

Raspberry Pi Comandos Hoja de trucos

74

Comandos Raspberry Pi



RaspberryTips

GESTIÓN DE FICHEROS

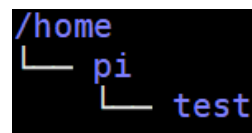
Estos comandos son los básicos que todo principiante en Linux debería aprender para navegar por el árbol de archivos de Linux desde un terminal



Recordatorio:

La organización de archivos de Linux es un árbol, que comienza en /
Cada subcarpeta añade un nuevo nivel bajo /

Por ejemplo, en la imagen puedes ver el árbol para esta carpeta : /home/pi/prueba



• CD <CARPETA>

Cambia de directorio, ve a la carpeta especificada

Ruta absoluta: `cd /home/pi/test`
Ruta relativa: `cd test`



NB: "Absoluta" es cuando utilizas la ruta completa.
En el caso de "relativa", sólo se introduce la ruta desde el directorio actual (en el segundo ejemplo, es necesario estar ya en la carpeta /home/pi).

• MKDIR <CARPETA>

Crea una nueva subcarpeta en la ruta actual o en la especificada

Directorio actual: `mkdir test`
Específico: `mkdir /home/pi/test`



NB: El primer ejemplo crea una carpeta en su directorio actual (ruta relativa)
El segundo crea un nuevo directorio en el parámetro exacto (ruta absoluta)

• MV <SRC> <TARGET>

Mueve un archivo o directorio a otra ubicación (cortar/pegar)

Mover un archivo: `mv test.txt /home/pi`
Mover una carpeta: `mv /home/pi/test /home/pi/test2`



NB: El comando "mv" siempre está en modo recursivo

• MORE <FICHERO>

Muestra el contenido del archivo, página por página, desde el principio

Ruta absoluta: `more test.txt`
Ruta relativa: `more /home/pi/test.txt`



NB: Para los archivos largos, pulse "espacio" para continuar o "q" para salir.

• LS (CARPETA)

Lista los archivos y directorios, en la carpeta actual o en la especificada

Directorio actual: `ls`
Específico: `ls /home/pi/test`



NB: Puede utilizar opciones con ls para obtener una vista más detallada de los archivos y carpetas, ej: `ls -latr /home/pi`

• CP <SOURCE> <TARGET>

Copia un archivo o directorio en otra ubicación (copiar/pegar)

Copiar un fichero `cp test.txt /home/pi`
Copia recursiva: `cp -r /home/pi/test /home/user/`



NB: Utilice la opción recursiva para copiar una carpeta y todos sus archivos y carpetas

• CAT <FICHERO>

Muestra el contenido del fichero, sin paginación

Mostrar en archivo: `cat test.txt`
Usar patrón: `cat *.txt`



NB: Un patrón permite mostrar el contenido de todos los archivos similares

• TAIL <FICHERO>

Muestra el final del archivo

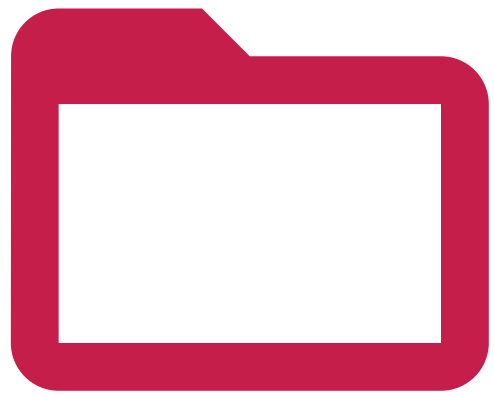
Uso básico: `tail test.txt`
Recuento de líneas: `tail -n20 test.txt`
Visualización en tiempo real: `tail -f test.txt`



NB: La opción -n permite pedir que se muestre un número específico de líneas
La opción -f actualiza la visualización cada vez que se modifica el archivo (perfecto para la supervisión de archivos de registro)



GESTIÓN DE FICHEROS (2)



• HEAD <FICHERO>

Similar a "tail" pero para mostrar el principio del fichero

Visualizar 10 líneas: `head test.txt`
Con recuento de líneas: `head -n20 test.txt`

• GREP

Grep es una potente (y compleja) herramienta para buscar cadenas en un texto o archivo

Buscar una cadena en un archivo `grep "dhcp" /var/log/syslog`
Filtrar la salida de un comando `ls -latr | grep ".php"`
Con un guión: `/home/pi/script.sh | grep error`



NB: La opción | (pipe), permite ejecutar un comando sobre otra salida. Es necesario utilizar comillas para búsquedas complejas con espacios o caracteres especiales.

• NANO <FICHERO>

Abre y edita el archivo especificado. Nano es un potente editor de texto en un terminal

Uso básico: `nano /home/pi/test.txt`



NB: Nano creará el archivo si no existe

• TAR

Tar es la forma Linux de gestionar archivos comprimidos

Crear un nuevo archivo: `tar -cvfz archive.tar.gz /home/pi/test`
Extraer archivos: `tar -xvfz archive.tar.gz`



Options:
-c es para Comprimir, -x para eXtraer
-v: modo verbose, -z: usar gZip para comprimir, -f especificar el nombre del archivo
Utilice "man tar" para obtener más información

• TOUCH <FICHERO>

Crear un nuevo archivo vacío

Directorio actual: `touch test.txt`
Específico: `touch /home/pi/test.txt`



NB: La mayoría de las veces, nano es una mejor opción para crear un archivo, ya que puedes editarlo directamente

También son posibles usos avanzados:

Expresiones regulares: `grep "dhcp|dns" /var/log/syslog`
Opciones de comando: `grep -A2 -B4 'Fatal error' /var/log/syslog`
Búsqueda invertida: `grep -v 'Notice' /var/log/syslog`

El | en las expresiones regulares le permite usar O (una o más condiciones). La opción -A también captura las líneas X "después" de la condición coincidente, -B es para "antes". Por último, la opción -v sirve para filtrar las líneas que no coinciden con la condición

• RM <FICHERO>

Elimina un archivo o directorio

Eliminar archivo: `rm test.txt`
Eliminar directorio: `rm -rf /home/pi/test`



NB: Es necesario utilizar las opciones -rf para eliminar un directorio aunque no esté vacío (recursivo + forzado)

• ZIP / UNZIP

Zip es similar a tar, pero se utiliza principalmente en sistemas Windows

Crear un nuevo archivo: `zip -r archive.zip /home/pi/test`
Extraer archivos: `unzip archive.zip`



NB: La opción -r sirve para comprimir todo el contenido de la carpeta. Puede utilizar la opción -d para extraer los archivos de una carpeta específica. Utilice "man zip" o "man unzip" para conocer todas las opciones disponibles



GESTIÓN DE FICHEROS (3)



- **PWD**

Un comando sencillo para mostrar el directorio actual

Ejemplo: `pwd`

- **FIND**

Find te permite buscar archivos en tu Raspberry Pi, hay un montón de opciones

Buscar un nombre de archivo: `find /home/pi -iname test.txt`

Filtrar extensiones: `find /home/pi -iname *.php`

Encontrar sólo directorios: `find / -type d -iname test`



NB: -iname significa "insensitive case", puede utilizar -name si lo prefiere
Puede utilizar "-type f" para encontrar sólo archivos

- **TREE**

Otra herramienta para obtener detalles sobre tu ubicación actual, en formato de árbol

Directorio actual: `tree`

Carpeta específica: `tree /home/pi/`



NB: Existen algunas opciones para filtrar la salida, seleccionando sólo el directorio, gestionando los enlaces simbólicos o estableciendo un nivel máximo de profundidad

Opciones más avanzadas:

Tamaño de los archivos: `find / -size +10M`

Archivos modificados recientemente: `find /home -mtime -2`

Ejecutar comando en los resultados:

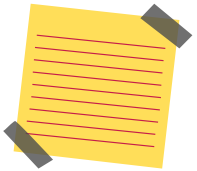
`find /var/log -iname *.log.gz -exec rm {} \;`

El primer comando muestra todos los archivos de más de 10M en el disco
El comando -mtime -2 comprueba los ficheros modificados en los últimos dos días
El parámetro {} del último comando será sustituido por el nombre del fichero
Consulte "man find" para más información



COMANDOS DE RED

Estos son los principales comandos que debes conocer para gestionar y utilizar la red en tu Raspberry Pi



Reminder:

Los modelos recientes de Raspberry Pi vienen con dos interfaces : Ethernet y Wifi
Ethernet se llama eth0 y la Wifi es wlan0

• IFCONFIG

Muestra su configuración de red actual (dirección IP, dirección Mac, ...)

Uso: `ifconfig`



NB: Puede añadir un nombre de interfaz para mostrar sólo ésta, por ejemplo: "ifconfig wlan0"

• IFUP / IFDOWN

Permite activar o desactivar una interfaz específica

Habilitar interfaz: `sudo ifup eth0`

Desactivar interfaz: `sudo ifdown eth0`



NB: Puede ayudar a desactivar la interfaz inalámbrica mientras está conectado por cable

• HOSTNAME

Muestra o establece el nombre de host Raspberry Pi

Mostrar nombre de host: `hostname`

Establece un nuevo nombre de host: `sudo hostname rpitips`

• SSH <USUARIO>@<IP>

Se conecta a otro sistema Linux con SSH

Ejemplo: `ssh pi@192.168.1.1`

• RSYNC

Similar a SCP con más opciones como la comparación delta y algunas otras optimizaciones.

Sintaxis: `rsync <archivo> <usuario>@<ip>:<ruta>`

Ejemplo: `rsync test.txt pi@192.168.1.1:/home/pi/`

• IWCONFIG

Muestra información sobre la configuración de la red inalámbrica (SSID, velocidad, ...)

Uso: `iwconfig`



NB: También puede mostrar una interfaz específica con iwconfig wlan0

• PING <HOST>

Comprueba si el host está vivo

Uso: `ping 192.168.1.1`



NB: Lea "man ping" para ver todas las opciones disponibles

• WGET <URL>

Descargar un archivo con el terminal

Uso básico: `wget http://192.168.1.1/test.txt`

Cambiar nombre de archivo:

`wget http://192.168.1.1/test.txt -O destino.txt`

• SCP

Copia un archivo a través de la red mediante SSH

Sintaxis: `scp <archivo> <usuario>@<ip>:<ruta>`

Ejemplo: `scp test.txt pi@192.168.1.1:/home/pi/`

Copia local:

`rsync /home/pi/* /media/usb/`

Copia recursiva remota:

`rsync -auzr /home/pi/Documents/* pi@192.168.1.1:/home/pi/Documents/`



NB: Utilice "man rsync" para obtener todas las opciones posibles



GESTIÓN DE PAQUETES

Una vez que tengas la red funcionando, probablemente actualizarás tu sistema e instalarás el paquete necesario. En esta página, tienes todos los comandos necesarios para hacerlo desde un terminal.



Vocabulario:

En Linux, cada software es un **paquete**, así como cada **dependencia**. Se descargan nuevos paquetes de **repositorios** (servidores que alojan paquetes). Necesita usar una herramienta llamada **apt** para buscar, instalar y actualizar paquetes en Debian/RPI OS. Todos estos comandos necesitan privilegios de root, tiene que usar **sudo** antes de cada uno de ellos.

• APT UPDATE

Descarga la última lista de paquetes de sus repositorios.

Uso: `sudo apt update`



NB: Para añadir un nuevo repositorio, puede editar la configuración de apt en `/etc/apt/sources.list`, o seguir las instrucciones del editor de software.

• APT UPGRADE

Descarga e instala la última versión de cada paquete disponible en el repositorio.

Uso: `sudo apt upgrade`



NB: Es necesario ejecutar `apt update` antes de hacer esto, para obtener las últimas versiones. La opción `-y` le permite aceptar automáticamente la instalación.

• RPI-UPDATE

Actualiza todo en tu Raspberry Pi, úsalo con precaución.

Uso: `sudo rpi-update`

• APT INSTALL <PACKAGE>

Instala el paquete especificado en su sistema.

Uso: `sudo apt install phpmyadmin`



NB: Utilice el siguiente comando de búsqueda para conocer el nombre exacto de un paquete.

• APT REMOVE <PACKAGE>

Desinstalar un paquete del sistema.

Uso: `sudo apt remove vim`



NB: Te doy el comando para listar los paquetes actualmente instalados en la siguiente línea.

• APT SEARCH

Muy útil para encontrar el nombre exacto del paquete antes de instalarlo.

Uso básico: `apt search openjdk`
Con `grep`: `apt search openjdk | grep jre`



NB: No necesitas `sudo` para esto.

• INSTALACIÓN MANUAL

A veces, es necesario instalar los paquetes manualmente, si el editor no proporciona un repositorio.

Descargue el archivo con `wget`:

`wget https://www.realvnc.com/download/file/viewer.files/VNC-Viewer-6.19.325-Linux-ARM.deb`

Instalación manual:

`sudo dpkg -i VNC-Viewer-6.19.325-Linux-ARM.deb`



NB: Puede utilizar `dpkg -r` para eliminar un paquete manualmente, o `dpkg-reconfigure` para rehacer la configuración tras la instalación.

• LISTA PAQUETES INSTALADOS

`Dpkg` también puede ser útil para listar los paquetes actualmente instalados.

Sintaxis: `dpkg -l`
Con `grep`: `dpkg -l | grep php`

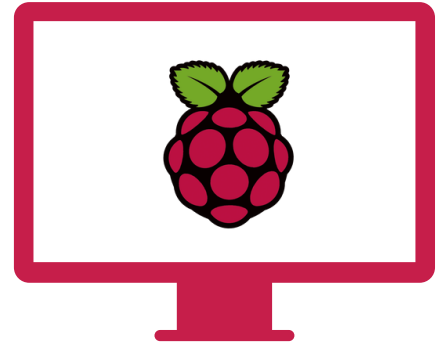


NB: Lea la salida `"man dpkg"` para obtener todas las opciones posibles de este comando.



GESTIÓN DEL SISTEMA

Ahora que tienes todos los paquetes instalados, puede que necesites aprender comandos más avanzados sobre cómo gestionar tu sistema operativo Raspberry Pi



• REINICIAR

Este comando reiniciará tu Raspberry Pi inmediatamente

Uso: `sudo reboot`

• SERVICE

Cada demonio tiene un servicio asociado, puedes iniciarlo o pararlo cuando quieras

Inicio: `sudo service apache2 start`
Parar: `sudo service apache2 stop`
Reiniciar: `sudo service apache2 start`
Recargar config: `sudo service apache2 reload`



NB: Utilice "service <service>" para listar todas las opciones disponibles, por ejemplo "service apache2"
El tabulador le ayudará a encontrar el nombre del servicio

• LISTA DE PROCESOS

Muestra todos los procesos en ejecución

Uso básico: `ps aux`
Sólo por un usuario concreto: `ps -u pi`



NB: Te doy el comando para listar los paquetes actualmente instalados en la siguiente línea

• HTOP

Una gran alternativa a top, para visualizar la carga y el proceso del sistema en una interfaz intuitiva

Uso: `htop`



NB: Si htop no está instalado por defecto, instálelo con "apt install htop"

• APAGAR

Detiene el Raspberry Pi, ahora o en un momento específico

Para ahora: `sudo shutdown -h now`
En un momento determinado: `sudo shutdown -h 20:00`

• COMENZAR EN BOOT

La mayoría de las veces, los servicios se inician automáticamente al arrancar, pero si es necesario puede hacerlo manualmente

Inicio en el arranque: `sudo update-rc.d ssh enable`
No arranques al iniciar: `sudo update-rc.d -f ssh remove`



NB: Para iniciar un script en el arranque, añádalo al archivo /etc/rc.local

• KILL / KILLALL

Detener inmediatamente un proceso específico o todos los procesos desde el mismo comando

Kill: `kill 12345`
Killall: `killall php`



NB: Utilice el comando ps para encontrar el ID del proceso a matar

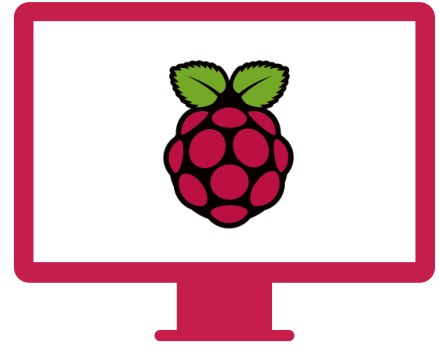
• DF

Muestra la lista de particiones, una buena forma de comprobar el espacio restante en disco

Uso básico: `df`
Más legible: `df -h`
Partición específica: `df -h /media/usb`



GESTIÓN DEL SISTEMA (2)



• DU

Muestra el espacio de disco utilizado en la carpeta actual o en la especificada.

Uso básico: `du`
Carpeta específica: `du /home/pi`
Resumir: `du --summarize /home/pi`
20 más grandes: `du -ak | sort -nr | head -20`



NB: Hay muchas más opciones, consulte el "man du" para encontrar más ayuda sobre este tema.

• DATE

Como su nombre indica, muestra la fecha y hora actuales

Salida completa: `date`
Formato específico: `date +%m-%d-%Y`



NB: El comando "man date" le proporciona la lista de todas las opciones disponibles y el formato

• CHOWN

Cambia el propietario y el grupo del archivo

Cambiar el propietario: `sudo chown pi /usr/local/bin/script.sh`
Cambiar propietario y grupo de archivos: `sudo chown pi:www-data /var/www/html/mysite`

• CPU TEMPERATURE

No es un comando fácil de recordar, pero resulta muy útil cuando se hace overclocking o se ejecutan aplicaciones que consumen mucha energía.

Uso: `vcgencmd measure_temp`



NB: `vcgencmd` está oculto en el paquete `libraspberrypi-bin`, puede que necesites instalarlo manualmente en Raspberry Pi OS lite:
`"sudo apt install libraspberrypi-bin"`

• MOUNT

Montar una nueva partición (llave usb por ejemplo)

Montar disco: `sudo mount /dev/sda1 /mnt/usb`
Desmontar: `sudo umount /mnt/usb`



NB: Es un comando complejo para principiantes, pero este post te dará toda la información necesaria
<https://raspberrytips.es/montar-unidad-usb-raspberry-pi/>

• UPTIME

Muestra el tiempo de actividad actual de la Raspberry Pi (cuánto tiempo encendida).

Uso básico: `uptime`
Última fecha de arranque: `uptime -s`

• CHMOD

Cambia los permisos de archivos o carpetas

Permisos dígitos: `chmod 644 script.sh`
Letras permisos: `chmod +x script.sh`



NB: `Chmod` es un comando complejo para principiantes, puedes consultar esta herramienta para saber como leer y establecer permisos correctamente:
<https://chmod-calculator.com/>

• MAN <COMMAND>

Ya lo he dado muchas veces en este documento, pero `man` permite encontrar ayuda sobre un comando

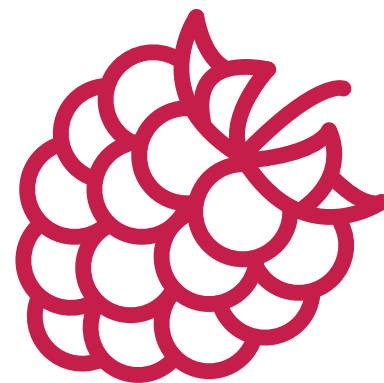
Ejemplo: `man find`



NB: Pulse la barra espaciadora para pasar a la página siguiente y "q" para salir.



RASPBERRY PI OS COMMANDOS



Como sistema operativo similar a Debian, RPI OS usa la mayoría de los mismos comandos
Pero aquí encontrará las órdenes específicas de RPI OS



Nota:

Hay algunos comandos que sólo funciona en Raspberry Pi OS
No son esenciales para usar una Raspberry Pi (excepto el primero probablemente)
Pero en la mayoría de los sitios web no los encontrarás ya que no están presentes en otras distribuciones de Linux

• RASPI-CONFIG

Esta es la herramienta principal para configurar tu Raspberry Pi desde un terminal

Uso: `sudo raspi-config`



NB: Raspi-config te permite muchos cambios en la configuración de tu Raspberry Pi, como contraseña, opciones de red, opciones de arranque, opciones de localización, opciones de interfaz (ssh), overclocking y otras opciones avanzadas.

• LIBCAMERA-VID

Es lo mismo pero para capturar vídeo desde tu cámara

Uso básico: `libcamera-vid -o video.h264 -t 10000`



NB: -t opción es para el tiempo que desea capturar el vídeo

• RPI-UPDATE

Ya vimos este comando en la sección de actualizaciones del sistema, actualizará todo en tu sistema

Uso: `sudo rpi-update`

• LIBCAMERA-STILL

Este comando le permite tomar una foto de la cámara Raspberry Pi

Uso básico: `libcamera-still -o image.jpg`



NB: En este post encontrarás toda la información necesaria sobre cómo utilizar tu cámara:

<https://raspberrytips.es/solucion-problemas-camara/>

• RASPI-GPIO

Establece u obtén valores de tus pines GPIO en un terminal

Obtener valor: `sudo raspi-gpio get`

Establecer valor: `sudo raspi-gpio set 20 a5`



VARIOS COMMANDOS

En esta parte, quería darle todos los otros comandos útiles que no encajan en los demás



● HISTORY

Linux almacena cualquier comando que escribas en un fichero de archivo, puedes leerlo con "history"

Todos los comandos: `history`
Últimos 20: `history | tail -n 20`
Borrar todo el historial: `history -c`
Borrar una línea: `history -d 123`

● |

Ya te mostré la tubería en muchos ejemplos, te permite combinar múltiples comandos para encontrar exactamente lo que quieres

Sintaxis: `<comando1> | <comando2>`
Grep ejemplo: `cat test.txt | grep error`
Doble: `du -ak | sort -nr | head -20`

● !

Ejecutar un comando específico del historial

Sintaxis: `!<history_id>`
Ejemplo: `!123`



NB: El ID del historial cambia con cada nuevo comando que escriba (incluido !), asegúrese de utilizarlo sólo una vez o compruebe de nuevo el ID

● >

Crear un archivo para almacenar la salida del comando

Sintaxis: `<comando> > <archivo>`
Ejemplo: `cat test.txt | grep error > error.log`



NB: El último comando pone todas las líneas que contienen "error" en el archivo test.txt
Este comando no muestra nada

● CRONTAB

Le permite programar tareas en su Raspberry Pi

Listar las tareas actuales: `crontab -l`
Editar tareas: `crontab -e`



NB: The crontab syntax is a tough to understand for beginners, use this tool to check your line is correct:
<https://crontab.guru/>

● SCREEN

Ejecutar un terminal virtual, para permitir que una sesión que se ejecuta en segundo plano

Inicio: `screen -S <nombre>`
Salir: `CTRL+A CTRL+D`
Reanudar: `screen -r <nombre>`
Parar: `CTRL+D`

● !!

Similar a ! pero para volver a ejecutar el último comando

Uso: `!!`



NB: Puede ser útil para ejecutar varias veces los mismos comandos complejos

● >>

Añadir la salida del comando al final de un archivo

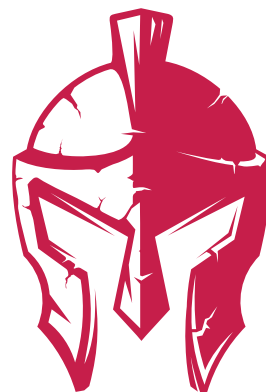
Uso: `cat test.txt | grep error >> error.log`



NB: Es el mismo uso que >
Pero en este caso, añadirá las líneas al archivo error.log, y mantendrá el principio como estaba



COMANDOS PARA LOS GUERREROS



Y por último, ahora que ya eres un experto con el terminal, vamos a ver algunos comandos complicados para poner a prueba tus límites :) Pueden ser difíciles de usar, con muchas opciones, o difíciles de analizar

• AWK

Awk se asemeja a un lenguaje de programación. Permite buscar cadenas y transformarlas para mostrarlas de otra forma.

Sintaxis: `awk [-F] [-v var=value] 'program' file`
Ejemplo básico: `awk -F":" '{print $1}' /etc/passwd`



NB: El último comando muestra sólo la primera columna. No puedo explicarte el uso de awk en detalle en unas pocas líneas. Consulta esta guía para aprender más sobre esto: <https://do.co/2VC8mmn>

• CUT

Otra forma de transformar texto en una línea de comandos, probablemente más fácil de entender.

Sintaxis: `cut <opcion> <archivo>`
Ejemplo básico: `cut -d : -f 1 /etc/passwd`



NB: -d establece el delimitador a utilizar, y -f el campo a conservar. Utilice "man cut" para obtener más información sobre otras opciones.

• LSOF

Significa "LiSt Open Files", muestra todos los archivos abiertos actualmente en su Raspberry Pi.

Uso: `lsuf`



NB: Utiliza grep con una tubería para encontrar el archivo que buscas.

• NETSTAT

Supervisa tu actividad en la red.

Puertos de escucha: `netstat -l`
Añade el ID del proceso: `netstat -lp`
Lo mismo en tiempo real: `netstat -lpc`



NB: Hay muchas otras opciones para netstat, puedes consultar la página "man netstat" para saber más.

• SED

Similar a awk, pero sólo para expresiones regulares.

Sintaxis: `sed <opcion> <codigo> <archivo>`
Ejemplo básico: `sed '/^#/d' /etc/apache2/apache2.conf`



NB: El último comando elimina los comentarios de la configuración. En cuanto a awk, necesitarás serios tutoriales y experiencia para dominarlo.

• WC

WC son las siglas de "Words Count" (recuento de palabras).

Sintaxis: `wc <opcion> <archivo>`
Recuento de líneas: `wc -l /var/log/syslog`



NB: -l es para líneas, -w para palabras y -m para caracteres. También se puede utilizar después de una tubería (para contar las líneas de un comando grep, por ejemplo).

• WATCH

Supervisa la salida de un comando, ejecutándolo en cada intervalo especificado.

Uso básico: `watch date`
Tiempo específico: `watch -n10 date`



NB: El tiempo de actualización por defecto es de 2s.

• DMSG

Muestra un archivo de registro de todos los eventos ocurridos en la última secuencia de arranque.

Uso: `dmesg`



NB: La mayoría son normales. Puedes usar grep para buscar errores o algo específico.

Gracias por su lectura!
Hasta pronto en RaspberryTips
Patrick

