



Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores | Bergantells Usuaris de GNU/Linux de Mallorca i Afegitons

Utilizar un lector de tarjetas flash (MMC/SD) bajo Linux (65129 lectures)

Per **Paco Ros**, [Paco](http://pacoros.wordpress.com) (<http://pacoros.wordpress.com>)

Creado el 11/10/2003 17:55 modificado el 11/10/2003 17:55

Últimamente se han puesto de moda estos dispositivos portátiles conocidos como "discos duros portátiles" o "llaveros USB" que nos permiten trasladar cómodamente la información de un lugar a otro.

Existe una variante de ellos que tiene la misma estructura, pero que funciona utilizando tarjetas de memoria (Compact Flash, Smart Media, Multimedia Card, Secure Digital...).

Este truco se centra en la utilización de los segundos, aunque todo lo aquí expuesto es perfectamente válido para los primeros. Vamos a configurar el dispositivo para que funcione bajo Linux.

Este tipo de dispositivos se han popularizado como sustitutivo del disquete. Aunque son muchos y variados los dispositivos que cuentan con un lector interno compatible y que funciona exactamente igual

En mi caso, he adquirido un lector de tarjetas MMC y SD en unos conocidos grandes almacenes. El precio del dispositivo es de 18 euros y el precio de las tarjetas MMC de 128 MB ronda los 45 euros. Tenéis unas imágenes del adaptador [aquí](#)⁽¹⁾ y [aquí](#)⁽²⁾ (evidentemente la tarjeta MMC no venía incluida).

Una solución económica para transportar cantidades importantes de datos entre máquinas y una manera excelente de poder utilizar dispositivos portátiles como cámaras digitales, agendas electrónicas, reproductores MP3, etc... sin tener que preocuparnos por la compatibilidad del dispositivo con Linux.

Vamos a centrarnos ya en configurar el lector bajo Linux.

Lo primero que hay que hacer es asegurarse de el kernel dispone de todo lo necesario:

SCSI Support (CONFIG_SCSI, scsi.o)

SCSI disk support (CONFIG_BLK_DEV_SD, sd.o, sd_mod.o?)

SCSI generic support (CONFIG_CHR_DEV_SG, sg.o)

Support for USB (CONFIG_USB, usb.o)

Preliminary USB device file system (CONFIG_USB_DEVICEFS, devices.o?)

Soporte USB: EHCI HCD (CONFIG_USB_EHCI_HCD, usb-ehci-hcd.o), UHCI (CONFIG_USB_UHCI, usb-uhci.o), OHCI (CONFIG_USB_OHCI, usb-ohci.o)

USB Mass Storage support (CONFIG_USB_STORAGE, usb-storage.o)

Si no tenemos alguno de esos en el kernel, tocará recompilar (podéis averiguar si vuestro kernel tiene lo necesario usando `modprobe -l`) aunque en las distribuciones modernas los "stock" kernels ya llevan incluidas todas esas opciones.

Después tenemos que asegurarnos de que disponemos de HOTPLUG. No es imprescindible, pero sí muy recomendable. Todas las distribuciones modernas cuentan con él y, por si alguien no lo tiene aún que busque el paquete `hotplug` en su distribución y lo instale.



Si lo hemos hecho todo bien podemos conectar el dispositivo y este aparecerá en `/proc/scsi/scsi` o haciendo un `cdrecord -scanbus`:

```
wanda:/mnt# cat /proc/scsi/scsi
Attached devices:
(...)
Vendor: Generic  Model: USB SD Reader    Rev: 1.00
Type:   Direct-Access                    ANSI SCSI revision: 02

wanda:/mnt# cdrecord -scanbus
Cdrecord-Clone 2.01a19 (i686-pc-linux-gnu) Copyright (C) 1995-2003 Jörg Schilling
Linux sg driver version: 3.1.25
Using libscg version 'schily-0.7'
(...)
scsibus1:
1,0,0 100) 'Generic ' 'USB SD Reader  ' '1.00' Removable Disk
(...)
```

Ya lo tenemos montado. Uno de los grandes problemas con los que me he encontrado ha sido indetificar del dispositivo de bloques de `/dev` que se iba a utilizar para montar el dispositivo. Para ello he utilizado el paquete `sg3-utils` que sirve para controlar varios parámetros de los dispositivos SCSI (`apt-get install sg3-utils`) y ejecutando:

```
wanda:/dev# sg_scan -i
/dev/sg0: scsi0 channel=0 id=0 lun=0 [em] type=5
LG      CD-ROM CRD-8322B 1.02 [wide=0 sync=0 cmdq=0 sftre=0 pq=0x0]
/dev/sg1: scsi0 channel=0 id=1 lun=0 [em] type=5
SONY    CD-RW  CRX140E 1.0n [wide=0 sync=0 cmdq=0 sftre=0 pq=0x0]
/dev/sg2: scsi1 channel=0 id=0 lun=0 [em] type=0
Generic USB SD Reader 1.00 [wide=0 sync=0 cmdq=0 sftre=0 pq=0x0]
```

...y posteriormente...

```
wanda:/dev# sg_map
/dev/sg0 /dev/scd0
/dev/sg1 /dev/scd1
/dev/sg2 /dev/sda
```

Podemos averiguar rápidamente dónde está el dispositivo a montar. En mi caso, se encuentra en `/dev/sda`.

Ahora viene otra parte divertida del asunto. Si nuestra tarjeta flash es nueva, no habrá problema. Ejecutamos `fdisk` (mirad la ayuda una vez abierto el programa que sale con la letra "m") y creamos una partición FAT16 sobre la que posteriormente creamos un sistema de archivos (eso que algunos llaman "formatear") con `mkfs`:

```
wanda:/mnt# fdisk /dev/sda
wanda:/mnt# mkfs.vfat /dev/sda1
```

El tema de las particiones es delicado y va a depender de los dispositivos con los que, posteriormente, vayamos a utilizar la tarjeta de memoria.

Por ejemplo, en mi caso, mi cámara digital (Kodak DX4330) crear una partición FAT16 y, en esa partición, después, crea el sistema de ficheros. Si la tarjeta de memoria no tiene esa partición, no puede leer los datos y solicita el "formateo" de la tarjeta al encender la cámara.

Por otra parte, dispongo de una Palm que funciona con Tarjetas MMC/SD y acepta sin problemas cualquier formato FAT16. Pensad que esa tarjeta de memoria se comporta como un disco duro de pequeña capacidad y deberéis adecuar las particiones y el sistema de ficheros a utilizar a vuestras necesidades.

Si vuestra tarjeta ya está formateada porque antes la habéis utilizado en un dispositivo (una cámara, una agenda electrónica o cualquier otro) debéis averiguar si tiene o no particiones definidas. Esto es importante porque si tiene particiones definidas, el dispositivo a montar será `/dev/sda1` y si no será simplemente `/dev/sda`. Podéis averiguar fácilmente eso utilizando `fdisk /dev/sda` y luego pulsando "p" para ver las particiones y "q" para salir sin hacer cambios.



Una vez averiguado el dispositivo de bloques a montar (/dev/sda o /dev/sda1) creamos un directorio donde montar el dispositivo y pasamos a montarlo de manera sencilla con:

```
# mkdir /mnt/flash
# mount /dev/sda /mnt/flash -t vfat
```

Como siempre, hay una manera más cómoda de hacer esto y es editando el /etc/fstab para añadir la línea:

```
# Lector de tarjetas de memoria USB

/dev/sda1 /mnt/flash vfat noauto,user,rw,gid=100,umask=002 0 0
```

Permitiendo, así, además, que los usuarios del grupo 100 puedan leer y escribir sobre la tarjeta. En mi máquina, el grupo 100 es "users" y para añadir un usuario al grupo 100 hay que hacer:

```
# adduser <nombreusuario> <nombregrupo>
```

o bien editar el fichero /etc/group y añadir los usuarios a mano.

Una vez editado el fichero /etc/fstab podremos montar nuestros dispositivos simplemente con:

```
# mount /mnt/flash
```

Linux utiliza intensamente el buffer caché sobre este tipo de dispositivos y evita, en la medida de lo posible las escrituras, ya que el número de escrituras en estas tarjetas de memoria es limitado. Por eso, es muy importante desmontar el dispositivo antes de desconectarlo (umount /mnt/flash) para que ninguno de nuestros datos se pierda.

Lista de enlaces de este artículo:

1. <http://bulma.net/~pacoros/articulos/lectorflash1.jpg>
2. <http://bulma.net/~pacoros/articulos/lectorflash2.jpg>

E-mail del autor: paco.ros_ARROBA_gmail.com

Podrás encontrar este artículo e información adicional en: <http://bulma.net/body.phtml?nIdNoticia=1883>