-alternatives --config x-terminal-emulator
pi@raspberrypi:~ $ sudo update-alternatives --config x-terminal-emulator
There is only one alternative in link group x-terminal-emulator (providing /usr/
bin/x-terminal-emulator): /usr/bin/lxterminal
Nothing to configure.
pi@raspberrypi:~ $
```

Que como vemos, por defecto, en el caso de Raspbian sólo aparece LXTerminal.

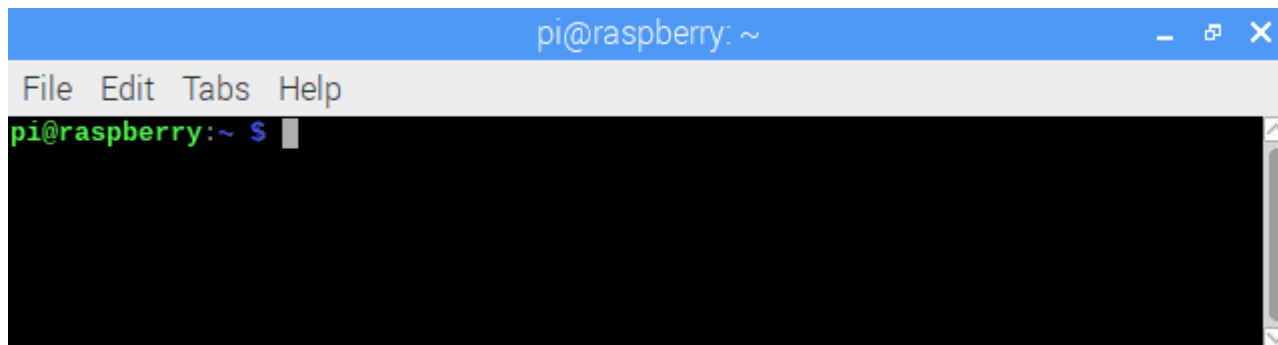
COMO LANZAR LA CONSOLA DE COMANDOS EN RASPBERRY PI

Lanzar Bash, la consola de comandos por defecto en Raspberry, es muy sencillo. **Tenemos un botón disponible en la barra superior** que lanza Bash directamente.



En otras distribuciones, como Ubuntu, el CLI está asociado al atajo de teclados Control+Alt+T. En Raspbian no, por lo que si queremos el atajo tendremos que configurarlo nosotros mismos.

Al inicio de cada línea en la que el usuario puede introducir un comando, veremos una **cabecera con información proporcionada al usuario**.



1	pi@raspberrypi:~\$
---	--------------------

El significado de estos datos es:

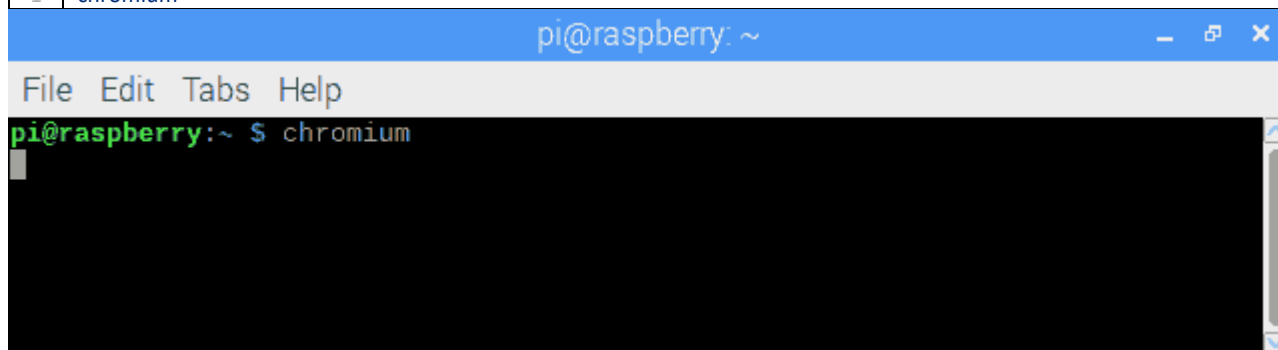
- pi: indica el usuario conectado a la terminal
- @raspberrypi: indica el nombre de la máquina a la cual estamos conectados
- ~: indica la ruta en la cual nos encontramos, en este caso ruta de inicio predeterminada
- \$: indicador para comenzar a escribir comandos (será # si estamos loggeados como usuario root)

EJECUTAR COMANDOS EN LA CONSOLA DE COMANDOS

Ejecutar una instrucción es tan sencillo **como escribir el comando y pulsar la tecla enter**. Si el comando es correcto, el CLI realizará las acciones oportunas y el emulador de terminal mostrará el resultado en la pantalla.

Por ejemplo, para lanzar Chromium, el navegador instalado por defecto en Raspbian, simplemente escribimos el siguiente comando

1	chromium
---	----------



Muchos de los comandos aceptan parámetros adicionales que modifican el comportamiento del comando. Así, chromium admite pasar una URL como parámetro, y al abrirse cargará directamente la URL indicada. Por ejemplo, para cargar directamente la página de Google escribimos.

1	chromium www.google.com
---	---

Si algún comando ejecuta un proceso, y queremos cancelarlo, podemos pulsar el atajo de teclado

1	control + C
---	-------------

Si queremos limpiar la pantalla, quitando todos los comandos escritos con anterioridad, ejecutamos el comando

1	clear
---	-------

AUTOCOMPLETADO CON TABULACIÓN

Uno de los grandes "trucos" a la hora de emplear Bash y que nos ahorrará una enorme cantidad de tiempo es la **función de autocompletado con la tecla tabulación**.

Si en cualquier momento mientras escribimos un comando, un nombre de archivo, o algunos argumentos, pulsamos la tecla tabulación **Bash sugiere la opción que empieza por lo que hemos escrito**. En caso de haber más de una ocurrencia, se recorren cíclicamente las opciones disponibles.

Si existen muchas ocurrencias, al pulsar tabulación no aparecerá nada. **Al pulsarla una segunda vez nos mostrará un listado con los comandos disponibles**. Podemos añadir algunas letras más y pulsar tabulación para terminar el comando.

Así, siguiendo con el ejemplo de chromium, si pulsamos escribimos 'ch' y pulsamos dos veces tabulación nos mostrará un listado con múltiples opciones disponibles.

```
pi@raspberrypi:~$ ch
chage      chardetect3  chfn      chown      chrt
chardet    chatr       chgpasswd chpasswd   chsh
chardet3   chcon      chgrp     chromium   chvt
chardetect chcpu      chmod     chroot
pi@raspberrypi:~$ chrom
```

Si completamos hasta 'chrom' y pulsamos tabulación, Bash autocompleta a chromium

a función de autocompletar es especialmente útil al tratar con rutas o nombres de ficheros.

HISTORIAL DE COMANDOS

Otra función que nos ahorrará mucho tiempo y que es una de las mejoras que introdujo Bash respecto a su antecesor SH, es el **histórico de comandos**.

En cualquier momento podemos **recuperar un comando escrito con anterioridad pulsando la tecla "arriba"** del teclado de flechas. Pulsarlo sucesivamente nos ira trayendo el comando anterior, hasta el inicio de la sesión del terminal. Si nos hemos pasado, podemos pulsar la tecla "abajo" para pasar al comando posterior.

UN POQUITO DE AYUDA

Bash dispone de ciertos comandos que nos permiten obtener ayuda sobre los comandos disponibles.

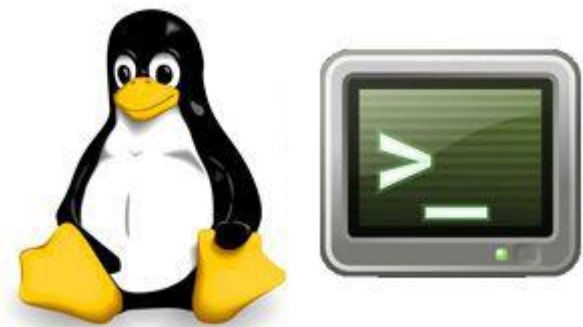
```
pi@raspberrypi:~$ whatis chromium
chromium (1) - the web browser from Google
pi@raspberrypi:~$ whereis chromium
chromium: /usr/bin/chromium /usr/lib/chromium /etc/chromium.d /usr/share/chromium
m /usr/share/man/man1/chromium.1.gz
pi@raspberrypi:~$
```

Algunos de estos comandos son

1	# Mostrar un resumen de la funcion del comando xxxxx
2	whatis xxxxx
3	
4	# Mostrar la localización más probable para el programa xxxxxx
5	whereis xxxxx
6	
7	# Listar las páginas de manual que tratan acerca del comando xxxxx
8	apropos xxxxx
9	
10	# Mostrar el manual de uso o configuración del programa xxxxx
11	man xxxxx

12	
13	# Mostrar las páginas de manual que contengan la palabra xxxxx
14	man -k xxxxx

LISTADO DE COMANDOS LINUX MÁS ÚTILES



Os presento una recopilación de comandos Linux más útiles. Pese a que las interfaces gráficas en Linux son cada día más funcionales y vistosas, si de verdad queréis exprimir al máximo el potencial de este sistema operativo es inevitable el uso de la consola de comandos.

El siguiente listado muestra un resumen organizado por categorías con los comandos Linux que os resultarán más útiles si queréis aprender e iniciaros como usuarios o administradores de sistemas Linux.

CONTROL DE PROCESOS Y TRABAJOS

EJECUTAR COMANDOS Y APLICACIONES

1	#ejecutar aplicacion en carpeta actual
2	<code>./aplicacion</code>
3	
4	#ejecutar comando
5	<code>comando</code>
6	
7	#iniciar proceso en background
8	<code>comando &</code>
9	
10	#cancelar comando
11	<code>control + c</code>
12	
13	#poner comando en background
14	<code>control + z</code>
15	
16	#recuperar proceso puesto en background
17	<code>bg</code>
18	
19	#poner trabajo en primer plano
20	<code>fg jobID</code>
21	
22	#iniciar comando como root
23	<code>sudo comando</code>
24	
25	#iniciar ejecutable con interface grafica como root
26	<code>sudo ejecutable</code>

27	
28	#gestor de sesiones multiples en terminal
29	screen -S nombre_sesion

LISTAR PROCESOS Y TAREAS

1	#mostrar procesos en background con jobId y PID
2	jobs -l
3	
4	#mostrar procesos
5	ps
6	
7	#mostrar procesos activos
8	ps -eafw
9	
10	#mostrar árbol de sistema de procesos.
11	pstree
12	
13	#mostrar procesos ordenados por consumo de memoria
14	ps aux sort -k 5

FINALIZAR PROCESOS

1	#finalizar proceso para recargar configuracion
2	kill -1 ID_Processo
3	
4	#forzar cierre de proceso por PID
5	kill -9 PID
6	
7	#finalizar proceso por nombre
8	killall -9 nombre

ALIAS DE COMANDOS

1	#configurar un alias temporal para comando
2	alias cmd='comando'
3	
4	#eliminar alias
5	unalias cmd

INFORMACION Y SUPERVISIÓN DEL SISTEMA

1	#limpiar pantalla de terminal (mismo efecto que control+l)
2	clear
3	
4	#reiniciar sesion de terminal
5	reset
6	
7	#terminar sesión de terminal local o remoto (ssh) y finalizar procesos iniciados
8	exit

GESTIÓN DE RECURSOS

1	#mostrar tareas ejecutandose y su uso de recursos
2	top
3	
4	#mostrar tareas ejecutables y recursos mejorado
5	htop
6	
7	#muestra el estado de la RAM en megabytes
8	free -h

ESPACIO DE DISCO

1	#mostrar una lista de las particiones montadas
2	df -h
3	
4	#mostrar el tamaño de los archivos y directorios ordenados por tamaño
5	ls -lSr more
6	

7	#Estimar el espacio usado por el directorio 'dir1'
8	du -sh directorio
9	
10	#mostrar el tamaño de los archivos y directorios ordenados por tamaño
11	du -sk * sort -rn

INFORMACIÓN DE SISTEMA

1	#mostrar datos de usuarios conectados
2	who -a
3	
4	#mostrar historial de reinicio
5	last reboot
6	
7	#Mostrar arquitectura y versión de Linux y Kernel
8	uname -a
9	
10	#mostrar el kernel cargado.
11	lsmod
12	
13	#mostrar componentes de hardware del sistema.
14	dmidecode -q
15	
16	#Listar particiones de disco duro
17	*cat /etc/fstab
18	
19	#mostrar características de disco duro
20	hdparm -i /dev/hda
21	
22	#mostrar dispositivos PCI
23	lspci
24	
25	#mostrar dispositivos USB
26	lsusb
27	
28	#mostrar eventos de proceso de carga de kernel
29	tail /var/log/dmesg
30	
31	#mostrar los eventos del sistema
32	tail /var/log/messages
33	
34	#mostrar lista de archivos abiertos por procesos
35	lsof -p \$\$
36	
37	#mostrar lista de archivos abiertos en un camino dado del sistema
38	lsof /directorio
39	
40	#mostrar llamadas del sistema hechas y recibidas por un proceso
41	strace -c ls >/dev/null
42	
43	#mostrar las llamadas a la biblioteca
44	strace -f -e open ls >/dev/null
45	
46	#mostrar interrupciones en tiempo real
47	watch -n1 'cat /proc/interrupts'

APAGADO Y REINICIO DE SISTEMA

1	#cerrar sesion usuario
2	logout
3	
4	#apagar el sistema ahora
5	shutdown now
6	
7	#reiniciar sistema ahora
8	shutdown -r now

9	
10	#apagado programado
11	shutdown horas:minutos &
12	
13	#cancelar apagado programado
14	shutdown -c

FECHAS

1	#mostrar la fecha del sistema
2	date
3	
4	#mostrar calendario de un año
5	cal año
6	
7	#mostrar calendario de mes y año
8	cal mes año 2011

AYUDAS

1	#manual online de comando
2	man comando
3	
4	#muestra un resumen descriptivo de la funcion de comando
5	whatis comando
6	
7	#buscar comandos por la tarea realizada (inverso del anterior)
8	apropos texto

OPERACIONES CON ARCHIVOS Y DIRECTORIOS

os comandos que requieran introducir nombres de archivos o directorios podéis **pulsar dos veces tabulación** para autocompletar la ruta, u obtener un listado de archivos disponibles.

DESPLAZARSE ENTRE DIRECTORIOS

1	#ir a directorio de raiz
2	cd
3	
4	#/ir a directorio anterior
5	cd ..
6	
7	#entrar en directorio (ruta absoluta)
8	cd /directorio1/directorio
9	
10	#entrar en directorio (ruta relativa)
11	cd directorio1/directorio2
12	
13	#ir a directorio de usuario
14	cd ~
15	
16	#ir a ultimo directorio visitado
17	cd -
18	
19	#mostrar ruta actual
20	pwd

LISTAR ARCHIVOS Y DIRECTORIOS

1	#mostrar archivos y directorios
2	ls
3	
4	#mostrar archivos y directorios con detalles
5	ls -l
6	
7	#mostrar archivos y directorios incluidos los ocultos
8	ls -a

MANIPULACIÓN DE ARCHIVOS Y DIRECTORIOS

1	#renombrar o mover un archivo o directorio
2	<code>mv origen destino</code>
3	
4	#copiar un archivo
5	<code>cp archivo direccion</code>
6	
7	#copiar un directorio
8	<code>cp -r origen destino</code>
9	
10	#borrar el archivo llamado archivo
11	<code>rm archivo</code>
12	
13	#borrar directorio si está vacío
14	<code>rm -d directorio</code>
15	
16	#borrar directorio y su contenido
17	<code>rm -r directorio</code>
18	
19	#crear nuevo directorio
20	<code>mkdir directorio</code>
21	
22	#crear varios directorios simultáneamente
23	<code>mkdir directorio1 directorio2</code>
24	
25	#crear ruta de directorios
26	<code>mkdir -p /directorio1/directorio2</code>
27	
28	#crear archivo vacío
29	<code>touch archivo</code>
30	
31	#cambiar fecha de archivo (formato año, mes, día, y hora)
32	<code>touch -t 19901230000 archivo</code>

ENLACES SIMBÓLICOS

1	#crear un enlace simbólico al archivo o directorio
2	<code>ln -s archivo lnk1</code>
3	
4	#crear enlace físico al archivo o directorio
5	<code>ln archivo lnk1</code>

CODIFICACIÓN

1	#calcular md5 de un archivo
2	<code>md5sum archivo</code>
3	
4	#codificar archivo GNU Privacy Guard.
5	<code>gpg -c</code>
6	
7	#decodificar archivo GNU Privacy Guard.
8	<code>gpg archivo.gpg</code>

OPERACIONES DE CONTENIDO DE ARCHIVO

TUBERIAS Y REDIRECCIONES I/O

1	#dirigir salida de comando a nuevo archivo
2	<code>comando > archivo_out.txt</code>
3	
4	#dirigir salida de comando para añadir a archivo archivo
5	<code>comando >> archivo_out.txt</code>
6	
7	#dirigir entrada de comandos
8	<code>comando < archivo_in.txt</code>
9	

10	#dirigir salida estandar y salida de error a un fichero:
11	comando &> archivo_out.txt
12	
13	#tuberia, dirigir salida de comando1 como entrada de comando2
14	comando1 comando2

MOSTRAR CONTENIDO DE ARCHIVO

1	#muestra contenido de archivo
2	echo archivo
3	
4	#mostrar contenido de archivo
5	cat archivo
6	
7	#mostrar contenido de archivo empezando por el final
8	tac archivo
9	
10	#mostrar contenido de archivo desplazandose linea a linea
11	more archivo
12	
13	#mostrar contenido de archivo desplazandose adelante o atrás
14	less archivo
15	
16	#mostrar dos primeras lineas de archivo
17	head -2 archivo
18	
19	#mostrar dos ultimas lineas de archivo
20	tail -2 archivo
21	
22	#mostrar en tiempo real las ultimas lineas de archivo (seguimiento)
23	tail -f archivo

MANIPULACIÓN DE TEXTO

1	#convertir minúsculas en mayúsculas
2	echo 'archivo' tr '[:lower:]' '[:upper:]'
3	
4	#eliminar lineas 1 a 5 de archivo
5	sed '3,5d' archivo
6	
7	#eliminar lineas 5 a fin de archivo
8	sed '5,\$d' archivo
9	
10	#eliminar linas en blanco
11	sed '/^\$/d' archivo
12	
13	#eliminar linas en blanco y comentarios
14	sed '/ *#/d; /^\$/d' archivo
15	
16	#sustituir una cadena por otra
17	sed 's/cadena1/cadena2/g' archivo
18	
19	#visualizar unicamente las líneas que contienen cadena
20	sed -n '/cadena/p'

BUSQUEDA DE ARCHIVOS Y EN SU CONTENIDO

BUSCAR DE ARCHIVOS

1	#buscar archivo y directorio por su nombre en todo el sistema
2	find * -name nombre
3	
4	#buscar archivo y directorio por su nombre, dentro de directorio
5	find directorio -name nombre
6	
7	#buscar archivos y directorios pertenecientes a usuario, dentro de directorio
8	find directorio -user usuario

9	
10	#buscar archivos y directorios por su tipo, dentro de directorio
11	(d directory, f regular file, l symbolic link)
12	find directorio -type f tipo
13	
14	#buscar archivos y ejecutar comando
15	find directorio -name nombre -exec comando {} \;
16	
17	#buscar archivos con extensión .ps
18	locate *.ps
19	
20	#mostrar la ruta completa de un ejecutable
21	which ejecutable
22	
23	#mostrar la ubicación de un archivo binario, de ayuda o fuente
24	whereis ejecutable

BUSCAR EN CONTENIDO DE ARCHIVO

1	#buscar cadena en el/los archivos
2	grep cadena archivo
3	
4	#buscar cadena en el/los archivos sin coincidencia de mayusculas
5	grep -i cadena archivo
6	#buscar palabras que comienzan con cadena en el/los archivos
7	grep ^cadena archivo
8	
9	#buscar cadena como palabra completa en el/los archivos
10	grep -w cadena archivo
11	
12	#seleccionar las líneas de archivo que contienen números
13	grep [0-9] archivo
14	
15	#busca recursiva de cadena en el directorio
16	grep cadena -R direccion
17	

GESTIÓN DE USUARIOS Y GRUPOS

USUARIOS

1	#crear un nuevo usuario
2	useradd usuario
3	
4	#crear usuario, version completa
5	useradd -c "Nombre Usuario" -g grupo -d /home/usuario -s /bin/bash usuario
6	
7	#borrar usuario
8	userdel usuario
9	
10	#borrar usuario y eliminar su directorio home
11	userdel -r usuario
12	
13	#cambiar los atributos del usuario.
14	usermod -c "Nombre usuario" -g grupo -d /home/usuario -s /bin/bash usuario
15	
16	#cambiar contraseña del propio usuario.
17	passwd
18	#cambiar la contraseña de usuario (solamente por root).
19	passwd usuario
20	
21	#colocar un fecha de expiracion para contraseña de usuario
22	chage -E 2014-12-31 usuario

GRUPOS DE USUARIOS

1	#crear grupo usuarios
---	-----------------------

2	<code>groupadd nombre_grupo</code>
3	
4	<code>#borrar grupo de usuarios</code>
5	<code>groupdel nombre_grupo</code>
6	
7	<code>#renombrar grupo de usuarios</code>
8	<code>groupmod -n nombre_nuevo nombre_anterior</code>
9	
10	<code>#modificar el grupo actual de un usuario que pertenece a varios grupos</code>
11	<code>newgrp grupo</code>
12	
13	<code>#listar grupos del usuario actual</code>
14	<code>groups</code>
15	
16	<code>#listar todos los grupos</code>
17	<code>cut -d: -f1 /etc/group</code>

COMPROBACIONES

1	<code>#comprueba que el archivo de contraseñas /etc/passwd es correcto</code>
2	<code>pwck</code>
3	
4	<code>#comprueba que el archivo de grupos /etc/group es correcto</code>
5	<code>grpck</code>

PERMISOS Y ATRIBUTOS ESPECIALES

PERMISOS DE ARCHIVOS Y CARPETAS

1	<code>#usar + para colocar permisos y - para eliminar</code>
2	<code>Mostrar permisos.</code>
3	<code>ls -lh</code>
4	
5	<code>#asignar permisos 0777 a fichero</code>
6	<code>#modificar 0777 segun codificacion octal de permisos</code>
7	<code>chmod 0777 fichero</code>
8	
9	<code>#asignar permisos a todos los archivos de un directorio</code>
10	<code>chmod -R 0644 directorio</code>
11	
12	<code>#colocar a directorio permisos de lectura (r), escritura (w) y ejecución (x) al propietario (u), grupo (g) y otros (o).</code>
13	<code>#emplear las opciones necesarias para añadir o quitar los permisos deseados</code>
14	<code>chmod ugo+rwX directorio</code>
15	
16	<code>#cambiar usuario de archivo</code>
17	<code>chown usuario archivo</code>
18	
19	<code>#cambiar usuario a todos los archivos de un directorio</code>
20	<code>chown -R usuario directorio</code>
21	
22	<code>#cambiar grupo de archivo</code>
23	<code>chgrp grupo archivo</code>
24	
25	<code>#cambiar usuario y grupo de archivo.</code>
26	<code>chown usuario:grupo archivo</code>

PERMISOS SUID

1	<code>#visualizar todos los archivos del sistema con SUID configurado</code>
2	<code>find / -perm -u+s</code>
3	
4	<code>#colocar bit SUID en archivo binario. El usuario que ejecute este archivo adquiere los mismos privilegios como dueño</code>
5	<code>chmod u+s /bin/archivo</code>
6	
7	<code>#eliminar bit SUID en archivo binario</code>
8	<code>chmod u-s /bin/archivo</code>
9	

10	#colocar bit SGID en directorio. Similar a SUID pero para directorios
11	chmod g+s /home/directorio
12	#eliminar bit SUID en archivo binario.
13	chmod g-s /home/directorio
14	
15	#colocar un bit STIKY en un directorio. Permite el borrado de archivos solamente a los dueños legítimos
16	chmod o+t /home/directorio
17	#eliminar bit STIKY en un directorio.
18	chmod o-t /home/directorio

ATRIBUTOS ESPECIALES DE ARCHIVO

1	#usar + para colocar permisos y - para eliminar
2	#mostrar atributos especiales.
3	lsattr
4	
5	#permite escribir abriendo un archivo solamente modo append.
6	chattr +a archivo
7	
8	#permite que un archivo sea comprimido / descomprimido automaticamente.
9	chattr +c archivo
10	
11	#asegura que el programa ignore borrar los archivos durante la copia de seguridad.
12	chattr +d archivo
13	
14	#convierte el archivo en invariable, por lo que no puede ser eliminado, alterado, renombrado, ni enlazado.
15	chattr +i archivo
16	
17	#permite que un archivo sea borrado de forma segura.
18	chattr +s archivo
19	
20	#asegura que un archivo sea modificado, los cambios son escritos en modo synchronous como con sync.
21	chattr +S archivo
22	
23	#te permite recuperar el contenido de un archivo aún si este está cancelado.
24	chattr +u archivo

ARCHIVOS EMPAQUETADOS Y COMPRIMIDOS

ARCHIVOS TAR

1	#mostrar contenido de archivo tar
2	tar -tf archive.tar
3	
4	#crear archivo tar
5	tar -cvf archivo.tar directorio1
6	
7	#crear archivo tar compuesto por varios archivos y directorios
8	tar -cvf archivo.tar archivo1 archivo2 directorio1
9	
10	#crear archivo tar comprimido en bzip2
11	tar -cvfj archivo.tar.bz2
12	
13	#crear archivo tar comprimido en gzip
14	tar -cvfz archivo.tar.gz directorio1
15	
16	#extraer archivo tar
17	tar -xvf archivo.tar
18	
19	#extraer archivo tar en directorio
20	tar -xvf archivo.tar -C /directorio
21	
22	#extraer archivo tar preservando permisos de usuario
23	tar -xpvzf archivo.tar
24	

25	#descomprimir archivo tar comprimido en bzip2
26	tar -xvfj archivo.tar.bz2
27	
28	#descomprimir archivo tar comprimido en gzip
29	tar -xvzf archivo.tar.gz

ARCHIVOS ZIP

1	#crear un archivo comprimido en zip.
2	zip archivo.zip directorio1
3	
4	#comprimir en zip compuesto por varios archivos y directorios
5	zip -r archivo.zip archivo1 archivo2 directorio1
6	
7	#descomprimir un archivo zip
8	unzip archivo.zip

ARCHIVOS BZ2

1	#comprimir archivo bz2
2	bzip2 archivo
3	
4	#descomprimir archivo bz2
5	bunzip2 archivo.bz2

ARCHIVOS GZ

1	#comprimir archivo gz
2	gzip archivo
3	
4	#descomprimir archivo gz
5	gunzip archivo.gz
6	
7	#descomprimir archivo gz con compresion maxima
8	gzip -9 archivo

ARCHIVOS RAR

1	#crear archivo rar
2	rar a archivo.rar directorio1
3	
4	#crear archivo zip compuesto por varios archivos y directorios
5	rar a archivo.rar archivo1 archivo2 directorio1
6	
7	#descomprimir archivo rar.
8	unrar x archivo.rar

INSTALADORES DE PAQUETES, Y RESPOSITORIOS

1	#mostrar bibliotecas requeridas por programa o comando
2	ldd programa
3	
4	#descargar de Github
5	git clone git://github.com/directorio/proyecto.git

PAQUETES DEB (DEBIAN Y DERIVADOS)

1	#instalar / actualizar un paquete deb
2	dpkg -i paquete.deb
3	
4	#eliminar un paquete deb del sistema
5	dpkg -r paquete
6	
7	#mostrar todos los paquetes deb instalados en el sistema
8	dpkg -l
9	
10	#mostrar todos los paquetes deb con un nombre
11	dpkg -l grep nombre
12	
13	#obtener información en un paquete instalado en el sistema

14	<code>dpkg -s paquete</code>
15	
16	<code>#mostar lista de archivos dados por un paquete instalado en el sistema</code>
17	<code>dpkg -L paquete</code>
18	
19	<code>#mostrar lista de archivos dados por un paquete sin instalar</code>
20	<code>dpkg --contents paquete.deb</code>
21	
22	<code>#verificar a que paquete pertenece a un archivo</code>
23	<code>dpkg -S archivo</code>

ACTUALIZADOR DE PAQUETES APT (UBUNTU Y DERIVADOS)

1	<code>#instalar / actualizar un paquete deb.</code>
2	<code>apt-get install paquete</code>
3	
4	<code>#añadir repositorio</code>
5	<code>sudo sh -c 'echo repositorio' >> /etc/apt/sources.list</code>
6	<code>#tambien</code>
7	<code>sudo add-apt-repository ppa:repositorio</code>
8	
9	<code>#actualizar lista de paquetes</code>
10	<code>apt-get update</code>
11	
12	<code>#actualizar paquetes instalados</code>
13	<code>apt-get upgrade</code>
14	
15	<code>#eliminar un paquete del sistema</code>
16	<code>apt-get remove paquete</code>
17	
18	<code>#buscar un paquete</code>
19	<code>apt-cache search paquete</code>
20	
21	<code>#verificar resolución de dependencias</code>
22	<code>apt-get check</code>
23	
24	<code>#limpiar cache desde paquetes descargados</code>
25	<code>apt-get clean</code>

OPERACIONES CON SISTEMAS DE ARCHIVOS

ANÁLISIS DEL SISTEMA DE ARCHIVOS

1	<code>#Comprobar integridad de archivos en el disco hda1 en sistema Linux</code>
2	<code>fscck /dev/hda1</code>
3	
4	<code>#Comprobar integridad de archivos en el disco hda1 en sistema ext2</code>
5	<code>fscck.ext2 /dev/hda1</code>
6	
7	<code>#Comprobar integridad de archivos en el disco hda1 en sistema ext3</code>
8	<code>fscck.ext3 /dev/hda1</code>
9	
10	<code>#Comprobar integridad de archivos en el disco hda1 en sistema Fat</code>
11	<code>fscck.vfat /dev/hda1</code>
12	
13	<code>#Comprobar integridad de archivos en el disco hda1 en sistema Dos</code>
14	<code>fscck.msdos /dev/hda1</code>
15	
16	<code>#Comprobar bloques defectuosos en el disco hda1.</code>
17	<code>badblocks -v /dev/hda1</code>

FORMATEAR SISTEMA DE ARCHIVOS

1	<code>#formatear hda en sistema Linux</code>
2	<code>mkfs /dev/hda1</code>
3	
4	<code>#formatear hda en sistema FAT32</code>

5	mkfs -t vfat 32 -F /dev/hda1
6	
7	#formatear hda en sistema ext2
8	mke2fs /dev/hda1
9	
10	#formatear hda en sistema ext3
11	mke2fs -j /dev/hda1

MONTAR SISTEMAS DE ARCHIVOS

1	#montar un disco duro hda2
2	mount /dev/hda2 /mnt/hda2
3	
4	#montar un disquetera
5	mount /dev/fd0 /mnt/floppy
6	
7	#montar un cdrom o dvdrom
8	mount /dev/cdrom /mnt/cdrom
9	
10	#montar cd regrabable dvdrom
11	mount /dev/hdc /mnt/cdrecorder
12	
13	#montar un cd o dvd regrabable
14	mount /dev/hdb /mnt/cdrecorder
15	
16	#montar un usb pen-drive o una memoria
17	mount /dev/sda1 /mnt/usbdisk
18	
19	#montar un archivo o una imagen iso
20	mount -o loop file.iso /mnt/cdrom
21	
22	#desmontar un dispositivo llamado hda2
23	umount /dev/hda2
24	
25	#forzar el desmontaje (cuando el dispositivo está ocupado)
26	fuser -km /mnt/hda2

IMÁGENES ISO Y GRABADORES DE CDROM

1	#montar una imagen iso
2	mount -o loop cd.iso /mnt/iso
3	
4	#crear imagen iso de cdrom en disco
5	mkisofs /dev/cdrom > cd.iso
6	
7	#crear imagen iso de un directorio
8	mkisofs -J -allow-leading-dots -R -V "Label CD" -iso-level 4 -o ./cd.iso data_cd
9	
10	#grabar imagen iso en cdrom
11	cdrecord -v dev=/dev/cdrom cd.iso
12	
13	#limpiar o borrar un cd regrabable
14	cdrecord -v gracetime=2 dev=/dev/cdrom -eject blank=fast -force

TRABAJO CON LA SWAP

1	#crear archivo de sistema swap en hda3
2	mkswap /dev/hda3
3	
4	#activar particion swap en hda3
5	swapon /dev/hda3

COPIAS DE SEGURIDAD Y BACKUPS

1	#hacer un backup completo de directorio
2	dump -0aj -f /tmp/archivo.bak /directorio
3	
4	#realizar backup incremental de directorio
5	dump -1aj -f /tmp/archivo.bak /directorio

6	
7	#restaurar un backup iterativamente
8	restore -if /tmp/archivo.bak
9	
10	#sincronizar directorios
11	rsync -rogpav --delete /directorio1 /directorio2
12	
13	#volcar contenido de disco duro a archivo
14	dd if=/dev/sda of=/tmp/archivo
15	#encontrar y copiar todos los archivos con extensión .txt de un directorio a otro
16	find /home/usuario -name '*.txt' xargs cp -av --target-directory=/home/backup/ --parents
17	
18	#encontrar todos los archivos con extensión .log y hacer un archivo bzip
19	find /var/log -name '*.log' tar cv --files-from=- bzip2 > log.tar.bz2
20	

TELECOMUNICACIONES Y OPERACIONES DE RED

DESCARGA ARCHIVOS INTERNET

1	#descargar archivo desde paginaweb
2	wget www.paginaweb.com/archivo
3	
4	#descargar un archivo con la posibilidad de parar la descargar y reanudar más tarde.
5	wget -c www.paginaweb.com/archivo
6	
7	#descargar paginaweb completa
8	wget -r www.paginaweb.com

OPERACIONES DE RED

1	#mostrar la configuración de Ethernet
2	ifconfig eth0
3	
4	#activar interface eth0
5	ifup eth0
6	
7	#deshabilitar interface eth0
8	ifdown eth0
9	
10	#configurar una dirección IP
11	ifconfig eth0 192.168.1.1 netmask 255.255.255.0
12	
13	#configurar eth0 en modo común para capturar paquetes (sniffing)
14	ifconfig eth0 promisc
15	
16	#activar la interface 'eth0' en modo dhcp
17	dhclient eth0
18	
19	#mostrar mesa de recorrido
20	route -n
21	
22	#configurar entrada predeterminada
23	route add -net 0/0 gw IP_Gateway
24	
25	#configurar ruta estática para buscar la red '192.168.0.0/16'
26	route add -net 192.168.0.0 netmask 255.255.0.0 gw 192.168.1.1
27	
28	#eliminar ruta estática
29	route del 0/0 gw IP_gateway
30	
31	#activar recorrido ip
32	echo "1" > /proc/sys/net/ipv4/ip_forward
33	
34	#mostrar nombre del host

35	hostname
36	
37	#buscar nombre del host para resolver el nombre a una dirección ip
38	host www.paginaweb.com
39	
40	#buscar nombre del host para resolver el nombre a una dirección ip y viceversa
41	nslookup www.paginaweb.com
42	
43	#mostrar estado de enlace de todas las interfaces
44	ip link show
45	
46	#mostrar estado de enlace de eth0
47	mii-tool eth0
48	
49	#mostrar estadísticas de tarjeta de red eth0
50	ethtool eth0
51	
52	#mostrar todas las conexiones de red activas y sus PID
53	netstat -tup
54	
55	#mostrar todos los servicios de escucha de red en el sistema y sus PID
56	netstat -tupl
57	
58	#mostrar todo el tráfico HTTP
59	tcpdump tcp port 80
60	
61	#mostrar redes inalámbricas
62	iwlist scan
63	
64	#mostrar configuración de una tarjeta de red inalámbrica
65	iwconfig wlan0
66	
67	#buscar en base de datos Whois
68	whois www.paginaweb.com