

fstab (Español)

El archivo [/etc/fstab](#) es usado para definir cómo las particiones, distintos dispositivos de bloques o sistemas de archivos remotos deben ser montados e integrados en el sistema.

Cada sistema de archivos se describe en una línea separada. Estas definiciones se convertirán con [systemd](#) en unidades montadas de forma dinámica en el arranque, y cuando se recargue la configuración del administrador del sistema.

El archivo es leído por la orden `mount`, a la cual le basta con encontrar cualquiera de los directorios o dispositivos indicados en el archivo para completar el valor del siguiente parámetro. Al hacerlo, las opciones de montaje que se enumeran en `fstab` también se aplicarán.

Artículos relacionados

[Persistent block device naming](#)

[NTFS Write Support](#)

[Firefox Ramdisk](#)

[Boot debugging](#)

[udev \(Español\)](#)

Contents

[Ejemplo de archivo](#)

[Definiciones de los campos](#)

[Identificación de los sistemas de archivos](#)

[Nombre del kernel](#)

[Etiqueta](#)

[UUID](#)

[Consejos y trucos](#)

[Montaje automático con systemd](#)

[UUID para swap](#)

[Utilizar rutas que contienen espacios](#)

[Dispositivos externos](#)

[Opciones atime](#)

[tmpfs](#)

[Uso](#)

[Mejorar el tiempo de compilación](#)

[Escribir en FAT32 como usuario normal](#)

[Volver a montar la partición root](#)

[Véase también](#)

Ejemplo de archivo

He aquí una muestra de archivo `/etc/fstab` , utilizando nombres descriptivos del kernel:

```
/etc/fstab
```

# <file system>	<dir>	<type>	<options>	<dump>	<pass>
/dev/sda1	/	ext4	defaults,noatime	0	1
/dev/sda2	none	swap	defaults	0	0
/dev/sda3	/home	ext4	defaults,noatime	0	2

Definiciones de los campos

El archivo `/etc/fstab` contiene los siguientes campos separados por un espacio o una tabulación:

<file system>	<dir>	<type>	<options>	<dump>	<pass>
---------------	-------	--------	-----------	--------	--------

- **<file system>** - Define la partición o dispositivo de almacenamiento para ser montado.
- **<dir>** - Indica a la orden `mount` el punto de montaje donde la partición (`<file system>`) será montada.
- **<type>** - Indica el tipo de sistema de archivos de la partición o dispositivo de almacenamiento para ser montado. Hay muchos sistemas de archivos diferentes que son compatibles como, por ejemplo: `ext2` , `ext3` , `ext4` , `reiserfs` , `xfs` , `jfs` , `smbfs` , `iso9660` , `vfat` , `ntfs` , `swap` y `auto` . El `type auto` permite a la orden `mount` determinar qué tipo de sistema de archivos se utiliza. Esta opción es útil para proporcionar soporte a unidades ópticas (CD/DVD).
- **<options>** - Indica las opciones de montaje que la orden `mount` utilizará para montar el sistema de archivos. Tenga en cuenta que algunas opciones de montaje son para sistema de archivos específicos [mount\(8\)](https://jlk.fjfi.cvut.cz/arch/manpages/man/mount.8) (<https://jlk.fjfi.cvut.cz/arch/manpages/man/mount.8>). Algunas de las opciones más comunes son:
 - `auto` - El sistema de archivos será montado automáticamente durante el arranque, o cuando la orden `mount -a` se invoque.
 - `noauto` - El sistema de archivos no será montado automáticamente, solo cuando se le ordene manualmente.
 - `exec` - Permite la ejecución de binarios residentes en el sistema de archivos.
 - `noexec` - No permite la ejecución de binarios que se encuentren en el sistema de archivos.
 - `ro` - Monta el sistema de archivos en modo sólo lectura.
 - `rw` - Monta el sistema de archivos en modo lectura-escritura.
 - `user` - Permite a cualquier usuario montar el sistema de archivos. Esta opción incluye `noexec` , `nosuid` , `nodelv` , a menos que se indique lo contrario.
 - `users` - Permite que cualquier usuario perteneciente al grupo `users` montar el sistema de archivos.
 - `nouser` - Solo el usuario `root` puede montar el sistema de archivos.
 - `owner` - Permite al propietario del dispositivo montarlo.
 - `sync` - Todo el I/O se debe hacer de forma sincrónica.
 - `async` - Todo el I/O se debe hacer de forma asíncrona.
 - `dev` - Intérprete de los dispositivos especiales o de bloque del sistema de archivos.

- **nodev** - Impide la interpretación de los dispositivos especiales o de bloques del sistema de archivos.
 - **suid** - Permite las operaciones de suid, y sgid bits. Se utiliza principalmente para permitir a los usuarios comunes ejecutar binarios con privilegios concedidos temporalmente con el fin de realizar una tarea específica.
 - **nosuid** - Bloquea el funcionamiento de suid, y sgid bits
 - **noatime** - No actualiza el inode con el tiempo de acceso al filesystem. Puede aumentar las prestaciones (véase [opciones atime](#)).
 - **nodiratime** - No actualiza el inode de los directorios con el tiempo de acceso al filesystem. Puede aumentar las prestaciones (véase [opciones atime](#)).
 - **relatime** - Actualiza en el inode solo los tiempos relativos a modificaciones o cambios de los archivos. Los tiempos de acceso vienen actualizados solo si el último acceso es anterior respecto al de la última modificación. (Similar a noatime, pero no interfiere con programas como mutt u otras aplicaciones que deben conocer si un archivo ha sido leído después de la última modificación). Puede aumentar las prestaciones (véase [opciones atime](#)).
 - **discard** - Emite las órdenes **TRIM** para dispositivos de bloques subyacentes cuando se liberan los bloques. Recomendado para usar si el sistema de archivos se encuentra en un [SSD](#).
 - **flush** - La opción **vfat** permite eliminar datos con más frecuencia, de modo que los cuadros de diálogo de copia o las barras de progreso se mantenga hasta que se hayan escrito todos los datos.
 - **nofail** - Monta el dispositivo cuando está presente, pero ignora su ausencia. Esto evita que se cometan errores durante el arranque para los medios extraíbles.
 - **defaults** - Asigna las opciones de montaje predeterminadas que serán utilizadas para el sistema de archivos. Las opciones predeterminadas para **ext4** son: **rw**, **suid**, **dev**, **exec**, **auto**, **nouser**, **async**.
-
- **<dump>** - Utilizado por el programa **dump** («*volcado*») para decidir cuándo hacer una copia de seguridad. **Dump** comprueba la entrada en el archivo **fstab** y el número de la misma le indica si un sistema de archivos debe ser respaldado o no. La entradas posibles son 0 y 1. Si es 0, **dump** ignorará el sistema de archivos, mientras que si el valor es 1, **dump** hará una copia de seguridad. La mayoría de los usuarios no tendrán **dump** instalado, por lo que deben poner el valor 0 para la entrada **<dump>**.
 - **<pass>** -Utilizado por [fsck](#) para decidir el orden en el que los sistemas de archivos serán comprobados. Las entradas posibles son 0, 1 y 2. El sistema de archivos raíz («*root*») debe tener la más alta prioridad: 1 -todos los demás sistemas de archivos que desea comprobar deben tener un 2-. La utilidad **fsck** no comprobará los sistemas de archivos que vengan ajustados con un valor 0 en **<pass>**.

Identificación de los sistemas de archivos

Hay tres maneras de identificar una partición o un dispositivo de almacenamiento en **/etc/fstab** : por el nombre descriptivo del kernel, por la etiqueta o por la UUID. La ventaja de usar etiquetas o UUID, es que no dependen del orden en el que las unidades son (físicamente) conectadas a la máquina. Esto es útil si se cambia el orden de los dispositivos de almacenamiento en la BIOS, o si se cambia el cableado de conexión de los dispositivos de almacenamiento. Ocurre, también, a veces, que la BIOS cambia el orden de los dispositivos de almacenamiento. Léase más sobre este tema en el artículo [Persistent block device](#)

naming.

Para mostrar información básica acerca de las particiones, ejecute:

```
$ lsblk -f
```

NAME	FSTYPE	LABEL	UUID	MOUNTPOINT
sda				
└sda1	ext4	Arch_Linux	978e3e81-8048-4ae1-8a06-aa727458e8ff	/
└sda2	ntfs	Windows	6C1093E61093B594	
└sda3	ext4	Storage	f838b24e-3a66-4d02-86f4-a2e73e454336	/media/Storage
sdb				
└sdb1	ntfs	Games	9E68F00568EFD9D3	
└sdb2	ext4	Backup	14d50a6c-e083-42f2-b9c4-bc8bae38d274	/media/Backup
sdcc				
└sdcc1	vfat	Camera	47FA-4071	/media/Camera

Nombre del kernel

Ejecute `lsblk -f` para mostrar la lista de las particiones, y coloque el nombre visualizado precedido del prefijo `/dev`.

Véase el [ejemplo](#) anterior.

Etiqueta

Nota: Cada etiqueta («*label*») debe ser única, para evitar posibles conflictos.

Para etiquetar un dispositivo o partición, véase [este artículo](#). También puede instalar y usar [gparted](https://www.archlinux.org/packages/?name=gparted) (<https://www.archlinux.org/packages/?name=gparted>), pero para cambiar el nombre de la partición raíz tendrá que hacerse desde una distribución Linux «live» (Parted Magic, Ubuntu, etc.), ya que la partición se debe desmontar primero.

Ejecute `lsblk -f` para mostrar la lista de las particiones, y coloque la etiqueta visualizada precedida del prefijo `LABEL=`:

```
/etc/fstab
```

# <file system>	<dir>	<type>	<options>	<dump>	<pass>
LABEL=Arch_Linux	/	ext4	defaults,noatime	0	1
LABEL=Arch_Swap	none	swap	defaults	0	0

UUID

Todas las particiones y dispositivos tienen un UUID único. Los UUID son generados por las utilidades de creación del sistema de archivos (por ejemplo, `mkfs.*`) al crear o formatear una partición.

Ejecute `lsblk -f` para mostrar la lista de las particiones, y coloque el número identificador

de la unidad visualizado precedido del prefijo `UUID=` :

Sugerencia: Si desea conocer solo el UUID de una partición específica, escriba:

```
$ lsblk -no UUID /dev/sda2
```

/etc/fstab

# <file system>	<dir>	<type>	<options>	<dump>	<p>
ass>					
UUID=24f28fc6-717e-4bcd-a5f7-32b959024e26	/	ext4	defaults,noatime	0	1
UUID=03ec5dd3-45c0-4f95-a363-61ff321a09ff	/home	ext4	defaults,noatime	0	2
UUID=4209c845-f495-4c43-8a03-5363dd433153	none	swap	defaults	0	0

Consejos y trucos

Montaje automático con systemd

Si tiene una partición `/home` grande, tal vez sería mejor permitir que los servicios que no dependen de `/home` se inicien, mientras `/home` es comprobada. Esto se puede lograr mediante la adición de las siguientes opciones en la entrada de la partición `/home` en `/etc/fstab` :

```
noauto,x-systemd.automount
```

Esto comprobará el sistema de archivos y montará `/home` cuando se acceda a la misma por primera vez, y el kernel demorará todos los accesos a los archivos de `/home` , almacenándolos en un búfer, hasta que la partición esté lista.

Nota: Esto hará que el sistema de archivos de la partición `/home` se marque con el tipo `autofs` , que se ignora por **mlocate** de forma predeterminada. El aumento de velocidad de automontaje de `/home` puede no ser más que de uno o dos segundos, dependiendo de su sistema, por lo que este arreglo puede no valer la pena.

Lo mismo se aplica a los montajes del sistema de archivos remoto. Si quiere que se monte solo cuando se acceda, tendrá que usar el parámetro `noauto,x-systemd.automount` . Además, puede utilizar la opción `x-systemd.device-timeout=#` para especificar un tiempo de espera para el caso de que el recurso de red no esté disponible.

Si tiene sistemas de archivos cifrados con `keyfiles`, también puede añadir el parámetro `noauto` para las entradas correspondientes de `/etc/crypttab` . `systemd` no abrirá el dispositivo cifrado en el arranque, sino que esperará hasta que realmente se acceda al mismo y entonces lo abrirá automáticamente con el archivo de claves («*keyfiles*») especificado antes de montarlo. Esto podría ahorrar unos segundos en el arranque si se está

usando, por ejemplo, un dispositivo RAID cifrado, porque *systemd* no tiene que esperar a que el dispositivo esté disponible. Por ejemplo:

```
/etc/crypttab
-----
data /dev/md0 /root/key noauto
```

UUID para swap

En caso de que la partición de intercambio no tenga un UUID, será posible agregarlo manualmente. Es el caso de que no se muestre el UUID de swap cuando se ejecuta la orden `lsblk-f`. Estos son algunos pasos a seguir para poder asignar un UUID a la partición swap:

Identifique la partición swap:

```
# swapon -s
```

Desactive la partición swap:

```
# swapoff /dev/sda7
```

Vuelva a crear la partición swap con un nuevo UUID asignado a la misma:

```
# mkswap -U random /dev/sda7
```

Active nuevamente swap:

```
# swapon /dev/sda7
```

Utilizar rutas que contienen espacios

Si algún punto de montaje contiene espacios en el nombre, use el carácter de la barra invertida `\` seguido por el código de 3 dígitos `040` para emular el espacio:

```
/etc/fstab
-----
UUID=47FA-4071    /home/username/Camera\040Pictures  vfat  defaults,noatime    0 0
/dev/sda7         /media/100\040GB\040(Storage)      ext4  defaults,noatime,user 0 2
```

Dispositivos externos

Los dispositivos externos que van a ser montados cuando estén presentes pero ignorados si estén ausentes, pueden requerir el uso de la opción `nofail`. Esta opción evita que se muestren mensajes de error durante el arranque.

```
/etc/fstab
-----
/dev/sdg1      /media/backup  jfs      defaults,nofail 0 2
```

Opciones atime

El uso de las opciones `noatime`, `nodiratime` o `relatime` pueden mejorar las prestaciones del disco. De forma predeterminada, Linux mantiene un registro (lo escribe en el disco) de cada lectura efectuada `atime`. Esto es útil cuando se usa Linux para servidores, pero no tiene mucho valor cuando se usa para escritorio. El inconveniente de la opción predeterminada `atime` es que incluso la lectura de un archivo desde la memoria caché (lectura desde la memoria, en lugar desde el disco directamente), aún en este caso, ¡resultará registrado! Usando la opción `noatime` deshabilita completamente la actualización del tiempo de acceso a los archivos cada vez que se lee un archivo. Esto funciona bien para casi todas las aplicaciones, excepto para algunas raras como [Mutt](#) que necesitan dicha información. El uso de la opción `relatime` permite la actualización de los tiempos de acceso al archivo solo si el archivo viene modificado (a diferencia de `noatime` donde el tiempo de acceso a los archivos leídos no vendrá actualizado y será anterior respecto al tiempo de modificación del archivo). La opción `nodiratime` deshabilita la escritura de los tiempos de acceso a los archivos, solo para los directorios, mientras que para el resto de los archivos todavía los tiempos de acceso serán actualizados. El mejor compromiso podría ser el uso de `relatime` que no influyen en el funcionamiento de los programas como [Mutt](#), pero todavía se obtendrá un aumento del rendimiento porque los tiempos de acceso a los archivos vendrán actualizados solo si los archivos son modificados.

Nota: La opción `noatime` ya incluye `nodiratime`. No es necesario especificar ambas. [\[1\] \(http://lwn.net/Articles/244941/\)](http://lwn.net/Articles/244941/)

tmpfs

[tmpfs](#) es un sistema de archivos temporal que reside en la memoria y/o en la partición de intercambio, dependiendo de lo mucho que se llene. El montaje de carpetas en `tmpfs` puede ser una manera eficaz de acelerar los accesos en la lectura de los archivos, o para asegurar que sus contenidos se borren automáticamente después del reinicio.

Algunas carpetas donde `tmpfs` es comúnmente utilizado son [/tmp](http://www.pathname.com/fhs/2.2/fhs-3.15.html) (<http://www.pathname.com/fhs/2.2/fhs-3.15.html>), [/var/lock](http://www.pathname.com/fhs/2.2/fhs-5.9.html) (<http://www.pathname.com/fhs/2.2/fhs-5.9.html>) y [/var/run](http://www.pathname.com/fhs/2.2/fhs-5.13.html) (<http://www.pathname.com/fhs/2.2/fhs-5.13.html>). NO lo use para [/var/tmp](http://www.pathname.com/fhs/2.2/fhs-5.15.html) (<http://www.pathname.com/fhs/2.2/fhs-5.15.html>), porque esa carpeta es para los archivos temporales que deben ser conservados después del reinicio. Arch usa el directorio `/run` como `tmpfs`, con `/var/run` y `/var/lock` simplemente existentes como enlaces simbólicos para permitir la compatibilidad. `tmpfs` también se utiliza por `/tmp` en la configuración predeterminada del archivo `/etc/fstab`.

Nota: Al utilizar [systemd](#), los archivos y las carpetas temporales en tmpfs se pueden recrear en el arranque usando [tmpfiles.d](#)^{[[broken link](#): invalid section]}.

De forma predeterminada, una partición tmpfs tiene establecido el tamaño máximo a la mitad de la memoria RAM total, pero esto se puede personalizar. Tenga en cuenta que el actual consumo de memoria/swap depende de lo mucho que se llenen, y la partición tmpfs no consumen memoria hasta que no viene usada.

Para usar tmpfs para /tmp , añada esta línea a /etc/fstab :

/etc/fstab					
tmpfs	/tmp	tmpfs	nodev,nosuid	0	0

Se puede o no especificar el tamaño aquí, pero debe dejar la opción `mode` inalterada en este caso, para asegurarse de que tiene los permisos correctos (1777). En el ejemplo anterior, /tmp se establece para utilizar hasta la mitad de la memoria RAM total. Para definir explícitamente un tamaño máximo, utilice la opción `size` de la orden `mount`:

/etc/fstab					
tmpfs	/tmp	tmpfs	nodev,nosuid,size=2G	0	0

He aquí un ejemplo más avanzado que muestra cómo agregar puntos de montaje tmpfs para los usuarios. Esto es útil para los sitios web, los archivos tmp de mysql, ~/.vim/ , y otros. Es importante tratar de obtener las opciones de montaje ideales para el propósito que se está tratando de lograr. La meta es tener tantos valores seguros como sea posible para evitar abusos. Limitar el tamaño y la especificación de uid y gid + mode es muy seguro.

[Más información.](#)

/etc/fstab					
tmpfs	/www/cache	tmpfs	rw,size=1G,nr_inodes=5k,noexec,nodev,nosuid,uid=648,gid=648,mode=1700	0	0

Véase la página `man` relativa a la orden `mount` para obtener más información. Una opción útil de montaje según la página del manual es `default` .

Reinicie para que los cambios surtan efecto. Tenga en cuenta que si bien puede ser tentador simplemente ejecutar `mount -a` para hacer efectivos los cambios inmediatamente, esto hará que todos los archivos que actualmente residen en estos directorios sean inaccesibles (esto es especialmente problemático para ejecutar programas con archivos de bloqueo, por ejemplo). Sin embargo, si todos los directorios están vacíos, es seguro ejecutar `mount -a` en lugar de reiniciar (o móntelos manualmente).

Después de aplicar los cambios, es posible que desee verificar que todo es correcto,

comprobando `/proc/mounts` con la orden `findmnt`:

```
$ findmnt --target /tmp

-----
TARGET SOURCE FSTYPE OPTIONS
/tmp    tmpfs    tmpfs  rw,nosuid,nodev,relatime
```

Uso

En general, las tareas y programas con intensas demandas de I/O realizando frecuentemente operaciones de lectura/escritura, pueden beneficiarse del uso de una carpeta como `tmpfs`. Algunas aplicaciones pueden obtener incluso una mejora sustancial mediante la descarga de parte (o todos) sus datos en la memoria compartida. Por ejemplo, [reubicar el perfil de Firefox en la RAM](#) muestra una mejora significativa en las prestaciones.

Mejorar el tiempo de compilación

Nota: La carpeta `tmpfs` (`/tmp`, en este caso) tiene que ser montada sin la opción `noexec`, lo que evitará que los scripts de compilación u otras utilidades se ejecuten. De otro modo, dado que el tamaño predeterminado, como se indica [más arriba](#), equivale a la mitad de la memoria RAM disponible, podría quedarse sin espacio.

Puede ejecutar [makepkg](#) con una carpeta `tmpfs` como el directorio para la compilación (que también puede ser configurado en `/etc/makepkg.conf`):

```
$ BUILDDIR=/tmp/makepkg makepkg
```

Escribir en FAT32 como usuario normal

Para poder escribir en una partición FAT32, es necesario hacer algunos cambios en el archivo `/etc/fstab`.

```
/etc/fstab
-----
/dev/sdX  /mnt/some_folder  vfat  user,rw,umask=000  0 0
```

El parámetro `user` significa que cualquier usuario (que no sea `root`) puede montar y desmontar la partición `/dev/sdX`. El parámetro `rw` da acceso de lectura-escritura; la opción `umask` quita los permisos seleccionados (por ejemplo `umask=111` quita los permisos de ejecución). El problema es que esta entrada quita también los permisos de ejecución sobre los directorios, así que hay que corregirla con `dmask=000`. Y, ¿por qué el uso de estas opciones? Porque sin estas opciones todos los archivos son ejecutables. Puede utilizar la opción `showexec`, en lugar de las opciones `umask` y `dmask`, que muestra todos los archivos ejecutables de Windows (`com`, `exe`, `bat`) coloreados.

Por ejemplo, si su partición FAT32 está en `/dev/sda9`, y desea montarla en `/mnt/fat32`, entonces se debería utilizar:

```
/etc/fstab
-----
/dev/sda9    /mnt/fat32      vfat    user,rw,umask=111,dmask=000    0    0
```

Volver a montar la partición root

Si por alguna razón la partición raíz se ha montado incorrectamente con acceso de sólo lectura, vuelva a montar la partición con la siguiente orden:

```
# mount -o remount,rw /
```

Véase también

- **Full device listing including block device** (<https://www.kernel.org/pub/linux/docs/lanana/device-list/devices-2.6.txt>)
- **Filesystem Hierarchy Standard** (<http://www.pathname.com/fhs/2.2/index.html>)
- **30x Faster Web-Site Speed** (<http://www.askapache.com/web-hosting/super-speed-secrets.html>)^{[[enlace roto](#) 2020-08-02]} (Detailed tmpfs)
- **Agregando una comparticion a fstab**

Retrieved from "[https://wiki.archlinux.org/index.php?title=Fstab_\(Español\)&oldid=629149](https://wiki.archlinux.org/index.php?title=Fstab_(Español)&oldid=629149)"

This page was last edited on 4 August 2020, at 12:01.

Content is available under [GNU Free Documentation License 1.3 or later](#) unless otherwise noted.