

LICENCIA

copyright (c) 2007 Mainer Likona Este documento está licenciado con la GPL. Usted puede usar, copiar, mejorar y distribuir este documento bajo los términos de esta licencia. Una copia de esta licencia se puede bajar de <http://www.gnu.org/licenses/gpl.txt>

Introducción a la línea de comandos

Que es la línea de comandos?

La línea de comandos es una forma de acceder al sistema para realizar las tareas diarias sin utilizar interfaces gráficas. En vez de utilizar el ratón para ejecutar programas pinchando en su icono, en la terminal hemos de escribir los comandos que queremos ejecutar. Hay distintas terminales virtuales en el menú: En "Aplicaciones->herramientas del sistema->consola", "Aplicaciones->Debian->Consolas X". La consola es la herramienta por excelencia del administrador de sistemas, ya que permite controlar de manera más precisa el comportamiento de la máquina. También permite ejecutar muchas más tareas que las que permiten las herramientas gráficas, y da la oportunidad de automatizar tareas, permitiendo que se ejecuten distintos programas enlazados entre sí sin tener que estar presente.

Ejemplos

Quiero coger de un fichero ciertas entradas y escribirlas en otro fichero. Quiero poner a bajar algo en el emulador de casa y estoy en el trabajo. Algo ha ralentizado el sistema (estoy intentando abrir 200 correos a la vez por error) y quiero matar el programa, pero sin tener que reiniciar.

Cuando trabajamos con la terminal, hemos de escribir los nombres de los comandos y los ficheros. Para ayudarnos en esta tarea podemos utilizar la tecla tabulador. Si estamos escribiendo un comando y tecleamos tabulador, nos completará su nombre. En caso de que haya dos o más comandos con el mismo principio, completará hasta donde pueda. Si tecleamos dos veces tabulador, se nos mostrará en pantalla todos los comandos que empiecen de esa manera.

Como funciona la línea de comandos

Para ejecutar un comando, solamente tenemos que escribir su nombre.

Ejemplos

```
iceweasel
oowriter
```

A los comandos se les pueden pasar opciones y ficheros. Si tecleamos "oowriter prueba.odt" se nos abrirá el documento en el OpenOffice.

Podemos pasarle a los comandos opciones que alteran su comportamiento. Las opciones serían el equivalente de elegir una opción de menú dentro de un programa. Las opciones pueden tener un formato largo y uno abreviado. Por regla general, el formato abreviado consta de un guión y una letra (-h, -l) y el formato largo consta de dos guiones juntos y una palabra, generalmente en inglés (--help, --verbose). Si queremos pasar más de una opción, las podemos agrupar. Ejemplos:

```
netstat -ei
netstat --help
```

La ortografía es vital. Después del nombre del comando hay que poner un espacio, después van las opciones, otro espacio y los ficheros separados por espacios.

En resumen: La estructura general de un comando es esta: $\text{Comando} \begin{matrix} _ \\ -h \\ -v \\ -l \end{matrix} _ \text{ fichero}$

Documentación en GNU/Linux

La mayoría de los comandos de GNU/Linux vienen con instrucciones de uso conocidas como páginas de manual. Para poder leerlos tenemos el comando ``man nombrePrograma``.

Las páginas de manual tienen la misma estructura:

- Al principio vemos un resumen del comando y todas las opciones disponibles.
- Luego tenemos una descripción de las opciones.
- Al final suele haber ejemplos.

Para buscar en una pagina de manual, teclearemos ``/cosaquebusco``

Para buscar un comando del que no nos acordamos del nombre, podemos usar ``apropos`` Ejemplo: Se que existe un comando para configurar la wifi pero no me acuerdo del nombre. Como lo busco?

```
$apropos wireless
iwconfig (8)      - configure a wireless network interface
iwevent (8)       - Display Wireless Events generated by drivers and setti...
iwgetid (8)       - Report ESSID, NWID or AP/Cell Address of wireless network
iwlist (8)        - Get more detailed wireless information from a wireless...
iwpriv (8)        - configure optionals (private) parameters of a wireless...
iwspy (8)         - Get wireless statistics from specific nodes
wireless (7)      - Wireless Tools and Wireless Extensions
```

Gestor de arranque de sistemas operativos: Grub

El GRand Unified Bootloader (GRUB) es un gestor de arranque múltiple que se usa comúnmente para iniciar dos o más sistemas operativos instalados en un mismo ordenador.

Una de las características más interesantes es que no es necesario instalar una partición nueva o un núcleo nuevo, pudiendo cambiar todos los parámetros en el arranque mediante la Consola de GRUB.

Mientras los gestores de arranque convencionales tienen una tabla de bloques en el disco duro, GRUB es capaz de examinar el sistema de archivos. Actualmente, soporta los siguientes sistemas de archivos:

- ext2/ext3 usado por los sistemas UNIX y su variante GNU/Linux
- ReiserFS.
- XFS de SGI.
- UFS.
- VFAT, como FAT16 y FAT32 usados por Windows 9.x
- NTFS usado por los sistemas Windows NT/2000/XP/Vista
- JFS de IBM.

Hoy en día Grub2 está reemplazando a Grub, que pasa a llamarse Grub legacy. Grub2 trae muchas mejoras, por lo que merece la pena actualizar.

Comandos basicos

Fichero de configuracion: `/boot/grub/menu.lst`

- `update-grub` #Actualiza la lista de sistemas operativos arrancables. Al instalar kernels nuevos se ejecuta automaticamente.
- `grub-install(hd0,0)` #Instala el gestor de arranque en la primera particion del primer disco. Util para recuperar desastres.

Parametros básicos de configuración

Los parámetros básicos son:

```
# kopt=root=/dev/sda1 ro
```

Opción que indica donde está la particion con los kernels.

```
# groot=(hd0,0)
```

Opción para indicar donde esta Grub. Se indica a la manera de BSD, (disco, partición). (hd0,0) indica la primera particion (0) del primer disco en el bus (hd0)

Estos tres parámetros son lo mínimo que necesita. Otros parametros interesantes:

```
default          0
```

Indica el sistema que se arrancará por defecto.

```
timeout          10
```

Indica cuanto tiempo espera antes de arrancar el sistema por defecto.

Como arrancar un GNU/Linux?

```
title             Debian GNU/Linux, kernel 2.6.22-3-686          #un nombre cualquiera
root              (hd0,0)                                       #primera partición del primer disco. Poner la que nos correspon
```

```
kernel      /boot/vmlinuz-2.6.22-3-686 root=/dev/sda1 ro vga=791 splash=silent      #el kernel
initrd      /boot/initrd.img-2.6.22-3-686                               # initrd necesario para arrancar el kernel
```

Como arrancar un Windows?

Para arrancar windows hay que pasar los siguientes parámetros:

```
title      Güindous # un nombre cualquiera
root       (hd0,0)  # primera partición del primer disco. Poner la que nos corresponda
makeactive # parametro para pasar el control a otra particion
chainloader +1    # pasar el arranque a otra particion
```

Como meter un dibujo?

En grub legacy los dibujos tienen que cumplir estas condiciones:

- Tamaño: 640x480
- Colores: 14
- Formato: xpm.gz

Una vez que tenemos una imagen de estas características, la copiamos a /boot/grub y añadimos esta línea:

```
`` splashimage = (hd0,1)/boot/grub/inicio.xpm.gz ``
```

En el momento del arranque podemos acceder a la consola de grub para pasar los parámetros a mano. Muy útil si nos hemos confundido en la configuración y el sistema no arranca.

Grub 2

Grub ha sido reescrito desde cero en esta nueva versión. Ha habido muchos cambios:

Servicios en GNU/Linux

Dentro de las aplicaciones que podemos instalar en cualquier distribución, encontramos aplicaciones de usuario y servicios, como podría ser un servidor web, un servidor de bases de datos, de correo etc.

El proceso de instalación es igual al de cualquier aplicación de Ubuntu, pero hay que conocer ciertos aspectos de su funcionamiento:

- Podemos ponerlos en marcha mediante los scripts de /etc/init.d nombreservicio start. Si pasamos la opción stop, el servicio se detiene y con restart, reinicia.
- Podemos definir modos de ejecución en los que se arranquen o se paren servicios automáticamente
- Podemos ver sus registros de error en /var/log/nombreServicio o en los registros de error genéricos /var/log/syslog y /var/log/messages.
- Los ficheros de configuración suelen estar en /etc/nombreServicio
- El fichero que nos da la información necesaria para su postinstalación está en /usr/share/doc/nombreServicio/README.Debian. Si este fichero existe, conviene leerlo, ya que nos informa exactamente de los primeros pasos a dar.

Estructura de paquetes APT

El APT es un conjunto de librerías que gestionan los paquetes que están instalados y sus dependencias. Este sistema se usa tanto en Debian como en Ubuntu. En el sistema existe una base de datos con la información sobre los paquetes disponibles e instalados. Esta base de datos se actualiza con la información de los repositorios, que veremos más abajo. Dispone de distintas herramientas.

Los ficheros principales son:

/etc/apt/sources.list	Aquí especificamos los repositorios.
/var/cache/apt/archives	Aquí se guardan los paquetes .deb
/etc/apt/apt.conf.d	Directorio donde se guardan las configuraciones.

Los comandos principales son:

dpkg	Herramienta para instalar .deb "a pelo"
apt-get	Antigua herramienta para instalar y desinstalar paquetes
aptitude	Herramienta para interactuar con el APT equivalente a Synaptic

Dpkg instala o desinstala paquetes, pero no controla que esten instaladas sus dependencias, ni si el paquete esta roto o no. Apt-get y aptitude utilizan dpkg para instalar y desinstalar, pero ademas llevan un control de dependencias, aunque aptitude es mas eficiente.

Para instalar una aplicacion, simplemente escribimos: ``aptitude install aplicacion``

Para actualizar la lista de paquetes ``aptitude update``

Para borrar un paquete ``aptitude delete paquete``

Para hacer busquedas ``aptitude search kde``

Para limpiar la cache de paquetes ``aptitude clean``

Para instalar un conjunto de paquetes .deb: ``dpkg -i *.deb``

Para volver a configurar el entorno grafico ``dpkg-reconfigure xserver-xorg``

Para ver que ficheros ha instalado un paquete ``dpkg -L paquete``

Ficheros de configuracion

En Ubuntu los paquetes se dividen en cuatro categorias:

main	paquetes libres con soporte oficial
restricted	paquetes no libres pero soportados
universe	paquetes libres sin soporte oficial
multiverse	paquetes no libres

Introducción a redes en GNU/Linux

Para configurar redes en Ubuntu|Debian GNU/Linux tenemos los siguientes ficheros. Para escribir en ellos hay que seguir una sintaxis concreta que se explica en el man o para los amantes del copy&paste, en Google, buscando el nombre del fichero.

/etc/network/interfaces	Aqui se configuran las interfaces de red.
/etc/hosts	Aqui se asignan nombres a las maquinas. Antecesor de DNS
/etc/hostname	Aqui donde se define el nombre de nuestro ordenador
/etc/resolv.conf	Aqui se especifican los servidores DNS.

Para configurar redes en cualquier GNU/Linux disponemos de los siguientes comandos:

ifconfig	Comando basico para cambiar IPs, mascaras, etc.
route	Lo mismo pero con gateways.
ipmasq	Si lo instalamos, nos configura el firewall para funcionar como router de manera segura.
netstat	Nos muestra informacion sobre las conexiones UDP/TCP, info sobre puertos...

Existe un comando especial para configurar wifis, a veces hay que instalar el paquete wireless-tools para usarlo:

`iwconfig` Asigna el essid, la clave, el ap etc a una interfaz wifi

Configuración básica de una red: ifconfig y route

En un sistema GNU/Linux encontraremos la interfaz lo, o loopback. Esta interfaz siempre tiene la IP 127.0.0.1.

Para configurar una red lo primero de todo asignarnos una IP. La manera básica de hacer esto es por medio de ifconfig.

Así asignamos la IP 192.168.0.12 a la primera interfaz de red ``ifconfig eth0 192.168.0.12 up``

Así desactivamos la primera interfaz wifi ``ifconfig wlan0 down``

Si nos sale una entrada como esta, la interfaz está bien configurada:

```
PING 192.168.0.12 (192.168.0.12) 56(84) bytes of data.  
64 bytes from 192.168.0.12: icmp_seq=1 ttl=64 time=0.082 ms  
64 bytes from 192.168.0.12: icmp_seq=2 ttl=64 time=0.074 ms
```

En estos momentos podemos conectarnos a otras máquinas con el rango de IPs 192.168.0.xxx, pero no podemos conectarnos a ninguna más. Para ello necesitamos una gateway.

Para configurar una gateway por defecto: ``#route add default gw 192.168.0.1``

Para ver si lo hemos hecho bien, ejecutamos ping: ``$ping 192.168.0.12``

Si nos aparece el mensaje "Network is unreachable" no se ha configurado algo bien. Hay que comprobar que el cable esté bien enchufado, y que la tarjeta funcione.

Para borrar la entrada por defecto de la tabla de rutas: ``#route del default``

Para borrar una entrada de la tabla de rutas: ``# route del -net 192.168.0.0 gw 0.0.0.0 netmask 255.255.255.0``

Ejemplo de una tabla de rutas de un ordenador con una sola interfaz de red:

```
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
0.0.0.0 192.168.0.1 0.0.0.0 UG 0 0 0 eth0
```

La primera entrada nos indica que los paquetes de la red 192.168.0.0 ya han llegado a la red destino y se mandan directamente por ethernet.

La segunda entrada es la entrada por defecto, que nos indica que cualquier paquete que no cumpla las reglas anteriores sea enviado a 192.168.0.1 por la interfaz eth0.

Ejemplo de una tabla de rutas de un ordenador que hace de gateway de la red 192.168.0.0

```
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
62.99.56.0 0.0.0.0 255.255.248.0 U 0 0 0 eth1
0.0.0.0 62.99.56.1 0.0.0.0 UG 0 0 0 eth1
```

La primera entrada es igual que la anterior, y quiere decir lo mismo. La segunda entrada indica que los paquetes dirigidos a la red 62.99.56.0 ya han llegado a su destino y que se encaminan por eth1. La tercera entrada, al igual que en el caso anterior, es la entrada por defecto y indica que todos los paquetes que no cumplan las reglas anteriores se envíen al 62.99.56.1 por la interfaz eth1.

Para cambiar la MAC de un ordenador:

```
#ifconfig eth0 hw FA:BA:DA:FA:BA:DA
```

Para configurar una wifi el comando básico es iwconfig. La configuración básica es esta:

```
iwconfig ath0 essid mai channel 5 ap 00:99:88:77:66:66
```

Para asignarle una clave wep

```
iwconfig ath0 key s:elnombredelaclave
```

Para poner una clave wpa hay que configurar el fichero `/etc/wpa_supplicant.conf`. Para poner la clave

```
ctrl_interface=DIR=/var/run/wpa_supplicant
network={
    ssid="mai"
    scan_ssid=1
    key_mgmt=WPA-PSK
    psk="password"
}
```

Instalación de drivers manual

Hay casos en los que la única manera de instalar el programa es bajarse el código fuente e instalarlo. En esos casos, el procedimiento es el siguiente:

- Bajarse el código fuente, generalmente un paquete .tar.gz
- Descomprimirlo mediante `tar xzfv paquete.tar.gz`
- Entrar en la carpeta que se ha creado.
- Buscar un fichero README o similares y leerlo
- Ejecutar `make`
- Ejecutar `make install`. Este comando necesita permisos de root.

El procedimiento puede variar de un caso a otro, ya que no existe ninguna norma escrita. El README suele traer instrucciones concretas para cada caso, por lo que en caso de problemas, es el primer lugar a visitar.

Drivers en GNU/Linux

Lo primero es saber exactamente que hardware tenemos. Si tenemos suerte el kernel ya soporta el dispositivo y no necesitamos cargar ningún driver. En caso contrario habrá que seguir estos pasos:

- Teclearemos lspci o lsusb. Por cada elemento enchufado a la placa nos saldrá una línea con su identificador.
- Buscaremos el identificador en el apt mediante aptitude search XXX o utilizando el synaptic. Si aparece el driver, lo instalamos.
- Si no aparece probaremos a instalar module-assistant y mirar si existe allí el modelo.
- Si no aparece consultaremos en Internet, buscando el identificador y la palabra "linux" o "ubuntu" para ahorrarnos problemas de windows.

Instalación de drivers mediante Synaptic

Para instalar drivers mediante Synaptic hay que seguir el mismo procedimiento que para instalar cualquier programa. También podemos instalar una versión de kernel mas nueva que la que estamos usando, ya que las versiones mas modernas traen mayor soporte de hardware. Para instalar un kernel hay que buscar "linux-image". En el caso de un driver concreto, hay que buscar el numero del chips que nos muestra lspci, o la combinacion de dos grupos de cuatro caracteres (a4c0:fada) que nos muestra lsusb.

Instalación de drivers mediante module-assistant

El programa module-assistant sirve para ayudarnos en la instalación de drivers que por sus características no son libres, y no pueden ser instalados directamente mediante synaptic/aptitude. Este programa hay que instalarlo desde synaptic/aptitude, y ejecutarlo desde la terminal. Pasos a seguir para instalar un programa:

- Abrir una terminal como root.
- Teclear m-a
- En el menú que se nos muestra, elegiremos update
- Elegimos prepare. Nos pedirá que instalemos algunos programas, le diremos que si.
- Elegimos select. Se nos muestra una lista de drivers. Elegimos el que queremos mediante la barra espaciadora.
- Elegimos Get
- Elegimos Build
- Elegimos Install

Con esto ya tenemos instalado el driver, así que deberíamos de tener el dispositivo en marcha.

Compilar el kernel de linux

Compilar el kernel sigue los mismos pasos que al compilar cualquier programa. Pero teniendo en cuenta que el kernel es un programa especial con muchas opciones de configuracion, existen maneras mas faciles de compilarlo.

¿Por qué compilar un kernel a la manera Debian?

El hecho de compilarlo a la manera debian nos proporciona ciertas ventajas que compilarlo a la tradicional. Cuando a la 'debian way' este se empaqueta automáticamente en un archivo que tiene extensión .deb, es decir, se crea un paquete. El simple hecho de que se genere un paquete con nuestro kernel compilado nos proporciona las siguientes ventajas:

- Cuando tengamos que eliminar el kernel debido a que probablemente ya tengamos uno nuevo, lo único que tenemos que hacer es eliminar el kernel viejo desde Synaptic o aptitude.
- Al poder tener el kernel en un paquete se no hace muy fácil guardarlo en un Pendrive, CD, DVD, o mejor aún, si tenemos un servidor web podríamos subirlo para que otros se descarguen el paquete con el kernel y utilicen nuestro propio kernel. Comúnmente el objetivo de esto es poder instalar ese mismo kernel en varias máquinas. Por ejemplo, si tienen 20 máquinas en su oficina y van a actualizar el kernel a todas, sería muy absurdo compilar el kernel en para cada una de las máquinas, pero sería inteligente y astuto utilizar herramientas de Debian para poder crear un kernel genérico, el cual vamos a poder distribuirlo en todas las máquinas.

Pasos para compilar

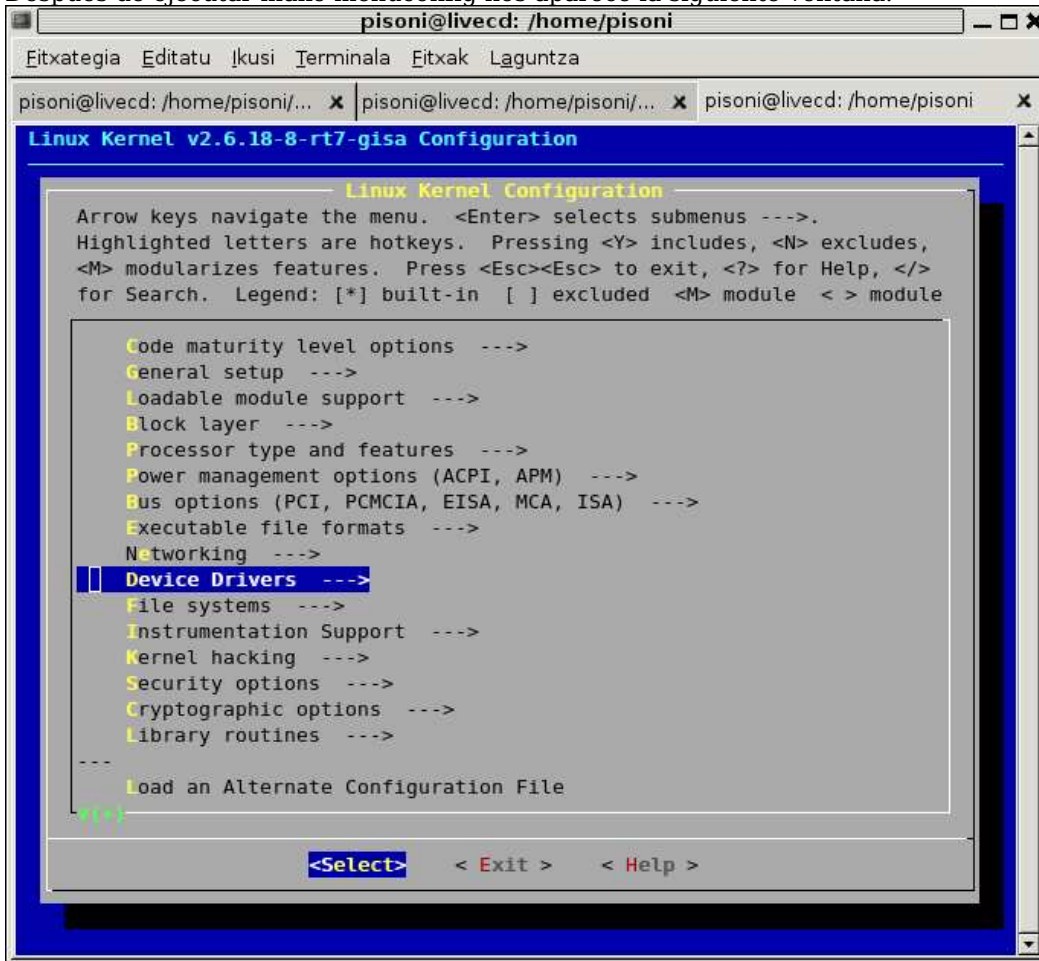
Para instalar un nuevo Kernel es necesario descargar el código fuente del mismo en la página oficial que es donde se encuentran todos los kernel de Linux, desde la primera versión hasta la última. También podemos instalar el paquete linux-source Para poder compilar a la debian necesitamos instalar el paquete kernel-package: aptitude install kernel-package.

Lo común es que las fuentes del kernel esten en /usr/src pero si deseas hacerlo en otra parte posees la libertad ;-).

Para compilar el kernel hay que seguir estos pasos:

cd /usr/src/linuxETC.	Entramos en el directorio donde esta el codigo
make mproper	Limpiamos el entorno (no necesario la 1 vez)
make menuconfig	Entramos al menu de configuracion. Si da error: aptitude install libncurses5-dev
make xconfig	Lo mismo que el anterior pero con ventanas. Si da error: aptitude install libqt3-dev
Configuramos el kernel	Lo vemos abajo
make-kpkg clean	Limpiamos el entorno
make-kpkg --initrd kernel_image kernel_headers --append-to-version elmio	Generamos el kernel
aptitude install *elmio*	Instala el kernel nuevo

Despues de ejecutar make menuconfig nos aparece la siguiente ventana:



Aqui tenemos todas las opciones en un perfecto ingles. Para movernos utilizamos los cursores. Las entradas que tienen este simbolo ``-->`` tienen subcategorias dentro. Para entrar en una categoria usamos el enter y para salir escape.

Podemos activar el modulo para que se quede metido en el kernel (*) o que se cargue unicamente cuando haga falta (M). Para elegir nos pondremos encima de la opcion y le daremos a la barra espaciadora.

Las opciones que mas nos interesan estan en Networking y device drivers. Aqui se pueden activar los drivers que soporta el kernel.