

Lo esencial del archivo Fstab

A) ¿Que es el archivo Fstab?

Fstab significa: **File System Table** (Tabla de Sistemas de ficheros).

Se encuentra en el directorio `/etc/fstab`.

Cuando encendemos nuestro PC con Linux, el sistema operativo se inicia, **en primer lugar** montando en modo solo lectura el sistema de ficheros raíz (/), para los procesos standar, y luego comprueba su consistencia; luego, **en segundo lugar** los desmonta y vuelve a montar, pero esta vez en modo lectura y escritura, y empieza a verificar la existencia y consistencia de todos los demás sistemas de ficheros que se usaran.

La lista de sistemas de ficheros disponibles en nuestro Pc, está contenida en el fichero `/etc/fstab`.

B) Para que sirve

Este archivo esta constituido por varias líneas de texto, y cada línea da información sobre cada uno de los sistemas de ficheros que deberían estar disponibles para tu PC, de modo que:

- para añadir un nuevo sistema de ficheros de forma que se monte y esté disponible siempre, al arrancar el sistema,
- o modificar a tu gusto el comportamiento de tu ordenador con un tipo concreto de dispositivo, (cdroms, dvd, pendrive-usb, etc) lo único que necesitas es añadir o cambiar una línea del fichero `/etc/fstab`.

Este archivo, contiene todos los parámetros para que se monten todas las particiones de uno o varios discos que tengamos en el ordenador.

El archivo `fstab` **es donde se guardan los diferentes datos sobre el montaje de los dispositivos físicos:**

- de discos duros fijos y extraíbles, y de sus particiones,
- de disquetes,
- cdroms,
- dvd,
- pendrives-usb

Lo que en el está escrito nos sirve fundamentalmente para que podamos tener acceso a nuestros discos/particiones, así como también a su funcionamiento, una vez que iniciamos Linux:

- montar la partición donde está Linux y lo que podemos hacer en ella.
- montar o no, al arrancar un disco duro externo, o un pen-drive que tengamos conectado en ese momento.
- poder o no borrar, o escribir en archivos de otras particiones, como la que contenga Windows, si tenemos ambos sistema operativos en el PC.

C) ¿ Que dispositivos podemos montar ?

Podemos montar cualquier dispositivo, que ya exista físicamente creado en el disco, tanto particiones, como discos: por ejemplo un disquet, un pen-drive, un disco duro externo, un dvd, un cdrom.

Todos ellos poseen un formato determinado, y una ubicación del tipo:

`/dev/nombre_de_dispositivo, por ejemplo: /dev/hda1`

Para ello tenemos que saber lo que tenemos montando en nuestro PC: para ver que particiones tenemos montadas: abrimos la Terminal de root, ponemos la contraseña, y una vez ya como root, escribimos:

`# fdisk -l` (ele minúscula)

Lo esencial del archivo Fstab

Y podremos ver algo así, dependiendo del tamaño de nuestro disco y particiones:

*Disco /dev/hda: 8455 MB, 8455200768 bytes
16 cabezas, 63 sectores/pista, 16383 cilindros
Unidades = cilindros de 1008 * 512 = 516096 bytes*

*Disposit. Inicio Comienzo Fin Bloques Id Sistema
/dev/hda1 *1 16383 8257000+ 7 HPFS/NTFS*

*Disco /dev/hdc: 60.0 GB, 60022480896 bytes
255 cabezas, 63 sectores/pista, 7297 cilindros
Unidades = cilindros de 16065 * 512 = 8225280 bytes*

*Disposit. Inicio Comienzo Fin Bloques Id Sistema
/dev/hdc1 2881 7297 35479552+ f W95 Ext'd (LBA)
/dev/hdc2 1 1281 10289601 83 Linux
/dev/hdc3 * 2715 2880 1333395 82 Linux swap / Solaris
/dev/hdc4 1282 2714 11510572+ 83 Linux
/dev/hdc5 2882 7297 35471520 b W95 FAT32*

Las entradas de la tabla de particiones no están en el orden del disco.

Toda esta información puede variar, según las características de tu disco duro (como el tamaño), así también como la cantidad de particiones que tengas en él (*en este caso 1 partición en el primer disco y 5 en el segundo disco*).

Si tienes más de un disco duro , como en este ejemplo, también se muestra:

- 1^{er} disco: *dev/hda*
- 2^o disco: *dev/hdc* .

Con esta información ya sabemos que tenemos dos discos duros: **/dev/hda** y **/dev/hdc**, y la ubicación (por ejemplo /dev/hdc2) y el tipo (Linux) de las particiones que hay en nuestro PC.

D) ¿ Donde podemos montar los dispositivos ?

Puntos de montaje:

El **punto de montaje** "es el lugar donde hacemos posible que se pueda acceder a un dispositivo ya montado"

Por ejemplo el 1^{er} disco *dev/hda1*;

Es decir que lo veamos como una carpeta, dentro de una ubicación del tipo:

/media/punto_de_montaje,

Por ejemplo: */media/Windows-XP;*

Es decir que podemos ver el contenido del sistema operativo Windows que esta en el 1^a disco (*dev/hda1*) en la carpeta */media/Windows-XP*

Lo esencial del archivo Fstab

E) Ver y Modificar el archivo Fstab:

Para verlo, con el administrador de archivos Nautilus llegamos hasta el directorio /etc y en él buscamos el archivo fstab, y lo podemos abrir con el editor de textos gedit; pero de esta manera solo podemos verlo, y estudiarlo, pero no editarlo, pues cualquier cambio que hagamos en él no se guardará al cerrar gedit.

Para modificarlo debemos tener permisos de root. Por lo tanto, debemos abrir el archivo de la siguiente manera:

- Usamos la combinación de teclas **Alt+F2** y en la pantalla que aparece escribimos: **gksu nautilus**
Se nos pedirá la contraseña de root , y se abrirá el administrador de archivos Nautilus, en modo root
- Navegamos por los directorios hasta: **/etc/fstab**
- **Abrimos el archivo fstab** con gedit, **lo modificamos según nuestras necesidades** y luego lo guardamos...y ahora si que se guardan los cambios que hayamos hecho en el archivo.
- Otra opción para editar y poder modificar abrimos un terminal y tecleamos:

```
# sudo gedit /etc/fstab
```

F) Comprender el significado de los datos que aparecen en el archivo fstab

Este es un ejemplo del archivo /etc/fstab

```
# /etc/fstab: static file system information.
( las líneas que empiezan por # no son "leídas" como un comando, sino que sirven como
"etiqueta" de la línea que va a continuación)# /etc/fstab: static file system information.

# <file system> <mount point> <type> <options> <dump> <pass>
# linux ficheros raíz
label=// ext3 defaults,errors=remount-ro 0 1
# linux home
dev/hdc4 /home ext3 defaults 0 2
# disco duro 1 con windows xp
dev/hda1 /media/hda1 ntfs-3g defaults,locale=es_ES.UTF-8 0 1
# usb-pendrive
/dev/sda1 /media/usb-disk vfat users,defaults,utf8,umask=0 0 2
# swap
/dev/hdc3 none swap sw 0 0
# disco duro 2
/dev/hdc1 /media/hdc1 auto users,defaults 0 2
# dvd
/dev/hdb /media/cdrom0 udf,iso9660 user,noauto 0 0
# cdrom
/dev/hdd /media/cdrom1 udf,iso9660 user,noauto 0 0
#disquet
/dev/fd0 /media/floppy0 auto rw,user,noauto 0 0
```

Vamos a empezar con cada uno de los componentes que integran las ordenes del archivo Fstab.

<file system> <mount point> <type> <options> <dump> <pass>

- I. **<file system> Sistema de Archivos:** es el dispositivo local o remoto que vamos a montar.

Lo esencial del archivo Fstab

Normalmente se trata del fichero de dispositivo Linux a montar (por ejemplo /dev/hdc4), pero también puede ser una entrada con el patrón nombre_de_ordenador_en_red: directorio para sistemas de ficheros en red tales como NFS.

Los sistemas de ficheros ext2 y ext3 también se pueden identificar por el nombre asignado a su etiqueta de volumen cuando fueron creados.

Por ejemplo, la entrada label=/ en el fichero de ejemplo /etc/fstab se podría sustituir por /dev/hdc2 porque se trata de la partición de disco donde reside el sistema de ficheros raíz.

Utilizar etiquetas es más flexible que utilizar ficheros de dispositivos de particiones concretas, ya que el fichero de dispositivo asociado a una partición puede cambiar si ese disco duro lo llevamos a otro ordenador o añadimos discos nuevos al nuestro.

II. **<mount point> Punto de montaje:** *es el directorio en donde queremos montar el sistema de ficheros,*

por ejemplo: /media/cdrom0 o /home

Para tipos especiales de particiones que no deben montarse, tales como las particiones de intercambio o swap, éste campo debería tener la entrada (la palabra) none.

III. **<type> Tipo:** *define el tipo del sistema de ficheros.*

Las entradas más frecuentes en éste campo son:

- **ntfs** : Si es de WindowsXP
- **fat** : Si es Windows98 con partición tipo fat 32,
- **vfat**: Si es Windows 98 con partición tipo fat 16
- **iso9660**: Para cdroms, dvd y regrabadoras de cd y dvd
- **sw**: Para la partición de intercambio swap
- **ext2 o ext3** : Para / y /home (en Guadalinux es ext3)
- **auto**: Si deseas, porque no estas seguro de que tipo se trata, puedes poner auto, y así fstab lo detecta automáticamente.
- **ignore**: Si un sistema de ficheros no se está usando pero se quiere mantener su entrada en /etc/fstab, puedes poner ignore, evitar el montaje del sistema de ficheros, su comprobación de consistencia, y el resto de operaciones.

IV. **<options> Opciones:** *Parámetros opcionales para Fstab*

Contiene una lista de parámetros, separados por comas, con las opciones que necesite el comando mount para montar el sistema de ficheros. Muchas de estas opciones son específicas de cada sistema de ficheros.

Las más frecuentes son:

1. **auto**: Con esta opción hacemos que el dispositivo que la contenga sea iniciado siempre que se inicie el sistema, es decir, cada vez que este sea encendido. Es decir que sera montado siempre. o bien cuando se ejecute mount -a.
2. **no auto**: Esta es la opción contraria a auto, con ella hacemos que el dispositivo no sea montado al iniciarse el sistema, y tan solo sera montado en el momento en el que queramos usarlo; esta opción suele ser utilizada con dispositivos como las disqueteras, cd-roms, dvd, pendrives-usb..., es decir, todos aquellos dispositivos que no es necesario que se monten al arranque del sistema.
3. **async**: Con esto hacemos que las operaciones que realicemos no se hagan en el mismo momento en que las realizamos/pedimos, si no que pueden ser realizada mas adelante.
4. **sync**: Es la opción contraria a async. Añadiendo la opción sync fijaríamos que los datos sean transferidos, guardados o borrados, en el momento en que pedimos que se hagan. Esta opción puede servir de mucho, y sobre todo ahorrarnos tiempo en la manera en la que se transfieren datos a dispositivos como los pen-drives y aquellos que se utilizan mediante conexión usb en los cuales necesitamos acceso/transferencia rápido/a.
5. **dev**: Al activar esta opción permitimos que puedan usarse nodos de dispositivo en el sistema de

Lo esencial del archivo Fstab

archivos, o, lo que es lo mismo, que se interpreten como tal los dispositivos especiales de bloques y de caracteres presentes en el dispositivo, el dispositivo (de bloques o de caracteres) es local (un disco duro interno o usb,) no es dispositivo de red

6. **nodev**: con esto impedimos que puedan usarse nodos de dispositivo en el sistema de archivos, es decir, con esta opción impedimos que se interpreten los dispositivos especiales de bloques y de caracteres presentes en el dispositivo.
7. **exec**: *esta opción hace que puedas ejecutar programas, ficheros script (guiones), o cualquier otra cosa cuyos permisos indiquen que es ejecutable, de ese sistema de ficheros.*
8. **noexec**: *Impide la ejecución de cualquier archivo en el dispositivo en el que esta opción sea activada, esta opción suele ser muy útil cuando en una misma maquina diversos usuarios tienen acceso a un mismo dispositivo y no queremos que estos puedan ejecutar archivos en los dispositivos, con esto ganamos en seguridad, ya que impedimos que los otros usuarios puedan ejecutar cualquier tipo de archivo "maligno" a través del cual lo que realmente quieren conseguir es escalar privilegios en el sistema,.*
9. **gid=Valor**: *Con esta opción indicamos que tan solo el grupo que posee el Valor especificado tenga el control sobre los archivos del dispositivo.*
Establece el ID (identificador) de grupo del sistema montado a ese valor cuando se termina de montar el sistema de ficheros.

Puede tratarse de un valor numérico gid=1000 o del nombre del grupo de usuarios (gid=users,gid=usuarioslectordvd)

10. **uid=Valor**: *Con esta opción indicamos que tan solo el usuario que posee el Valor especificado tenga el control sobre los archivos del dispositivo.*
11. **owner**: *indica que el propietario del sistema de ficheros (el que tiene derechos a montar y desmontar el dispositivo) será aquel que lo montó (el primer usuario distinto de root conectado al sistema localmente), pero normalmente sera el usuario root si el sistema de ficheros viene configurado para montarse automáticamente durante el arranque.*
12. **ro**: read-only. *Con esta opción hacemos que no se pueda escribir en el dispositivo, que tan solo se pueda leer.*
13. **rw**: read-write. *Con esto montamos la partición para que sea posible tanto leer como escribir en el dispositivo físico, esta opción es muy usada con dispositivos que permiten la escritura como por ejemplo los pendrive o los disquets, ya que sin esta opción, no podríamos guardar datos en dichos dispositivos.*
14. **suid**: indica que el permiso tenga efecto para los ejecutables presentes en el dispositivo. Permite el cambio del ID de usuario o grupo del usuario a los programas que efectúen operaciones sobre el sistema de ficheros, siempre que sus bits de permisos lo permitan.

Tenga mucho cuidado usando ésta opción con sistemas de ficheros ajenos que en realidad no estén administrando, ya que ejecutar un programa que establezca su UID (identificador de usuario) a root es una forma corriente de romper la seguridad de un sistema.** uid=valor: establece el ID de usuario del sistema montado al valor valor. Igual que con gid, podemos escribirlo con el número identificador del usuario uid=37 o con el nombre del usuario uid=hatteras

15. **nosuid**: es lógicamente la opción opuesta a suid. Salvo casos excepcionales, no debe haber ninguna razón para que se permita la ejecución de programas suid en los directorios /home/usuario. Esto lo podemos evitar usando la opción `nosuid` en el fichero /etc/fstab para las particiones que tengan permiso de escritura por usuarios distintos al root. También puede ser útil usar `nodev` y `noexec` en las particiones de los directorios personales de los usuarios (o en /home/usuario),lo que prohíbe la creación dispositivos de bloque o carácter y la ejecución de programas.
16. **user**: Permite a cualquier usuario del sistema montar o desmontar un dispositivo físico sin necesidad de ser root, esta opción es muy útil para dispositivos de uso frecuente, como cd-roms o disquetes. Pero también debe tenerse en cuenta, que el usuario que montó el dispositivo, solo el mismo usuario podrá desmontar el dispositivo. La opción opuesta es nouser.
17. **users**: indica que cualquier usuario puede montar y cualquiera también, puede desmontar el dispositivo.
18. **nouser**: debes ser el usuario root para montar el sistema de ficheros, nadie más que el superusuario podrá hacerlo.

Lo esencial del archivo Fstab

19. **defaults**: Esta opción es la unión de las opciones rw, suid, dev, exec, auto, nouser y async. Puedes consultar la ayuda del comando mount si necesitas más parámetros, aunque como ya dije estos son los más comunes, aunque tal vez no se apliquen a tus necesidades. Para consultar la ayuda del comando mount; abrir una Terminal, y ejecutar: **\$ man mount** (desgraciadamente esta ayuda está en inglés).
- V. **<dump>**: el comando dump, **es un comando estándar del sistema de ficheros Unix/Linux usado para respaldar información (back-ups)**; el valor de esta opción sirve para identificar los sistemas de ficheros que deberían incluirse en las operaciones de copia de seguridad cuando se ejecute el comando dump. Esta opción solo puede poseer el valor 0 o 1. Si su valor es 0 (o falta) indica al comando dump que no es necesario hacer copia de seguridad del sistema de ficheros asociado a la correspondiente línea de /etc/fstab. Si su valor es 1 se hará un backup con las opciones, creando backup por si surge algún tipo de problema, lo normal es tenerlo desactivado (0), ya que rara vez se produce un error.
- VI. **<pass>**: **Este campo le sirve al comando fsck (algo así como el scandisk en windows) para identificar los sistemas de ficheros cuya consistencia hay que comprobar en el siguiente reinicio, y el orden en el que hay que efectuar esa comprobación.**

Si su valor es 0 o falta el programa fsck asume que no es necesario comprobar el sistema de ficheros asociado a la línea en /etc/fstab , y el sistema no realizará nunca ninguna pasada para comprobar la integridad del dispositivo físico (normalmente ponemos 0 en el cdrom, dvd, disquet, pendrive y partición swap), y por ello es recomendable tenerlo en activado en las particiones:

- para “ / ” usamos 1
- para el resto de las particiones 2

El sistema realizará una pasada cada X desmontadas (en Guadalinex V4 creo que es cada 30 reinicios del sistema) o si el dispositivo ha sido desmontado incorrectamente para comprobar su integridad.

Más opciones en distintos tipos de dispositivos/particiones

1. **=errors=remount-ro**: Se pone en la línea de fstab correspondiente al **sistema raíz “ / ”**. Montará el sistema de archivos en modo solo lectura en caso de que haya algún problema con ellos, así se previene una potencial pérdida de datos por uso de un mal sistema de archivos. Si eso sucede en una de tus particiones, deberías probablemente reiniciar el sistema desde un livecd o disquet (si no es tu partición raíz, puedes reiniciar en modo de recuperación e intentar solucionar el problema) y luego ejecutar el comando fsck en el disco afectado.
- 2) Se ponen en la línea del fstab que hace referencia al **cdrom y/o dvd**:

=udf UDF (Universal Disk Format), es un sistema de archivos con estándar ISO 9660 propiedad de Adaptec que utiliza las grabadoras de CD/DVD como un dispositivo de almacenamiento lógico.

Este formato permite leer, escribir o modificar los archivos contenidos en discos CD/DVD regrabables (RW) del mismo modo que se hace en el disco duro, memorias USB o diskettes. Utiliza la tecnología de grabación por paquetes (Packet Printing) soportado por grabadoras CD-RW, DVD-RAM/RW, HD DVD y Blu-ray.

Al formatear un disco con UDF se pierden alrededor de 120 o 150 MB dependiendo de la versión. Por ejemplo un disco CD-RW de 700mb formateado con UDF permite utilizar tan solo de 550 a 570 MB para almacenar los documentos.

=iso9660 El estándar ISO 9660 es una norma publicada inicialmente en 1986 por la ISO, que especifica el formato para el almacenaje de archivos en los soportes de tipo disco compacto.

Lo esencial del archivo Fstab

El estándar ISO 9660 define un sistema de archivos para CD-ROM. Su propósito es que tales medios sean legibles por diferentes sistemas operativos, de diferentes proveedores y en diferentes plataformas, por ejemplo, MS-DOS, Microsoft Windows, Mac OS y Linux

3. En particiones Windows (fat, vfat,ntfs, pendrive)

A- en cualquiera de los tipos de particiones Windows:

Esto se suele poner en las particiones de windows 98 **fat y vfat**:

- **uid=valor**

Establece el identificador de usuario propietario de todos los archivos de la partición.

En linux, los usuarios son representados por un identificador de usuario, normalmente abreviado como uid.

Las características básicas son:

- ✓ Como mínimo los uid's deben estar comprendidos entre 0 y 32767.
- ✓ El superusuario debe tener siempre uid 0.
- ✓ Al usuario nobody siempre se le asigna por tradición el uid más alto posible (32767) (como oposición al superusuario).
- ✓ Recientemente, a los usuarios se les asigna un uid dentro del rango del sistema, 1-100, o entre en el rango 65530-65535.
- ✓ Los uid's entre 1 y 100 son reservados normalmente para que los use el sistema.
- ✓ La lista de todos los uid's de los usuarios se encuentran en el archivo /etc/passwd.

- **gid=valor o usuario**

Establece el identificador de grupo para todos los archivos de la partición.

El gid es un identificador de un grupo de usuarios. En los sistemas linux, los múltiples usuarios se pueden combinar para formar "grupos".

Los permisos del sistema de archivos se organizan en tres clases, del usuario, del grupo, y otros. El uso de grupos permite que las capacidades adicionales sean delegadas en una manera organizada, tal como el acceso a los discos, impresoras.

Este método también permite al superusuario delegar algunas tareas administrativas a los usuarios normales.

El gid, es un valor numérico usado para representar a un grupo específico.

La gama de los valores para un gid varía entre diversos sistemas; un gid puede estar 0 y 32767, con una restricción: el grupo de la conexión para el superusuario debe tener gid 0. Este valor numérico se utiliza para referir a grupos en los archivos del /etc/passwd y del /etc/group o sus equivalentes.

- **umask=valor o usuario**

Con el comando umask se puede definir con qué permisos por defecto se crearán los archivos en el momento de creación.

Por defecto el valor umask de root es 077, lo cual desactiva los permisos de lectura, escritura y ejecución para otros usuarios.

Por el contrario umask permite a un usuario que carece de estos permisos poder trabajar con la partición, es decir para que todos los usuarios puedan escribir en la partición, si su valor es 0, umask=0 (este umask=0 se suele por lo tanto poner en pendrives o discos duros externos)

Los permisos definen como los usuarios pueden o no trabajar con los archivos y directorios.

Los tres tipos de permisos son:

- **Read** (lectura)(r en los atributos del archivo o directorio): permite leer un archivo.

Lo esencial del archivo Fstab

- Para un directorio permite ver su contenido.
- **Write** (escritura) (w): Permite modificar un archivo.
Para un directorio indica que se pueden añadir o suprimir archivos.
- **Execute** (x): Permiso de ejecución si el archivo es binario.
Para un directorio significa que se puede ver los atributos de un archivo o entrar en él y en sus subdirectorios.

Un guión en la lista de atributos del archivo supone que el permiso no está activado:

rw-x indica que el archivo tiene permiso de lectura, de escritura, pero no de ejecución.

Estos tres permisos pueden aplicarse para el usuario, para el grupo al que pertenece el usuario o para el resto de usuarios, lo que hace un total de 9 atributos para cada archivo o directorio.

Antes de cambiar permisos en cualquier sistema de ficheros, esté seguro de que entiende lo que hace.

Nunca cambie permisos de un fichero simplemente porque parezca la forma fácil de hacer que algo funcione.

Siempre debe determinar por qué el fichero tiene esos permisos y propietario antes de modificarlos.

B- en windows ntfs

- **ntfs-3g:**

NTFS-3G Al contrario que el driver NTFS incluido en el kernel de Linux, tiene muy pocas limitaciones en cuanto a la escritura de archivos: permite crear, renombrar, mover o borrar archivos de cualquier tamaño en particiones NTFS, con la excepción de archivos comprimidos por NTFS o cifrados.

Tienes que tener instalados los paquetes ntfs-3g y ntfs-config.

- **locale=es_ES.UTF-8 :**

Cuando eliges 'Español' como idioma en la instalación ubuntu te asigna como 'locale' es_ES.UTF-8; como Guadalinex se instala en español también te asigna locale=es_ES.UTF-8

Si hay una partición de Windows en el disco duro el instalador la monta automáticamente durante la instalación y configura las opciones iocharset en el archivo /etc/fstab como iso-8859-15.

Éste no es el valor correcto y el sistema no puede visualizar bien los nombres de archivos y directorios, encontrando caracteres extraños en los nombres de algunos archivos / directorios de esa partición montada.

Este problema se debe al desacuerdo entre la codificación del sistema de la partición montada (Windows XP sigue usando codificación ISO8859-15) y la usada por el sistema operativo Linux (usa UTF-8 como juego de caracteres predeterminado) sobre la que se monta dicha partición.

La codificación usada en un partición montada y la del sistema sobre la que se monta, deben ser la misma para no tener problemas con determinados caracteres (ñ, €, acentos, etc).

Para corregir esto y que si que se puedan ver los archivos o los nombres de los directorios en el formato adecuado para el español, hay que poner la **opción iocharset=utf8**

Si hacemos un usuario nuevo en Linux, al iniciar sesión con él, no tendremos permiso para entrar en las particiones montadas distintas de la de Linux. Como pasa con las Ntfs.

Lo esencial del archivo Fstab

En /etc/fstab la línea correspondiente a la partición ntfs de windows sería:

```
/dev/hda1 /media/WindowsXP ntfs auto,user,exec,uid=1000,gid=1000 0 0
```

La configuración que establece esto, es:

```
uid=1000,gid=1000  
uid=identificador usuario  
gid=identificador grupo
```

Ello, indica que sólo el usuario con número 1000 y solo el grupo con número 1000 puede acceder a /media/WindowsXP.

El usuario que tiene asignado el número y grupo 1000 es 'usuario', que es con quien iniciamos Linux, luego de instalarlo.

Debemos cambiar uid=1000,gid=1000 por umask=000. De esta manera damos permiso de acceso libre a todos los usuarios.

Y la línea completa queda así:

```
/dev/hda1 /media/WindowsXP ntfs auto,user,exec,umask=000 0 0
```

C- en pendrives-usb

- **utf8:**

Esta opción es para que los archivos con nombres con caracteres especiales (ñ, letras con acentos como à y demás) se guarden en particiones Windows (particiones fat, vfat, ntfs) (discos duros internos, discos duros externos, pendrives-usb) de forma que se puedan leer bien sus nombres y no aparezcan en ellos caracteres "raros" (como un rombo en vez de una ñ).

Esto es porque como Guadalinex se instala en español, te asigna locale=es_ES.UTF-8 (Cuando eliges 'Español' como idioma en la instalación de ubuntu te asigna como 'locale' es_ES.UTF-8).

- **umask=0:**

Por defecto el valor umask de root es 077, lo cual desactiva los permisos de lectura, escritura y ejecución para otros usuarios.

Por el contrario umask permite a un usuario, que en principio carece de estos permisos, poder trabajar con la partición; es decir para que todos los usuarios puedan escribir en la partición, el valor de umask debe ser 0, umask=0 (este umask=0 se suele por lo tanto poner en pendrives o discos duros externos).