

Manual básico NFS

Enviado por Zx80 el Sáb, 2010-02-20 13:37. ::

- [Magazines, boletines y documentación libre](#)



El sistema NFS (Network File System) es un sistema de intercambio de archivos en una red, permitiendo a una máquina remota acceder a otra como si fuese una partición local. Es muy útil para máquinas que necesitan los mismos archivos porque así no hay que copiarlos a todas, sino solo a una, o también para evitarnos caminatas de una máquina a otra a copiar

los archivos.

Este manual, que he creado a partir de mi propia experiencia hace 5 minutos está indicado para gente con pocos conocimientos (y si yo he podido.... ;-)).

En mi caso, tengo instalado Xubuntu. En otras distribuciones puede que el nombre de los paquetes cambien algo, pero poniendo a buscar nfs en el gestor de instalación de cada uno nos saldrá el nombre adecuado.

INSTALANDO

Los paquetes necesarios para montar el servidor NFS son:

nfs-kernel-server

nfs-common

portmap

Portmap no debe ejecutarse en modo loopback o no funcionará bien. Por defecto ya lo carga bien, pero para asegurarnos, añadimos la opción "*do =not= bind loopback*" en el ***/etc/default/portmap*** aunque mejor ejecutamos en la terminal y como root:

```
# dpkg-reconfigure portmap
```

A la pregunta que nos hará en el entorno ncurses le decimos "no".

Dpkg-reconfigure automáticamente reinicia el portmap pero si no fuese así se puede hacer de forma manual mediante el comando:

```
# /etc/init.d/portmap restart
```

o también:

```
# restart portmap.
```

Asegurando el protocolo.

Al tener portmap corriendo y varios servicios a su cargo, cualquiera podría meterse en nuestra máquina y tendría todo un mundo para chafardear, por lo que un poco de seguridad tampoco nos iría mal. Miraremos si portmap está corriendo el servicio nfs:

```
# rpcinfo -p
```

Lo que nos devuelve:

```
100000 2 tcp 111 portmapper
100000 2 udp 111 portmapper
100024 1 udp 48711 status
100024 1 tcp 36660 status
100021 1 udp 39385 nlockmgr
100021 3 udp 39385 nlockmgr
100021 4 udp 39385 nlockmgr
100021 1 tcp 34721 nlockmgr
100021 3 tcp 34721 nlockmgr
100021 4 tcp 34721 nlockmgr
100003 2 udp 2049 nfs
100003 3 udp 2049 nfs
100003 4 udp 2049 nfs
100003 2 tcp 2049 nfs
100003 3 tcp 2049 nfs
100003 4 tcp 2049 nfs
100005 1 udp 57480 mountd
100005 1 tcp 58195 mountd
100005 2 udp 57480 mountd
100005 2 tcp 58195 mountd
100005 3 udp 57480 mountd
100005 3 tcp 58195 mountd
```

Comprobamos que nfs está corriendo y nos vamos a dos archivos que son el /etc/hosts.allow y /etc/hosts.deny para especificar reglas de acceso. Yo tengo puesto lo siguiente:

/etc/hosts.deny:

```
portmap:ALL
lockd:ALL
mountd:ALL
statd:ALL
```

/etc/hosts.allow:

```
portmap:192.168.0.0/255.255.255.0
lockd:192.168.0.0/255.255.255.0
mountd:192.168.0.0/255.255.255.0
statd:192.168.0.0/255.255.255.0
```

El primer archivo deniega a todos (ALL) el acceso a los servicios y el segundo permite solo a la red local acceder a ellos.

CARPETAS A COMPARTIR

Debemos especificar las carpetas que queremos compartir en el otro equipo. Para ello se edita

el archivo **/etc/exports**.

Yo en el tengo una carpeta llamada **/media/store** donde almaceno fotos, pelis, documentos, y la quiero hacer accesible desde mi portátil. Además también me interesa compartir el **/home**, así que el sobremesa será el servidor y el portátil será el cliente.

Podría interesarme que cualquier máquina que se conecte a mi red local pueda, si lo desea, conectarse a mi sobremesa poniendo el rango completo de IPs de mi red, pero solo me interesa que sea el portátil el que se conecta, al cual siempre se le asigna la IP 192.168.0.12:

```
/media/store 192.168.0.12/24(rw,async,no_root_squash)
```

```
/home/zx80 192.168.0.12/24(rw,async,no_root_squash)
```

Las opciones de exports son las siguientes:

- **()**: Son las opciones que trae por defecto.
- **ro**: El directorio se exportará como solo lectura (opción por defecto).
- **rw**: El directorio se exportará como lectura/escritura.
- **root_squash**: Los accesos desde el cliente con UID=0 (root) se convierten en el servidor en accesos con UID de un usuario anónimo (opción por defecto).
- **sync**: Modo de sincronización de datos. Se verificarán los datos al copiarlos, algo lento pero más seguro.
- **async**: Es el segundo modo de sincronización de datos. Es más veloz pero pueden haber pérdidas de datos en caso de caída de la conexión.
- **no_root_squash**: Se permite el acceso desde un UID = 0 sin conversión. Es decir, los accesos de root en el cliente se convierten en accesos de root en el servidor.
- **all_squash**: Todos los accesos desde el cliente (con cualquier UID) se transforman en accesos de usuario anónimo.
- **anonuid, anongid**: Con las opciones **root_squash** y **all_squash**, el acceso anónimo se efectúa con el UID y GID primario del usuario denominado **nobody** si éste existe en el servidor (opción por defecto). Si queremos usar otro nombre, **anonuid** y **anongid** establecen respectivamente qué uid y gid tendrá la cuenta anónima que el servidor utilizará para acceder contenido del directorio.
- **noaccess**: Niega el acceso al directorio especificado, útil para evitar la recursión en la carpeta exportada.

Yo he puesto la opción *no_root_squash* porque si no debo dar permisos y retocar varios archivos de sistema para poder tener acceso, ya que de otra forma, aunque pueda montar las carpetas no podré acceder a ellas ni como root. Una vez echos los cambios, para que el servidor NFS los reconozca ejecutamos el comando:

```
# exportfs -ra
```

Este comando nos devuelve el estado por cada carpeta exportada:

```
exportfs: /etc/exports [1]: Neither 'subtree_check' or 'no_subtree_check' specified for export "192.168.0.0/24:/home/zx80".
```

```
Assuming default behaviour ('no_subtree_check').
```

NOTE: this default has changed since nfs-utils version 1.0.x

exportfs: /etc/exports [2]: Neither 'subtree_check' or 'no_subtree_check' specified for export "192.168.0.0/24:/media/store".

Assuming default behaviour ('no_subtree_check').

NOTE: this default has changed since nfs-utils version 1.0.x

ACTIVANDO EL SERVIDOR

Pues una vez echo todo ahora toca entrar en acción. Lo primero que haremos es activar el servidor mediante el comando:

```
# /etc/init.d/nfs-kernel-server restart
```

Esto nos devuelve no solo el reactivado del demonio NFS, sino también un actualizado de /etc/exports, como si hubiéramos ejecutado `# exportfs -ra`:

** Stopping NFS kernel daemon [OK]*

** Unexporting directories for NFS kernel daemon... [OK]*

** Exporting directories for NFS kernel daemon...*

exportfs: /etc/exports [1]: Neither 'subtree_check' or 'no_subtree_check' specified for export "192.168.0.0/24:/home/zx80".

Assuming default behaviour ('no_subtree_check').

NOTE: this default has changed since nfs-utils version 1.0.x

exportfs: /etc/exports [2]: Neither 'subtree_check' or 'no_subtree_check' specified for export "192.168.0.0/24:/media/store".

Assuming default behaviour ('no_subtree_check').

NOTE: this default has changed since nfs-utils version 1.0.x

** Starting NFS kernel daemon [OK]*

ACTIVANDO EL CLIENTE

Para activar el cliente, en este caso el portátil, solo debemos crear un nombre de carpeta que queramos para montar las carpetas remotas en nuestra máquina:

```
# mkdir /mnt/zx80-home
```

```
# mkdir /mnt/zx80-store
```

y acto seguido hacer el montaje:

```
sudo mount -t nfs 192.168.0.10:/media/store /mnt/zx80-store
```

```
sudo mount -t nfs 192.168.0.10:/home/zx80 /mnt/zx80-home
```

La dirección 192.168.0.10 es la del servidor (mi sobremesa) y cada uno deberá cambiarla por la suya. Si no se sabe se puede mirar mediante el comando ifconfig.

Ahora ya podremos entrar en las carpetas como usuario y copiar, borrar, etc, los archivos que queramos.

Evidentemente, si también queremos que desde el servidor podamos entrar al cliente, se pueden seguir los pasos para montar un servidor en el cliente (variando los datos, claro) para que así haya reciprocidad a la hora de compartir.

Y se me olvidaba, si queréis que se monte automáticamente, solo debéis añadir unas líneas al **/etc/fstab**:

```
192.168.0.10:/home/zx80      /mnt/zx80-home      nfs    timeo=14,intr 0    0
192.168.0.10:/media/store  /mnt/zx80-store     nfs    timeo=14,intr 0    0
```