

Administración Linux: Sistema de Ficheros

En este módulo se explica algunas cuestiones de administración: jerarquía de sistema de ficheros, discos, particiones, sist...

Tabla de Contenidos

- 1 Introducción
- 2 Normas para la Jerarquía de Sistemas de Ficheros (FHS)
- 3 Discos y Particiones
- 4 Sistemas de archivos Ext3 y Ext4
- 5 Paginación y Procesos
 - 5.1 Espacios de paginación.
 - 5.2 Sistemas de archivos virtuales /proc y /sys
- 6 Discos redundantes (RAID)
- 7 Volúmenes lógicos
- 8 Sistemas de archivos remotos
 - 8.1 NFS (Network File System)
 - 8.2 SMB (Server Message Block)/CIFS (Common Internet File System)

Autor: Lina García Cabrera & Fernando Martínez

1 Introducción

La gestión adecuada del acceso a disco es otro de los aspectos importantes en el proceso de administración de sistemas operativos multiusuario y multitarea y es imprescindible mantener una estructura básica con un cierto nivel organizativo. El sistema operativo interactúa con los usuarios y las aplicaciones, y se hace necesario un modelo de seguridad dependiente de la forma en que se almacenan los ficheros en los dispositivos.

Un sistema de archivos puede verse desde dos categorías lógicas de ficheros:

- Archivos locales no compartibles o compartibles con otras máquinas.
- Archivos estáticos o variables.

Un **sistema de archivos** es un subárbol de directorios con un **directorio raíz** –que debe tener unos permisos acordes con las necesidades de acceso a sus archivos–, una **estructura lógica** de almacenamiento y un **punto de montaje** adecuado en el árbol de directorios global del servidor.

2 Normas para la Jerarquía de Sistemas de Ficheros (FHS)

Las **Normas para la Jerarquía de Sistemas de Ficheros (FHS, *Filesystem Hierarchy Standard*)** describen un conjunto de reglas que permiten, tanto a los usuarios como a los programas, predecir la localización de los ficheros y directorios instalados en el sistema.

3 Discos y Particiones

Todos los sistemas Unix –y, por lo tanto, todos los “dialectos” Linux– utilizan ficheros de dispositivos para acceder a los recursos de la máquina, almacenados en el directorio /dev. Sin embargo, **cada dialecto Unix tiene una notación diferente para identificar cada dispositivo de almacenamiento**. Aquí presentamos la más usual en sistemas Linux derivados tanto de Debian como de Red Hat:

- **Tipo de dispositivo** (sd para discos SCSI, hd para IDE).
- **Unidad** (a para el dispositivo 1, b para el 2, etc.).
- **Número de partición**.

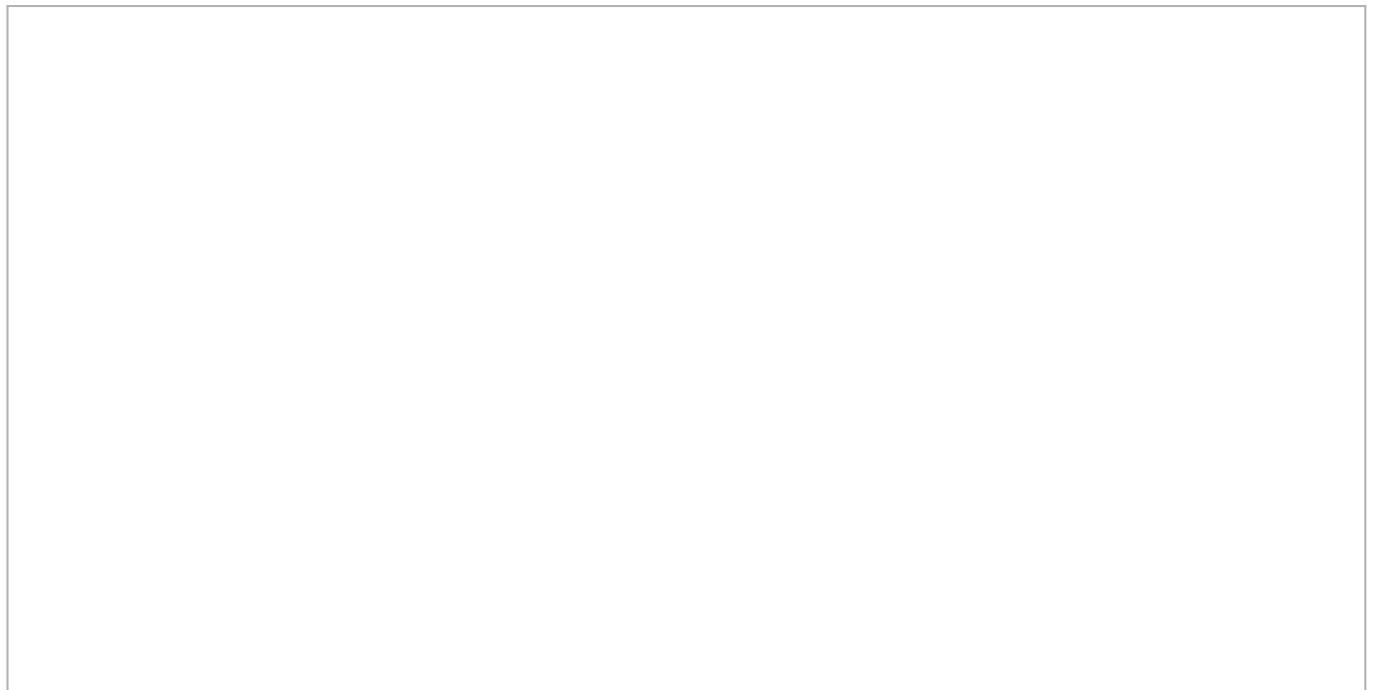
En caso de usar discos redundantes por *hardware*, el fichero del dispositivo se encuentra en un subdirectorio con el nombre del controlador **RAID** (por ejemplo, cciss para HP Smart Array) y su nombre tiene el siguiente formato:

- cN (nº de controlador, empezando por 0).
- dN (nº de disco, empezando por 0).
- pN (nº de partición, empezando por 1)..

Una **partición** es cada una de las subdivisiones que el gestor del sistema define en una unidad de disco del sistema, donde se almacena un determinado sistema de archivos o un espacio de paginación.

El administrador debe definir los distintos sistemas de archivos de su sistema, creando particiones en cada disco, teniendo en cuenta los recursos disponibles y la utilización principal que los usuarios harán de ellos.

El proceso de crear los sistemas de archivos básicos suele realizarse durante la instalación de la máquina, aunque pueden añadirse y ampliarse posteriormente.



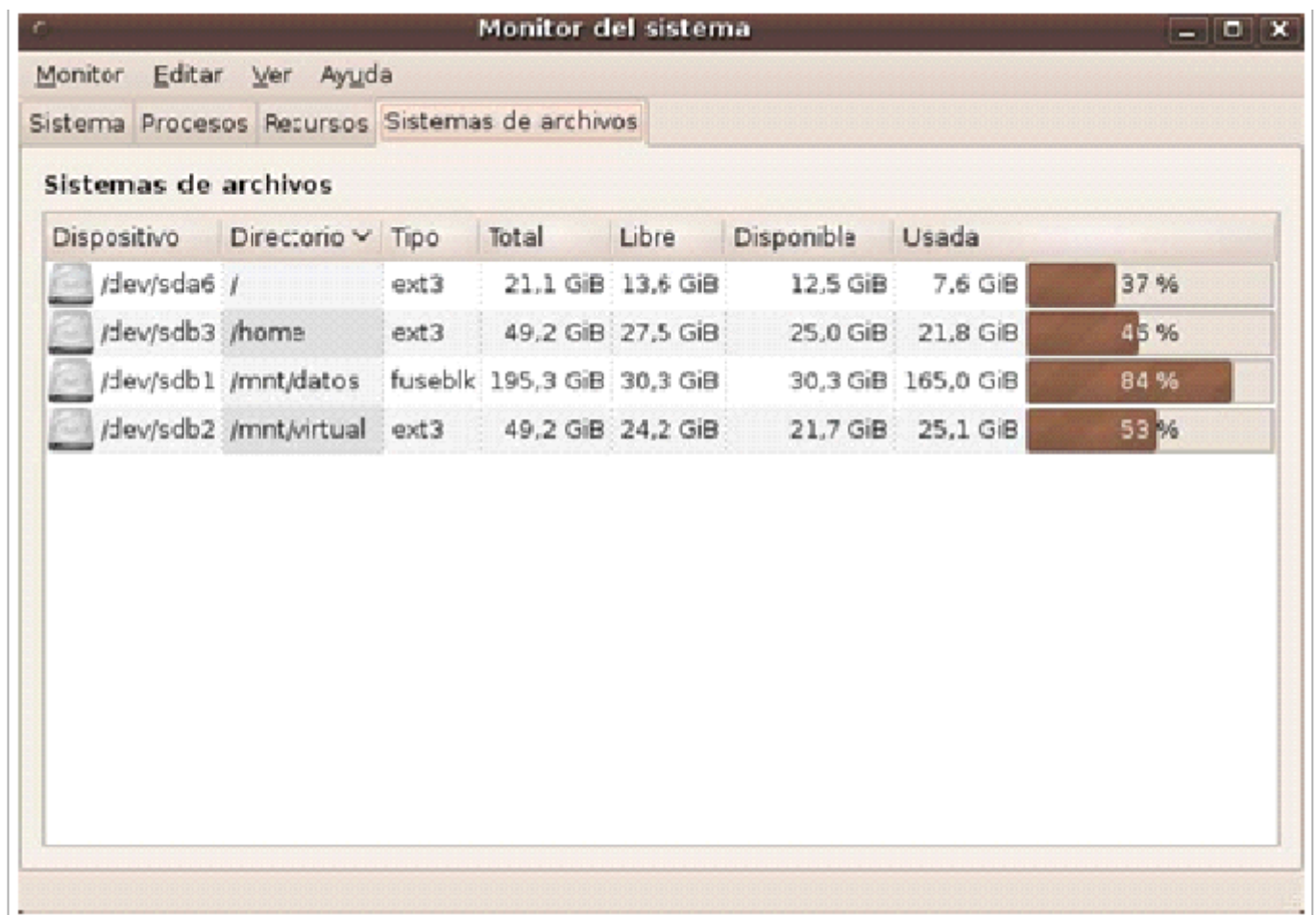


Ilustración 1 Particiones sobre dos discos, sda y sdb

4 Sistemas de archivos Ext3 y Ext4

Linux admite el montaje de distintos sistemas de archivos, tanto locales como remotos, ya que se ha programado una interfaz entre ellos y el núcleo, conocida como **Sistema de Archivos Virtual** (*Virtual File System*).

El sistema de archivos más utilizado hasta hace algunos años en Linux era el conocido como **Sistema de Archivos Extendido 2 (Ext2)**, que aumentaba las prestaciones de la primera versión, pero que seguía presentando problemas ante una caída inesperada del sistema, ya que necesitaba un largo proceso de comprobación y corrección.

Las modernas distribuciones Linux usan el **Sistema de Archivos Extendido 3 (Ext3)**, el cual incluye las siguientes mejoras:

- El diario de registros es la característica más importante, que mejora los procesos de revisión de integridad, ya que sólo se requiere la comprobación de dicho diario.
- Soporta mayores niveles de integridad de datos para evitar la corrupción del sistema de archivos, permitiendo elegir el tipo y el nivel de protección.
- Mayor flujo y mayor velocidad de accesos repetidos a datos.
- Fácil transición entre ext2 y ext3, sin necesidad de volver a formatear las particiones

El sistema de **archivos Ext4** incluye las siguientes mejoras con respecto a Ext3:

- Mayor tamaño del sistema de archivos (hasta 1 EB = 220 TB).
- Sin restricciones en el número de subdirectorios.

- Mayor velocidad de tratamiento de ficheros grandes mediante “extents”.
- Asignación previa de disco o asignación retardada.
- Comprobación del registro del sistema de archivos.
- Desfragmentación en directo sin necesidad de desmontar el sistema de archivos.
- Recuperación de ficheros borrados.
- Comprobaciones más rápidas del estado del sistema de archivos.
- Las marcas de tiempo cuentan con precisión de nanosegundos.
- Actualizable desde Ext3 (puede volverse a Ext3 perdiendo los “extents”).

La configuración permanente de los sistemas de archivos montados en un servidor se define en el fichero `/etc/fstab`, incluyendo datos sobre el dispositivo origen, el punto de montaje, el tipo del sistema de archivos, así como el conjunto de opciones de montaje, depuración y comprobación de la consistencia de los datos.

Etiquetado Tipo Opciones Volcado Orden

En `/etc/fstab` se encuentra una línea por dispositivo montando. Cada línea almacena la siguiente información:

1. **Etiqueta** o UUID de la partición, dispositivo o directorio remoto.
2. **Montaje**: Punto de montaje local.
3. **Tipo** de sistema de archivos(**ext3**, **ext4**, **swap**, **vfat**, **ntfs**, **nfs**, **cifs**, etc.).
4. **Opciones** de montaje (dependen del tipo de sistema de archivos).
5. Control de **volcado** automático de seguridad ante caídas del sistema.
6. **Orden** de comprobación de consistencia de datos durante el arranque del servidor (1 para /, incrementar en sistemas de archivos de distintos discos).

En la siguiente tabla se describen brevemente los órdenes más usuales para la manipulación de particiones y sistemas de archivos.

Orden	Descripción
fdisk	Manipulación de la tabla de particiones de un disco
mkfs	Formatea una nueva partición
mount	Monta un sistema de archivos en el árbol global de directorios
umount	Desmonta un sistema de archivos
tune2fs	Conversor entre sistemas de archivos ex2, ext3 y ext4.
parted	Gestión de particiones y sistemas de archivos

5 Paginación y Procesos

5.1 Espacios de paginación.

Un sistema operativo multiusuario y multitarea como Linux necesita una gran cantidad de memoria física para poder ejecutar todos los procesos. Los **espacios de paginación**

o **swapping** son particiones de disco que permiten ampliar virtualmente la memoria del sistema, guardando el estado de los procesos que en un determinado momento están a la espera de ser ejecutados, si la memoria física está agotada. Los factores principales que deben determinar el tamaño del espacio total de paginación son:

- La cantidad de memoria y de disco del sistema.
- El número de usuarios que tendrán acceso a la máquina.
- El número previsto de procesos/usuario.
- El número de servicios activos en el sistema.
- El número estimado de clientes/servicio.

Debido al crecimiento de la memoria en los nuevos servidores, la regla general es utilizar para paginación entre la mitad y el doble de la memoria física instalada. Ante casos de necesidad, el administrador puede ampliar la cantidad de paginación usando ficheros de disco que pueden ser posteriormente eliminados. La siguiente tabla describe las órdenes Linux usadas para manipular los espacios de paginación.

5.2 Sistemas de archivos virtuales /proc y /sys

Los **sistemas de archivos virtuales montados en /proc y en /sys están almacenados en memoria y contienen una jerarquía de ficheros y directorios especiales que mantienen el estado actual del núcleo del sistema Linux**, recopilando información sobre los dispositivos y los procesos en ejecución.

El siguiente ejemplo muestra el formato típico de la definición de ambos sistemas de archivos en el fichero /etc/fstab:

```
sysfs    /sys    sysfs    defaults    0 0
proc     /proc    proc     defaults    0 0
```

La mayoría de los ficheros virtuales de /proc aparecen con longitud 0, aunque pueden ser revisados como si fueran archivos de texto, algunos de ellos con gran cantidad de información.

En /proc hay una serie de directorios especiales que describen el estado actual de cada proceso en ejecución –denotados por el identificador del proceso (PID)–, incluyendo datos como: la línea de la orden ejecutada, los directorio raíz y de trabajo del proceso, estados de la memoria, de ejecución y de uso de los procesadores, las variables de entorno, etc.

Otros ficheros y directorios de interés son los que informan sobre procesadores, memoria, interrupciones, dispositivos, particiones, puntos de montaje, módulos del núcleo, parámetros de red, etc.

El **directorio especial /proc/sys contiene ficheros que sólo pueden ser modificados por el administrador para realizar cambios de configuración en el núcleo**, habilitando o desactivando ciertas características operativas. Debe tenerse gran precaución en la modificación de los archivos virtuales de /proc/sys.

La información completa sobre el sistema de archivos virtual /proc puede encontrarse en el paquete del código fuente del núcleo, que suele encontrarse en el subdirectorio /usr/src/VersionNúcleo/Documentation.

Por su parte, **el núcleo de Linux utiliza el sistema de archivos /sys para informar a las aplicaciones de usuario respecto de los objetos gestionados por el Kernel y las relaciones de dependencia entre ellos.**

La estructura principal de `/sys` agrupa información sobre las clases de dispositivos registrados, los buses físicos, los dispositivos conectados, los controladores y los módulos del núcleo.

6 Discos redundantes (RAID)

La **Matriz Redundante de Discos Independientes (RAID)** representa un conjunto de técnicas válidas para ahorrar costes o mejorar las prestaciones y la seguridad del acceso al almacenamiento masivo, combinando múltiples discos en un único dispositivo lógico.

El concepto principal de RAID es dividir los datos en ciertos trozos y distribuirlos en los dispositivos de la matriz, según el nivel de necesidad. Durante el proceso de lectura se sigue un algoritmo inverso de reconstrucción.

Las principales características del uso de discos en RAID son:

- Aumentar la velocidad de acceso a los datos.
- Incrementar la capacidad de almacenamiento, combinando discos de menor capacidad en un único disco lógico mayor.
- Mejorar la tolerancia a fallos de los discos.

Existen dispositivos y controladores preparados para realizar técnicas RAID en su propio hardware, lo que aumenta las prestaciones y el precio final de la máquina. Los nuevos sistemas operativos son aptos para realizar este cometido bajo software, si bien resultará más ineficiente.

Existen diversos **niveles RAID**, cada uno con sus características:

- **RAID0** permite combinar más de un disco y percibirlo como uno sólo. Al contar físicamente con dos discos la lectura de grandes archivos se optimiza, ya que la controladora RAID procurará distribuir el archivo entre ambos dispositivos.
- **RAID1**, donde dos discos idénticos actúan el uno como espejo del otro, de tal modo que si uno de los dos sufriera algún tipo de corrupción en sus datos, será posible recuperarlos a partir del otro.

Si las particiones que vayan a utilizarse para el RAID por software van a montarse sobre sistemas de archivos esenciales para Linux, deben definirse durante el proceso de instalación del sistema operativo.

Utilidades como **Disk Druid** permiten definir particiones RAID, asociarles el nivel de redundancia y generar el disco lógico (**metadispositivo**). La definición de la matriz se encuentra en el fichero `/etc/raidtab`.

El siguiente ejemplo muestra la configuración de un metadispositivo `md0` de tipo RAID 1 (espejo) formado por las particiones `sda1` y `sdb1` de 2 discos, que contienen un sistema de archivos de tipo **Ext3** montado sobre el directorio raíz.

```
$ cat /etc/raidtab
```

```
raiddev    /dev/md0
```

```
raid-level    1
```

```
nr-raid-disks    2
```

```
chunk-size    64k
```

```
persistent-superblock 1
```

```
nr-spare-disks    0
```

```
device    /dev/sda1
```

```
raid-disk    0
```

```
device    /dev/sdb1
```

```
raid-disk    1
```

```
$ df -h /dev/md0
```

S.ficheros	Tamaño	Usado	Disp	Uso%	Montado en
/dev/md0	40G	21,5G	19,5G	51%	/

7 Volúmenes lógicos

Los volúmenes lógicos son técnicas de gestión de almacenamiento disponibles a partir de la versión 2.4 del núcleo de Linux –heredadas del sistema operativo AIX, el dialecto Unix de IBM– que permiten redimensionar las particiones y distribuirlas en varios discos.

En algunas distribuciones de Linux puede existir la restricción impuesta por el **Gestor de Volúmenes Lógicos (LVM)** de que el directorio /boot deba encontrarse en una partición real y no formar parte de ningún volumen lógico.

Es obligatorio definir los volúmenes lógicos en el proceso de instalación cuando éstos vayan a almacenar sistemas de archivos propios del sistema.

El Gestor de Volúmenes Lógicos consta de tres elementos fundamentales:

Volumen físico Estructura que representa a un disco.

Volumen lógico Estructura equivalente a un sistema de archivos Linux.

Grupo de volúmenes Conjunto de varios volúmenes lógicos que pueden almacenarse en varios volúmenes físicos. Así, un disco puede contener varios sistemas de archivos y un sistemas de archivos puede estar grabado en varios discos.

El instalador del sistema debe seguir los siguientes pasos:

1. Si la distribución de Linux es antigua, crear una partición normal de tipo **Ext3** para el directorio /boot, ya sea incluido en el directorio raíz o en una partición propia.
2. Definir un volumen físico en cada disco.
3. Crear los grupos de volúmenes conjuntando adecuadamente los volúmenes físicos.
4. Definir los volúmenes lógicos de cada grupo de volúmenes, asignando para cada uno de ellos su tamaño inicial y su punto de montaje.

Es recomendable dejar algún espacio sin asignar para poder ampliar las particiones

que lo necesiten.

El sistema incluye una gran variedad de órdenes para gestionar cada uno de los componentes del gestor de volúmenes lógicos. La tabla describe la mayoría de estas órdenes según su función.

Operación	Volumen físico	Grupo de volúmenes	Volumen lógico
Crear	<code>pvccreate</code>	<code>vgcreate</code>	<code>lvcreate</code>
Eliminar	<code>pvremove</code>	<code>vgremove</code>	<code>lvremove</code>
Comprobar estado	<code>pvscan</code>	<code>vgscan</code>	<code>lvscan</code>
Cambiar tamaño	<code>pvresize</code>	<code>vgresize</code> <code>vgextend</code> <code>vgreduce</code>	<code>lvresize</code> <code>lvreduce</code> <code>lvextend</code>
Mostrar información	<code>pvs</code>	<code>vgs</code>	<code>lvs</code>
Mostrar atributos	<code>pvdisplay</code>	<code>vgdisplay</code>	<code>lvdisplay</code>

El siguiente cuadro muestra un ejemplo real usado para definir un grupo de volúmenes `vg0` con dos discos que usan un controlador **RAID** por *hardware*, en donde se definirán tres volúmenes lógicos; posteriormente, el administrador podrá usar cada uno de ellos para montar los sistemas de archivos del servidor.

```
# pvcreate /dev/cciss/c1d0 /dev/cciss/c1d1

# pvs

PV VG Fmt Attr PSize PFree
/dev/cciss/c1d0 lvm2 -- 1,36T 1,36T
/dev/cciss/c1d1 lvm2 -- 698,56G 698,56G

# vgcreate vg0 /dev/cciss/c1d0 /dev/cciss/c1d1

# vgs

VG #PV #LV #SN Attr VSize VFree
vg0 2 0 0 wz--n- 2,05T 2,05T

# lvcreate -L 300G vg0
# lvcreate -L 700G vg0
# lvcreate -L 700G vg0 # lvs

LV VG Attr LSize Origin Snap% Move Log Copy%
lvol0 vg0 -wi-a- 300,00G
lvol1 vg0 -wi-a- 700,00G
lvol2 vg0 -wi-a- 700,00G

#vgs

VG #PV #LV #SN Attr VSize VFree
```


8 Sistemas de archivos remotos

La conexión remota a otros ordenadores supone una gran ventaja en el proceso de compartir información. Los sistemas de archivos remotos permiten almacenar la información en un único nodo central y hacerla accesible a los distintos clientes, posibilitando la movilidad del usuario.

Los sistemas de archivos remotos más utilizados actualmente son: [NFS](#), [SMB/CIFS](#)

8.1 NFS (Network File System)

El **Sistema de Archivos en Red (NFS)** fue creado por Sun Microsystems para SunOS – su dialecto Unix –, usando las técnicas de **Llamadas a Procedimientos Remotos (RPC)**. NFS permite acceder a los archivos en nodos remotos exactamente en la misma manera que si fueran locales, de un modo completamente transparente al cliente e independientemente de la arquitectura del servidor.

IETF especifica en su RFC 3530 la versión 4 de NFS (**NFSv4**), redefiniendo completamente el protocolo e incluyendo mejoras como bloqueo de uso de ficheros, negociación de seguridad, ACLs, interoperabilidad entre plataformas, internacionalización, etc.

Internet Engineering Task Force (IETF) (en español Grupo Especial sobre Ingeniería de Internet¹) es una organización internacional abierta de normalización, que tiene como objetivos el contribuir a la ingeniería de Internet, actuando en diversas áreas, como transporte, encaminamiento, seguridad. Fue creada en EE. UU. en 1986. La IETF es mundialmente conocida por ser la entidad que regula las propuestas y los estándares de Internet, conocidos como RFC.

La siguiente tabla describe los servicios que deben activarse en los ordenadores servidor y cliente NFS.

Servicio	Descripción
portmap	Servicio de control de RPC
rpc.mountd	Control de montaje del cliente NFS
rpc.statd	Monitor del estado de la red (NSM) que notifica el reinicio del servidor NFS.
rpc.rquotad	Provee información de cuotas para usuarios remotos.
rpc.nfsd	Servidor NFS.

El fichero `/etc/exports` contiene la configuración **NFS** en el servidor. Existe una línea en el fichero para cada directorio exportado. Sus campos son:

- Directorio local a exportar.
- Nombre o IP del cliente (soporta comodines en nombre y en dominios).
- Opciones de exportación: sólo lectura (`ro`), lectura/escritura (`rw`), evitar acceso privilegiado para el root del cliente (`root_squash`), acceso privilegiado para root (`no_root_squash`), etc.

El cliente NFS puede configurar la importación de directorios en su fichero `/etc/fstab` o montarlo directamente con la orden **mount**.

```
# mount -t nfs4 Servidor:Directorio PuntoMontaje [Opciones]
```

8.2 SMB (Server Message Block)/CIFS (Common Internet File System)

El **Sistema de Archivos Común para Internet (CIFS, Common Internet File System)**

provee una serie de mecanismos abiertos e independientes de la plataforma utilizada, para que sistemas clientes soliciten servicios de ficheros a otras máquinas a través de la red.

Este protocolo es una implementación del conocido como **Bloque de Mensajes del Servidor (SMB, Server Message Block)**, usado principalmente por ordenadores con Windows.

Microsoft ha redefinido nuevos dialectos del protocolo (SMB2 lanzado con Windows Vista y SMB2.1 con Windows 7), para mejorar el rendimiento y reducir la complejidad de las comunicaciones. Las características principales de CIFS son:

- Acceso a ficheros, permitiendo compartir información en lectura y escritura.
- Acceso bloqueado y desbloqueado tanto a ficheros como a registros.
- Notificación de cambios en ficheros y directorios.
- Inclusión de atributos extendidos.
- Independencia del protocolo de resolución de nombres.

Las Extensiones de CIFS para UNIX son normas de reciente creación y sólo están implementadas en las versiones 2.6 de los servicios de ficheros del núcleo de Linux, mientras que los antiguos necesitan ser recompilados o generar un módulo propio para la gestión de clientes CIFS, aunque soportan el montaje de sistemas de archivos SMBFS.

El servidor de ficheros puede ser una máquina con sistema operativo Windows (a partir de NT) o con Linux y el servicio **Samba** activado.

Samba es una implementación libre del protocolo de archivos compartidos de Microsoft Windows (antiguamente llamado SMB, renombrado recientemente a CIFS) para sistemas de tipo UNIX.

En ambos casos, deben ser configurados los recursos que van a ser exportados. Cada distribución de Linux incluye una serie de paquetes con las herramientas básicas para el control de sistemas de archivos CIFS/SMB, los clientes para acceso a los recursos o el servidor de ficheros Samba. La tabla muestra los mandatos usados por el cliente Samba.

Orden	Descripción
smbclient	Cliente Samba con interfaz similar al cliente Ftp
smbpasswd	Permite cambiar la clave remota del usuario
smbcquotas	Gestiona las cuotas en recursos NTFS
smbcacs	Gestiona la lista de control de accesos (ACL) a los ficheros
smbpool	Envía un fichero a una cola de impresión remota
net	Herramienta de administración de Samba y de servidores remotos
pdbedit	Herramienta de gestión de la base de datos de usuarios de Samba
findsmb	Lista las máquinas que responden a una petición SMB en una subred
mount.cifs	Montador de sistemas de archivos CIFS
umount	Desmontador general de sistemas de archivos.