

# Acceder a los sistemas de archivos

## 1. mount

El comando **mount** permite acceder a los periféricos de tipo bloque (las particiones) donde está el sistema de archivos. El comando **mount** sitúa el sistema de archivos que se van a montar en un punto del sistema principal llamado punto de montaje (mountpoint).

```
mount -t typefs -o opciones periférico punto_de_montaje
```

### a. Montaje por periférico

Como la partición sdb1 dispone de nuevo de un sistema de archivos ext4, el comando siguiente monta la raíz del sistema de archivos que contiene sdb1 en el directorio /mnt/DATA.

```
# mount -t ext4 /dev/sdb1 /mnt/DATA
```

El comando **mount**, cuando se ejecuta sin parámetros, muestra todos los detalles en los sistemas de archivos actualmente montados (periféricos, sistemas de archivos, puntos de montaje, opciones):

```
# mount | grep '/dev/sd[a-z]'  
/dev/sda1 on /boot type xfs (rw,relatime,seclabel,attr2,inode64,  
logbufs=8,logbsize=32k,noquota)  
/dev/sdb1 on /mnt/DATA type ext4 (rw,relatime,seclabel)
```

Se puede acceder a la misma información visualizando el contenido del archivo `/etc/mtab`.

```
# grep '/dev/sd[a-z]' /etc/mtab
```

```
/dev/sda1 /boot xfs rw,seclabel,relatime,attr2,inode64,logbufs=8,
logbsize=32k,noquota 0 0
/dev/sdb1 /mnt/DATA ext4 rw,seclabel,relatime 0 0
```

## Montaje por etiqueta

Es tremendamente práctico utilizar etiquetas para los sistemas de archivos montados en periféricos. Por ejemplo, cuando se procede a reordenar los discos (movimientos en los arrays de discos SCSI, por ejemplo), se modifica la ordenación de los periféricos con respecto al sistema de archivos principal. Volver a nombrar los periféricos obliga a modificar el archivo `/etc/fstab` para cada uno de los discos. Esto se evita con el uso de etiquetas. Utilice el parámetro `-L` de mount, seguido del nombre del volumen, como en el siguiente ejemplo:

```
# mount -t ext4 -L DATA /mnt/DATA
# mount
...
/dev/sdb1 on /mnt/DATA type ext4 (rw)
```

Se puede obtener la lista de las etiquetas de disco actualmente reconocidas por Linux listando el directorio `/dev/disk/by-label`. Observe que la etiqueta es un vínculo simbólico hacia el archivo periférico correspondiente:

```
# ls -l /dev/disk/by-label/
total 0
lrwxrwxrwx 1 root root 9 abr 1 22:00 'Debian\x2010.9.0\x20amd64\x20n' ->
../sr0
```

## Montaje por UUID

Cada sistema de archivos dispone de un identificador único llamado **UUID** que es un sinónimo de GUID en Linux: un número aleatorio codificado en bits lo suficientemente complejo como para que todos sean diferentes en uno o varios sistemas dados. De este modo, si el disco cambia de posición lógica, el UUID no cambia y mount encuentra el sistema de archivos, cuando en teoría es más probable que dos sistemas de archivos

lleven la misma etiqueta.

Existen varios métodos para conocer el UUID de una partición. Si la herramienta **udev** se utiliza en su Linux, entonces probablemente usted disponga del comando **vol\_id**. Es posible que la ruta hasta el comando no esté en la variable de rutas predefinidas PATH. Es el caso en OpenSUSE, que lo coloca en **/lib/udev**.

```
# ./vol_id -u /dev/sdb1
67f6e4b8-635c-4103-9a81-877fb7db29fe
```

En las distribuciones más recientes, ya no existe vol-id. El comando **blkid** lo reemplaza eficazmente:

```
# blkid /dev/sdb1
/dev/sdb1: LABEL="DATA" UUID="b6dbdb96-7149-4377-beb6-28003f5253c2"
TYPE="ext4" PARTLABEL="root_bis" PARTUUID=
"37e2d6a8-6cfc-4f2e-a5aa-ae6cca7ba2ed"
```

Fíjese en la talla del valor PARTUUID que muestra una partición de tipo GPT. Para mostrar sólo el UUID, especifique únicamente este tag con el parámetro **-s UUID** y únicamente el valor con el parámetro **-o value**:

```
# blkid -o value -s UUID /dev/sdb1
b6dbdb96-7149-4377-beb6-28003f5253c2
```

Si su sistema de archivos es de tipo ext2 o ext3, el comando **dumpe2fs** devuelve mucha información, entre la que se encuentra el UUID:

```
# dumpe2fs -h /dev/sdb1 | grep UUID
Filesystem UUID:      67f6e4b8-635c-4103-9a81-877fb7db29fe
```

Al igual que en el caso de las etiquetas, el archivo **/dev/disk/by-uuid** contiene los enlaces simbólicos de los UUID que apuntan al archivo periférico correspondiente:

```
# ls -l /dev/disk/by-uuid/
total 0
total 0
lrwxrwxrwx. 1 root root 10 29 ene. 21:08 1625591b-2767-4832-8655-0a0ae98407f1 -> ../../sdb3
lrwxrwxrwx. 1 root root 9 29 ene. 20:44 2020-01-13-11-12-56-98 -> ../../sr0
lrwxrwxrwx. 1 root root 10 29 ene. 20:44 812331ea-4b31-4da1-b389-156ccbcda7da -> ../../dm-1
lrwxrwxrwx. 1 root root 10 29 ene. 21:08 93df9c77-321c-4ba9-b181-21964ab30380 -> ../../sdb2
lrwxrwxrwx. 1 root root 10 29 ene. 21:08 b6dbdb96-7149-4377-beb6-28003f5253c2 -> ../../sdb1
lrwxrwxrwx. 1 root root 10 29 ene. 20:44 bed293c8-19bf-4413-bf22-f2b6cae15f19 -> ../../sda1
lrwxrwxrwx. 1 root root 10 29 ene. 20:44 f2233fbe-4485-4433-933c-9a5aeb2ff348 -> ../../dm-0
```

Al final, los contenidos de **/dev/disk/by-partlabel** y **/dev/disk/by-partuuid** contienen los vínculos de las etiquetas y UUID que señalan los periféricos correspondientes, para los discos en formato GPT:

```
# ls -l /dev/disk/by-partlabel/
lrwxrwxrwx. 1 root root 10 29 ene. 21:08 home_bis -> ../../sdb2
lrwxrwxrwx. 1 root root 10 29 ene. 21:08 root_bis -> ../../sdb1
```



Observe que `vol_id` reconoce los UUID de la mayoría de los sistemas de archivos, incluyendo FAT y NTFS. Además, tanto `vol_id` como `dumpe2fs` muestran mucha más información.

Para montar un sistema de archivos por UUID, utilice el parámetro `-U` de `mount`:

```
# mount -t ext4 -U b6dbdb96-7149-4377-beb6-28003f5253c2 /mnt/DATA
```

### Volver a montar un sistema de archivos

No está obligado a desmontar y volver a montar después el sistema de archivos cuando haga una modificación de alguna opción del sistema de archivos (mediante el parámetro

`-o`). Solamente tendrá que usar la opción `remount` para que se tome en cuenta en seguida la modificación. No vuelva a insertar la línea de comandos completa, sino únicamente el periférico o el punto de montaje. En el ejemplo siguiente se vuelve a montar el sistema de archivos en modo de sólo lectura:

```
# mount -o ro,remount /mnt/DATA
# mount
...
/dev/sdb1 on /mnt/DATA type ext4 (ro,relatime,seclabel)
```

## b. Opciones de montaje

Cada sistema de archivos acepta un cierto número de opciones de montaje que se pueden especificar después del parámetro `-o` de `mount`. Se separan las opciones por comas. Salvo indicación contraria, las opciones siguientes funcionan con `ext2`, `ext3` y `ext4`.

Opción	Significado
<code>defaults</code>	Casi siempre presente, la opción <b>defaults</b> sustituye a las opciones <code>rw</code> , <code>suid</code> , <code>dev</code> , <code>exec</code> , <code>auto</code> , <code>nouser</code> y <code>async</code> por defecto.
<code>sync/async</code>	Activa o desactiva las escrituras síncronas. Con <code>async</code> la escritura pasa por una memoria intermedia que la difiere (más eficaz) devolviendo el control más rápido. Es preferible activar la escritura síncrona en soportes externos (pendrives, discos USB/Firewire/eSATA, etc.).
<code>exec/noexec</code>	Permite la ejecución/no ejecución de los archivos binarios en el soporte.
<code>noatime</code>	Evita la actualización de la marca temporal de acceso a un archivo (aconsejable para los soportes externos, discos SSD, páginas web, newsgroups, etc.).

<code>auto/noauto</code>	Con <code>auto</code> , el sistema de archivos se monta automáticamente. Con <code>noauto</code> , sólo se puede montar explícitamente (ver <code>fstab</code> ).
<code>user/nouser</code>	Cualquier usuario puede montar el sistema de archivos (implica <code>noexec</code> , <code>nosuid</code> y <code>nODEV</code> )/sólo <code>root</code> tiene permiso para montar el sistema de archivos (ver <code>fstab</code> ).
<code>remount</code>	Remontaje del sistema de archivos para tener en cuenta nuevas opciones.
<code>ro/rw</code>	Montaje en modo de sólo lectura o lectura y escritura.
<code>dev/nODEV</code>	Interpretar/No interpretar los archivos especiales.
<code>noload</code>	Para <code>ext3</code> , no carga el diario.
<code>usrquota/</code> <code>grpquota</code>	Ignorado por el propio sistema de archivos, pero utilizado por el subsistema de cuotas.
<code>acl</code>	Permite el uso de los Access Control Lists.
<code>user_xattr</code>	Para <code>ext2/3/4</code> y <code>XFS</code> , acepta los atributos extendidos en los archivos, por ejemplo para pegar en ellos información adicional (la codificación del texto, etc.), campos de indexación, etc.
<code>umask</code>	Para <code>FAT/NTFS</code> , aplica otra máscara global que la predeterminada (ex 133).
<code>dmask= / fmask=</code>	<code>FAT/NTFS</code> , diferencia las máscaras para los directorios y los archivos.

<code>uid=/gid=</code>	Como no se gestionan los permisos y propietarios, FAT/NTFS aplica un usuario o un grupo por defecto en los archivos (ej gid=users).
<code>ssd</code>	BTRFS, activa la optimización para los discos SSD. Las versiones más recientes de btrfs aplican esta opción automáticamente si detectan un disco "no rotativo".
<code>compress</code>	BTRFS, activa la compresión automática de los datos. A partir del núcleo 2.6.38 puede seleccionar el formato: compress=zlib, lzo o zstd.
<code>autodefrag</code>	BTRFS, activa la desfragmentación automática del sistema de archivos. Sólo disponible en el núcleo 3.0 y superior puede provocar problemas en el rendimiento.
<code>seclabel</code>	Esta opción la añade automáticamente el sistema cuando selinux (security enhanced Linux) está activado. No la añade.

### c. umount

El comando **umount** desmonta el sistema de archivos del punto de montaje.

```
# umount /mnt/DATA
```

Si se están utilizando uno o varios archivos del sistema de archivos por desmontar, **umount** no funcionará. Debe asegurarse de que ningún proceso acceda al sistema de archivos que pretende desmontar.

```
# umount /mnt/DATA
umount: /mnt/DATA: periférico ocupado
```

El comando **lsuf** le ayuda a determinar qué proceso está utilizando un archivo del punto de montaje en el momento de iniciar el comando. En nuestro ejemplo, el proceso que está accediendo al archivo es el shell bash ejecutado por el usuario seb (probablemente debido a que el directorio actual es /mnt/DATA).

```
# lsuf /mnt/DATA
COMMAND PID USER  FD  TYPE DEVICE SIZE  NODE NAME
bash    5366 seb  cwd  DIR  8,17 4096   2  /mnt/DATA
```

Como administrador, puede iniciar **fuser** para forzar la parada de los procesos que estén accediendo al punto de montaje. Es más que probable que no le guste nada al usuario correspondiente (en el caso presentado aquí, se parará su shell y se cerrará la sesión).

```
# fuser -km /mnt/DATA
```

#### d. /etc/fstab

El archivo **/etc/fstab** contiene una configuración estática de las diferentes opciones de montaje de los sistemas de archivos. Este archivo es invocado siempre que se inicia el sistema, ya que es aquí donde se especifican los periféricos y sus puntos de montaje. Contiene seis campos.

```
periférico punto_de_montaje typefs opciones dump fsck
```

Los espacios o las tabulaciones hacen de separadores de campos.



Campo	Descripción
periférico	El periférico que se va a montar. Se puede especificar como ruta de periférico (/dev/hda1 por ejemplo), etiqueta de sistema de archivos si existe (LABEL=/home) UUID (UUID=xxxx). Para GPT podemos emplear además PARTUUID (PARTUUID=xxxx) o PARTLABEL (PARTLABEL=root).
punto de montaje	El directorio de acceso al sistema de archivos montado.
typefs	El tipo (ext2, ext3, ext4, btrfs, ReiserFS, VFAT, XFS, etc.) del sistema de archivos.
opciones	Opciones de montaje separadas por comas.
dump	Frecuencia de volcado para las herramientas de copia de seguridad.
fsck	Frecuencia de verificación del sistema de archivos. 0=ignorar. 1=en primero, 2 en segundo, etc. Se verifican en paralelo los sistemas que tienen el mismo número.

A continuación presentamos un ejemplo (truncado, ya que los sistemas de archivos virtuales no aparecen) de archivo `/etc/fstab`:

```

/dev/sda3 /      ext4 acl,user_xattr          1 1
/dev/sda2 /boot  ext4 acl,user_xattr          1 2
/dev/sdb1 /home  ext4 acl,user_xattr          1 2
/dev/sda6 /public ext4 acl,user_xattr          1 2
/dev/sda1 /windows ntfs noauto,users,gid=users,umask=0002,utf8=true 0 0
/dev/sda5 swap   swap defaults                    0 0

```

En vez de especificar nombres de periféricos estáticos, puede ser preferible especificar una etiqueta (label, volumen) o un UUID.

```

LABEL=BOOT /boot          ext4      acl,user_xattr   1 2
UUID= b6dbdb96-7149-4377-beb6-28003f5253c2 /mnt/DATA ext3  acl,user_xattr 1 2

```

Para terminar, si no desea utilizar ninguna de estas soluciones, pero tiene que usar rutas, puede valerse de los vínculos simbólicos presentes en cada `/dev/disk/by-XXX`, donde xxx representa:

- ˆ id: los identificadores de controlador, hardware y partición de los diferentes volúmenes;
- ˆ label: los identificadores mediante etiqueta;
- ˆ partlabel: solamente para las particiones GPT, las etiquetas de las particiones;
- ˆ uuid: los identificadores UUID;
- ˆ pathuuid: solamente para las particiones GPT, los UUIDs de las particiones;
- ˆ path: los identificadores por ruta física (bus, lun, etc.).

### Montaje durante el arranque

Durante la secuencia de inicio, el archivo `/etc/fstab` es escaneado por uno de los scripts (init System V, upstart) o systemd (systemd.boot), al inicio del arranque, después de cargar el kernel y antes de iniciar los servicios. Todos aquellos sistemas de archivos que no tengan configurada la opción `noauto` se montan automáticamente porque el montaje automático está implícito. El primero es el sistema de archivos raíz /. Luego vienen la swap y los demás sistemas de archivos especificados (p. ej.: /home, /usr, etc.) para terminar con los sistemas virtuales /proc, /sys, /dev/pts, etc.

### Montaje manual

Se puede utilizar el contenido de `/etc/fstab` después de la puesta en marcha del sistema para montar y desmontar de manera puntual los sistemas de archivos que no disponen, por ejemplo, de la opción **noauto**, o los soportes masivos como los lectores CD/DVD. En este caso, use el comando `mount` incluyendo simplemente la etiqueta, los puntos de montaje o el periférico sin tener que volver a escribir toda la línea de comandos.

Mount buscará el resto de las opciones en /etc/fstab.

```
mount /home
mount -L /u01
mount LABEL=/boot
mount /dev/sda5
```

### Montarlo todo

Si ha realizado modificaciones importantes en la fstab, como añadir varios puntos de montaje, puede, en vez de montar cada sistema de archivos uno por uno, montarlos todos con el parámetro `-a` de mount: `#`

```
mount -a
```

### bind

La opción particular bind puede ser muy práctica para montar una parte de sistema de archivos en varios puntos de montaje. Permite evitar el uso de vínculos simbólicos y sus defectos (modificación del archivo al que se señala y nombre del propio vínculo). En el ejemplo siguiente se relaciona el sistema de archivos que tiene el label **u01** con el directorio **/u01**. Luego se relaciona **/u01/applis** con el directorio **/applis**.

```
LABEL=/u01  /u01      ext4  defaults 1 2
/u01/applis /applis    none  bind
```

## e. El caso de los CD e imágenes ISO

Los CD-Rom, DVD-Rom y otros soportes de este tipo se montan como cualquier otro soporte de bloques. Los CD-Rom y algunos DVD-Rom utilizan el sistema de archivos **iso9660**.

```
# mount -t iso9660 /dev/sr0 /media/cdrom
```

La mayoría de los DVD-Rom utilizan, sin embargo, el formato **UDF** (*Universal Disk Format*).

```
# mount -t udf /dev/sr1 /media/dvd
```



Las distribuciones Linux recientes se liberan del montaje manual de los soportes externos, ya se trate de un CD, DVD, pendrive o disco externo. Los servicios udev, al conectar o insertar el soporte, se encargan de crear los archivos especiales asociados, y luego de montar y desmontar los soportes de manera automática.

Un archivo ISO es una imagen del contenido de un CD o DVD. Es un sistema de archivos iso9660 o udf en un archivo. Es posible utilizar esta imagen como periférico con ayuda de la opción loop. Esta técnica consiste en relacionar la imagen con un periférico en modo loopback, y conseguir así que las herramientas lo traten como un disco más.

```
# mount -o loop -t iso9660 image.iso /mnt/iso
```