

1. cd

El comando cd establece el directorio de trabajo de un proceso.

Ejemplo: cambie el directorio a su directorio de inicio, asumiendo que su variable INICIO está definida.

```
#cd $HOME
```

2. cp

Utilice el comando cp para copiar archivos.

Ejemplo: Copie el archivo testdb a otro directorio del sistema.

```
# cp/tmp/testdb/ home/frank/testdb
```

3. cpio

Utilice el comando cpio para copiar y mover archivos y directorios. También puede utilizarlo para realizar copias de seguridad de directorios vacíos, restaurar archivos de un archivo o crear un archivo.

Ejemplo: Copie todos los archivos y directorios del directorio actual a otro directorio llamado /user/newfiles.

```
# find./-depth|cpio -pdmv/usr/newfiles
```

4. find

El comando find busca en el árbol de directorios enraizado en cada nombre de archivo dado evaluando la expresión dada de izquierda a derecha, de acuerdo con las reglas de precedencia, hasta que se conoce el resultado, momento en el que find pasa al siguiente nombre de archivo.

Ejemplo: busque todos los archivos y directorios en un directorio determinado y envíelos a la salida estándar.

```
# find ./ -depth
```

5. mv

El comando mv mueve un archivo a un directorio o sistema de archivos diferente.

Ejemplo: Mueva el archivo de prueba al directorio /tmp.

```
# prueba mv/tmp
```

6. rsync

El comando rsync sincroniza los datos de una ubicación de disco a otra.

Ejemplo: Transfiera todos los archivos que coincidan con el patrón *.d del directorio actual a los datos del directorio en la máquina hosta.

```
# rsync *.d hosta: datos/
```

7. traceroute

El comando traceroute determina una ruta al host con el fin de determinar los problemas de la red o del enrutador. Si el dominio no funciona o no está disponible, puede rastrear una dirección IP.

Ejemplo: ejecute traceroute a un dominio inaccesible para determinar la causa del problema.

```
# traceroute hostc.org
```

8. vi

El comando vi muestra vi, un editor basado en pantalla preferido por muchos usuarios de Unix.

Ejemplo: Abra vi en modo editor para editar el archivo /etc/hosts.

```
# vi/etc/hosts
```

Utilice 9 comandos básicos de Unix para tener un mayor control sobre su sistema Unix.

Comandos de gestión de archivos

9. awk

El comando awk busca patrones en un archivo y los procesa. Permite a un programador escribir pequeños programas en forma de declaraciones para realizar cambios en archivos de texto cuando aparecen ciertos patrones o extraer datos de manera eficiente de esos archivos.

Ejemplo: cuente el número de líneas en el archivo — similar a wc-l.

```
# awk 'END {print NR}'
```

10. cat

Utilice el comando cat para vincular el contenido del archivo y enviarlo para verlo o imprimirlo. También puede usarlo para mostrar un archivo.

Ejemplo: vea el archivo /etc/hosts.

```
# cat /etc/hosts
```

11. chmod

El comando chmod cambia el permiso de un archivo.

Ejemplo: cambie el permiso del archivo llamado newfile para leer, escribir y ejecutar (rwx) para propietario, grupo y otros.

```
# chmod 777 newfile
```

12. chown

Utilice el comando chown para cambiar la propiedad del archivo.

Ejemplo: cambie la propiedad del archivo llamado newfile a frank.

```
# chown frank newfile
```

13. exportfs

El comando exportfs mantiene una lista de los sistemas de archivos exportados del sistema de archivos de red (NFS).

Ejemplo: exportar todos los directorios.

```
# exportfs -a
```

14. ftp

El comando ftp permite copiar archivos de un lado a otro en diferentes máquinas host.

Ejemplo: iniciar ftp.

```
# ftp
```

15. head

El comando head genera la primera parte de un archivo.

Ejemplo: muestre las primeras 10 líneas en el archivo /etc/hosts.

```
# head -10 / etc / hosts
```

16. ls

El comando ls muestra información sobre archivos, como el contenido de un directorio. El comando ls tiene varios subcomandos, como ls -r, que modifica el campo de clasificación para invertir el orden en que se muestran los archivos.

Ejemplo: muestra una lista larga de información de archivo en el directorio de datos.

```
# ls -l datos
```

17. rm

El comando rm elimina un archivo o un grupo de archivos.

Ejemplo: Pida a los usuarios que se aseguren de que desean eliminar los archivos del directorio antes de hacerlo.

```
# rm -i *
```

18. tail

El comando tail genera la última parte de un archivo.

Ejemplo: muestra las últimas 10 líneas del archivo /etc/hosts.

```
# tail -10 / etc / hosts
```

Comandos de manipulación de archivos

19. alias

El comando alias le permite sustituir un nombre pequeño o más familiar en lugar de una cadena larga.

Ejemplo: Adapte el comando du para usar unidades de 1K.

```
# alias du = du -k
```

20. eco

El comando eco hace eco de una variable de cadena en la salida estándar.

Ejemplo: informe Hello World.

```
# echo Hello World
```

21. export

El comando de export establece el valor de una variable para que sea visible para todos los subprocesos que pertenecen al shell actual.

Ejemplo: Exportar la variable TERM.

```
# TERM = vt220; export $ TERM
```

22. grep

Utilice el comando grep para buscar en uno o más archivos una cadena de caracteres o patrón determinado. También puede usarlo para reemplazar un solo carácter o cadena de caracteres por otro.

Ejemplo: busque instancias de hello en el archivo text.txt.

```
# grep hello text.txt
```

23. ln

El comando ln establece vínculos entre archivos.

Ejemplo: establezca un enlace simbólico entre dos archivos: sourcefile y newfile.

```
# ln -s sourcefile newfile
```

24. sed

Utilice el comando sed para realizar una operación o un conjunto de operaciones en el texto de entrada y el texto modificado de salida.

Ejemplo: inserte una línea en blanco debajo de cada línea que coincida con la prueba.

```
# sed '/ test / G'
```

Comandos de gestión de red

25. ifconfig

El comando ifconfig verifica la configuración de la interfaz de red. Úselo para verificar o solucionar problemas de configuración de un usuario.

Ejemplo: muestra toda la información del adaptador.

```
# ifconfig -a
```

26. ifup

El comando ifup inicia una interfaz de red.

Ejemplo: Abra la interfaz en0.

```
# ifup en0
```

27. ifdown

El comando ifdown cierra la interfaz de red.

Ejemplo: elimine todas las interfaces que estén activas.

```
# ifdown -a
```

28. netstat

El comando netstat muestra el estado de la red mostrando simbólicamente el contenido de varias estructuras de datos relacionadas con la red. Tiene varios formatos de salida en función de la información presentada.

Ejemplo: mostrar información en la tabla de enrutamiento.

```
# netstat -r
```

29. ping

El comando ping envía solicitudes de eco al host que especifique en la línea de comando y enumera el tiempo de ida y vuelta de las respuestas. Cuando finaliza el ping, se resumen los resultados, lo que le brinda un tiempo promedio de ida y vuelta y un porcentaje de pérdida de paquetes. Utilice ping para determinar los problemas de conexión de red entre dos hosts.

Ejemplo: enviar solicitudes de eco a hosta.

```
# ping hosta
```

Comandos de recopilación de datos

30. df

El comando df informa sobre el uso del espacio en disco del sistema de archivos.

Ejemplo: muestra el espacio en disco en kilobytes.

```
# df -k
```

31. du

El comando du informa sobre los tamaños de los árboles de directorios.

Ejemplo: resume el tamaño de archivo del directorio desde el que ejecuta el comando.

```
# du -s
```

32. env

El comando env muestra información sobre el entorno actual.

Ejemplo: mostrar todas las variables de entorno.

```
# env
```

33. id

El comando id imprime identificadores únicos (UID) e ID de grupo (GID) reales y efectivos.

Ejemplo: Muestre todos los UID y GID en /etc/passwd.

```
# id
```

34. man

El comando man genera información sobre los comandos y un mecanismo de búsqueda de palabras clave para los comandos requeridos.

Ejemplo: imprima información sobre el comando tar.

```
#man tar
```

35. ps

El comando ps informa sobre el estado del proceso. Informa de dos formas: la forma System V y el método Berkeley. El método de Berkeley no usa el guión (-) antes de las banderas.

Ejemplo: Informar la información del proceso del paquete utilizando el método System V.

```
# ps -ef
```

36. pwd

El comando pwd muestra el nombre del directorio de trabajo actual.

Ejemplo: Informe el nombre del directorio en el que se encuentra actualmente.

```
# pwd
```

37. uname

El comando uname imprime el nombre del sistema y otra información relacionada sobre su sistema.

Ejemplo: muestra una lista larga de información completa del sistema.

```
# uname -a
```

38. vmstat

El comando vmstat proporciona una instantánea de todo lo que sucede en un sistema determinado, como el uso de la CPU o información sobre la memoria y la E/S.

Ejemplo: inicie vmstat y ejecútelo cada dos segundos durante 10 iteraciones.

```
# vmstat 2 10
```

Comandos de automatización del sistema

39. crontab

El comando crontab manipula cron para programar tareas.

Ejemplo: muestra tu archivo crontab.

```
# crontab -l
```

40. enable

El comando enable habilita o deshabilita una impresora.

Ejemplo: habilitar impresora1.

```
#enable impresora1
```

41. exit

El comando exit le permite salir de un programa, shell o red Unix.

Ejemplo: salir del caparazón.

```
# exit
```

42. gzip

El comando gzip comprime archivos sin utilizar algoritmos patentados. Utilice gunzip para descomprimir un archivo GZIP.

Ejemplo: comprima el archivo file1 y cámbiele el nombre file2 en formato GZIP.

```
# gzip -c archivo1> archivo2.gz
```

43. shutdown

El comando de shutdown apaga la computadora. Combínelo con variables como -h para detener o -r para reiniciar. Ejecute siempre man antes de ejecutar shutdown porque los indicadores difieren entre las variedades de Unix.

Ejemplo: reinicie la caja en AIX.

```
# shutdown -Fr
```

44. tar

El comando tar crea archivos tar. Puede utilizar tar en archivos creados previamente para extraer archivos, almacenar archivos adicionales o actualizar y enumerar archivos. El comando tar puede dirigir su salida a dispositivos, archivos u otros programas disponibles. Tar también puede acceder a dispositivos o archivos remotos.

Ejemplo: Cree un archivo tar en el dispositivo de cinta st0, con el contenido de / home.

```
# tar -cvf / dev / st0 / home
```

Comandos de gestión de usuarios

45. last

El comando last muestra los últimos usuarios que iniciaron sesión en el sistema.

Ejemplo: enumere los tiempos de apagado y los cambios en el nivel de ejecución de los usuarios anteriores.

```
# last -x
```

46. ssh

Utilice el comando ssh para conexiones de red seguras y tunelización de servicios TCP.

Ejemplo: conecte al usuario ken a hostb.com.

```
# ssh -l ken hostb.com
```

47. sudo

El comando sudo le permite dar a ciertos usuarios o grupos de usuarios la capacidad de ejecutar comandos como root u otro usuario mientras registra comandos y argumentos.

Ejemplo: Permita que los usuarios que usan sudo ejecuten el comando adduser como si fueran root.

```
$ sudo -u root adduser testuser
```

48. w

El comando w imprime un resumen del sistema actual y la información del usuario. Proporciona información actual sobre qué usuarios están conectados al sistema y qué están haciendo.

Ejemplo: imprima una breve lista de información sobre los usuarios actuales.

```
# w -s
```

49. who

El comando who muestra información sobre los usuarios que han iniciado sesión en el sistema.

Ejemplo: enumere toda la salida disponible del comando who para cada usuario.

```
# who-a
```


50. whoami

El comando `whoami` muestra información sobre el usuario con el que inicia sesión. Este comando puede ayudar a los administradores que tienen varios inicios de sesión.

Ejemplo: muestra el nombre del usuario con el que está conectado actualmente.

`whoami`

Puede usar miles de comandos adicionales para un control aún más profundo sobre su sistema Unix. Los comandos de Unix a menudo se superponen con los comandos de Linux, por lo que aprender a manejar Linux puede proporcionar una visión más profunda de Unix.

Comando	Sintaxis y Descripción	Ejemplo	Opciones
>, <, >>, << (redirección)	Permiten redireccionar la entrada o salida de un comando a un archivo.	<code>cat arch1 arch2 >> arch3</code> (añade <i>arch1</i> y <i>arch2</i> al final de <i>arch3</i>)	<, > Redireccionan la entrada y la salida (respectivamente) borrando previamente el archivo de destino <<, >> Redireccionan la entrada y la salida a continuación del archivo de destino.
 (tuberías)	Permiten redireccionar la entrada o salida de un comando a otro comando.	<code>ls -l lp</code> (realiza un listado del directorio actual por impresora)	-
*	En todos aquellos comandos, que esperan el nombre de un archivo o la ruta de un directorio, se sustituye por cualquier cadena de caracteres, incluido la cadena vacía.	-	-
?	En todos aquellos comandos, que esperan el nombre de un archivo o la ruta	-	-

	de un directorio, se sustituye por un carácter cualquiera.		
awk	Lenguaje para búsqueda y procesado de patrones y expresiones regulares.	-	-
cal	cal [mes] año Devuelve un calendario del año (y el mes) especificados.	cal 09 1993 (devuelve el calendario de septiembre de 1993)	-
calendar	Servicio de agenda que permite indicar fechas y horas en el que se le avise	-	Consiste en poner el archivo <i>calendar</i> en el directorio de usuario.
cat	cat archivo ... Concatena e Imprime	cat arch2 (envía el archivo <i>arch2</i> a la pantalla)	-
cc	Compilador de C que suele proporcionar cualquier UNIX	-	-
chmod	chmod [u g o] {+ -} {rwx} {archivo directorio} Cambio de modos o permisos de archivos	chmod g-rw cuentas (quita los permisos de lectura y escritura del archivo <i>cuentas</i> para los usuarios pertenecientes al grupo)	u Propietario g Miembros del grupo o Otros usuarios + Activa el permiso - Quita el permiso r Lectura w Escritura x Ejecución
cmp	cmp arch1 arch2 Compara dos archivos	cmp datos1 historico (compara los archivos <i>datos1</i> y <i>historico</i> retornando la primera línea y posición del primer octeto que difiera)	-
comm	comm [-1,-2,-3] arch1 arch2	comm clientes1 clientes2 (en la primera columna aparecen las líneas que sólo aparecen	1 No genera la primera columna 2 No genera la segunda columna

	Busca las líneas en común entre dos archivos.	en <i>clientes1</i> , en la segunda columna sólo las que aparecen en <i>clientes2</i> y en la tercera, aquellas que aparecen en ambos)	3 No genera la tercera columna
cut	cut [-c -f [-d]] archivo ... Recorta verticalmente campos de archivos	cut -f2,4,6 -d, (recorta los campos 2, 4 y 6 utilizando como delimitador de campo la coma.	c Divide contando por columnas (caracteres) f Divide contando por campos (fields) d Especifica el delimitador de campo
date	date Retorna el día y la hora	-	-
diff	diff [-b,-e] arch1 arch2 diff [-b,-e,-r] dir1 dir2 Busca las diferencias entre líneas de dos archivos o directorios.	diff -r /usr/pepe /usr/jose (averigua las diferencias entre todos los archivos de los dos directorios especificados y de sus subdirectorios)	b Ignora blancos al principio de línea e Formatea la salida para <i>ed</i> r Recorrer subdirectorios
echo	echo [cadena] Imprime <i>cadena</i>	-	-
ex	ex Editor de líneas	-	-
find	find ruta criterio acción... Busca archivos que cumplen el criterio y realiza una acción (o varias) sobre ellos.	find /tmp -name fact* -print (busca e imprime el nombre de los archivos en el directorio <i>/tmp</i> que comiencen por <i>fact</i>)	Los criterios para los archivos pueden ser: -name archivo Con ese nombre -size [+ -]n De tamaño n bloques -links [+ -]n Con n links -mtime [+ -]n Accedido hace n días -mtime [+ -]n Modificado

			<p>hace n días <code>-newer</code> <code>archivo</code> Modifica do después de <i>archivo</i></p> <p>Donde aparece n se asume el valor exacto. +n y -n significan mayor o menor que, reespectivamente.</p> <p>Las acciones pueden ser: <code>-print</code> Imprime la ruta de los archivos encontrados <code>-exe</code> <code>comando</code> Ejecuta <code>comando</code> <code>-ok</code> <code>comando</code> Ejecuta <code>comando</code> pidiendo confirmación.</p> <p>En estos dos últimos casos el archivo encontrado se representa con { }</p>
grep	<pre>grep [-n,-c,-v] patrón archivo</pre> <p>Busca ocurrencias de un patrón en un archivo</p>	<pre>grep valencia cities</pre> <p>(busca todas las ocurrencias de la palabra <i>valencia</i> en el archivo <i>cities</i>)</p>	<p>a Poner el número de cada línea donde aparece el patrón c Sólo aparecen las líneas que contienen el patrón v Aparecen las líneas que no contienen el patrón</p>
help	<pre>help</pre> <p>Invoca la ayuda</p>		-
kill	<pre>kill [-9] ident_proceso</pre>	<pre>kill -9 777</pre>	9 Produce una terminación

	Envía un mensaje de terminación a un proceso	(mata el proceso 777)	indiscutible del proceso
ln	ln arch1 arch2 Crea vínculos (<i>links</i>) entre archivos	ln juan /usr/invitados/juanperez (enlaza el archivo <i>juan</i> con el directorio <i>/usr/invitados/juanperez</i>)	-
login	login [usuario] Permite entrar a trabajar con otro usuario	login juan (pedirá la palabra de paso de juan para entrar a trabajar con ese usuario)	-
lp	lp [-d] archivo ... Lanza a impresora	lp facturas ivas (envía a impresora los archivos <i>facturas</i> y <i>ivas</i>)	d Selecciona impresora
lpstat	lpstat [-p] archivo ... Obtiene el estado de impresión de un archivo	lpstat facturas (obtiene información de la impresión del archivo <i>facturas</i>)	p Información sobre la impresora
ls	ls [-{a,c,l,p,r,s}] archivo ... Lista el contenido de un directorio	ls -la (lista todos los archivos del directorio actual con información extendida)	a Lista todos los archivos c Ordena por fecha l Listado Largo p Señala cuáles son directorios con / r Invierte el orden del listado s Indica el tamaño en bloques
mail	mail mail dirección ... Para enviar y recibir correo.	mail ploro@uv.es Este es el cuerpo del mensaje <CTRL-D> Envía el texto hasta CTRL-D a ploro@uv.es	Cuando se invoca sin parámetros se tienen al menos los siguientes subcomandos: <return> Lee los siguientes mensajes. Si no hay más sale de <i>mail</i> .

			<p>* Da un resumen de comandos de <i>mail</i></p> <p>- Vuelve al mensaje anterior p Reimprime el mensaje a [arch] Añade el mensaje a <i>arch</i> (por defecto <i>arch</i> es <i>mb ox</i>) d Elimina el mensaje q Sale de <i>mail</i> <CTRL-D> Igual que q.</p>
man	man [sección] Permite consultar el manual	man grep (obtiene ayuda sobre el comando <i>grep</i>)	-
mesg	mesg [-y -a] Habilita o inhabilita la entrada de mensajes <i>write</i> .	mesg y (habilita la entrada de mensajes)	y Habilita a Desabilita
mkdir	mkdir directorio Crea un nuevo directorio	mkdir trabajo (crea un nuevo subdirectorio denominado <i>trabajo</i> en el directorio actual)	-
mv	mv arch1 arch2 mv arch1 directorio Traslada o renombra archivos	mv facturas trabajo/ (mueve el archivo facturas al directorio trabajo)	-
passwd	passwd Permite cambiar la palabra de paso	-	-
paste	paste [-dn] 	paste -d, codfact facturas 	dn Establece el carácter de enlace

	Permite combinar líneas de varios archivos	(genera un archivo que contiene en cada línea, la línea correspondiente del archivo <i>codfact</i> , un carácter coma de separación y la línea correspondiente del archivo <i>facturas</i>).	
pr	pr [-d, -ln, -p, -t, -wn] Imprime (formatea) archivos	pr -pl24 facturas (visualiza por pantalla, de veinticuatro en veinticuatro líneas y separando por pausas, el archivo <i>facturas</i>)	d Líneas a doble espacio ln Establece la longitud (altura) de página en <i>n</i> líneas. p Hace pausa entre páginas t No incluye encabezamiento de cada página wn Establece la anchura de página en <i>n</i> columnas.
ps	ps [-a, -u, -x] Retorna los procesos activos	ps -aux (visualiza en formato extendido todos los procesos activos y sus usuarios)	a Retorna todos los procesos u Indica el usuario de cada proceso x Información extendida
pwd	pwd Indica el directorio actual	-	-
rm	rm [-l, -r] archivo ... Borra archivos	rm -r trabajo (elimina el directorio trabajo y todos los archivos y subdirectorios que cuelgan de él)	l Protege archivos existentes r Borra recursivamente todos los subdirectorios y archivos que cuelgan del directorio a borrar
rmdir	rmdir directorio ...	rm trabajo (borra el directorio trabajo en el caso que esté vacío)	-

	Borra directorios siempre y cuando estén vacíos		
sed	sed [opciones] Editor de flujo	-	-
sort	sort [-b,-d,-f,-n,-o archsal, -r] archent Ordena las líneas de un archivos según orden lexicográfico.	sort -f -o salida entrada (ordena el archivo <i>entrada</i> en el archivo <i>salida</i> ignorando mayúsculas y minúsculas)	b Ignorar los espacios del principio de línea d Ordena según diccionario f Ignora diferencias entre mayúsculas y minúsculas n Ordena los números por el valor y no por orden lexicográfico o archivo El resultado se escribe en archivo r Revierte el orden
spell	spell archivo Corrige ortográficamente un archivo	-	-
tail	tail [-n] archivo Recorta las últimas líneas de un archivo.	tail -100 diario (Imprime las últimas 100 líneas del archivo diario)	-
uniq	uniq [-u,-d,-c] archent [archsal] Suprime las líneas duplicadas en un archivo.	uniq -d ventas masvendidos (produce en el archivo <i>masvendidos</i> aquellas líneas de <i>ventas</i> que aparecen más de una vida)	u Sólo retorna aquellas líneas duplicadas d Retorna las líneas duplicadas c Retorna el número de veces que aparece repetida cada línea
vi	vi Editor de pantalla	vi facturas	Dentro del editor y desde el modo

		(edita el archivo facturas. Si no existe lo crea)	comando (que se accede con ESC): i Inserta caracteres x Borra caracteres :q Salir sin grabar :x Salir grabando cambios
wc	wc [-l,-w,-c] archivo ... Cuenta las palabras de un archivo (WordCount).	wc -l diario (cuenta las líneas del archivo <i>diario</i>)	l Cuenta líneas w Cuenta palabras c Cuenta caracteres
who	who [am I] Indica los usuarios conectados al sistema	who (informa sobre los usuarios conectados en ese momento)	am I Sólo proporciona información sobre el propio usuario.
write	write nombredeusuario Envía una nota a otro usuario del sistema	write vanessa Nos vemos a la salida? <CTRL-D> (Envía una nota a la usuario <i>vanessa</i>)	-