

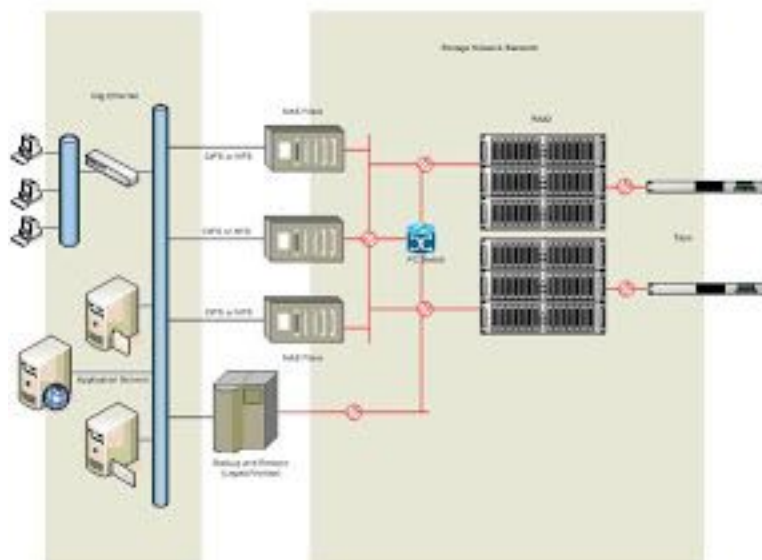
NFS

¿Qué es NFS?

NFS (Network File System o Sistema de Ficheros de Red) es una solución ofrecida por los sistemas operativos UNIX/Linux para compartir archivos en redes TCP/IP. Fue desarrollado por SUN Microsystems bajo licencia GPL, lo que facilitó que se convirtiera en un estándar a la hora de compartir archivos en red.

NFS nos permite montar un directorio de una máquina remota en un directorio de nuestra máquina como si fuera una partición local.

Mediante NFS, podemos compartir un directorio que se encuentra en un dispositivo de la red con otros ordenadores y dispositivos. El ordenador donde está el directorio es el servidor, mientras que los dispositivos que se conectan a este servidor son los clientes.



Precisamente, un sistema NFS, es perfecto para un NAS o Networked Attached Storage en un medio Linux. NFS es un protocolo nativo en Linux, justo lo contrario que Samba que utiliza el protocolo SMB desarrollado por Microsoft. Tanto Mac OSX como Windows 7 tienen un buen soporte para NFS. Por todo ello NFS es ideal para compartir directorios en red local.

NFS se compone de un servidor y un cliente, es necesario instalar las herramientas necesarias, tanto en el ordenador donde se encuentra el directorio y que actuará como servidor, como en los otros ordenadores que actuarán como clientes.

El servidor

Para instalar las herramientas necesarias en el ordenador que actuará como servidor, ejecutaremos la siguiente orden,

```
sudo apt install nfs-kernel-server
```

El siguiente paso es definir los directorios que queremos compartir. Para esto editamos el archivo **/etc/exports**. Así por ejemplo si queremos compartir el directorio compartida, tendríamos que configurarlo de la siguiente forma:

```
/home/david/compartida 192.168.10.0/24(rw,no_subtree_check,async)
```

Con esto, hemos compartido el directorio en cuestión con toda nuestra red (192.168.10.0/24), indicando que es de lectura y escritura (**rw**), no realiza comprobación de subdirectorios (**no_subtree_check**) lo que en ocasiones mejora el rendimiento, no es necesario que todas las escrituras se completen (**async**) aunque esto puede conllevar una pérdida de datos en caso de que se produzca un corte de conexión.

Las opciones más significativas son:

- **ro|rw**: el directorio será compartido en solo lectura (ro) y es la opción por defecto. El directorio será compartido en lectura y escritura (rw).
- **sync|async**: sync comunica al usuario los cambios realizados sobre los archivos cuando realmente se han ejecutado y es la opción recomendada. La opción async mejora el rendimiento y agiliza el funcionamiento del servicio, pero puede generar archivos corruptos si se produce algún tipo de fallo en el servidor.
- **no_subtree_check**: permite que no se compruebe el camino hasta el directorio que se exporta, en el caso de que el usuario no tenga permisos sobre el directorio exportado.

- **root_squash | no_root_squash | all_squash**
 - **root_squash** indica que un usuario identificado como root tendrá acceso al directorio compartido sólo con privilegios de usuario anónimo. De esta forma se ha degradado al root al usuario local de privilegios más bajos protegiendo así los archivos en el servidor NFS. Esta opción se conoce también con el nombre de 'aplastamiento del root'. Para el resto de usuarios se intenta conservar su UID y GID en el servidor.
 - **no_root_squash** desactiva la opción anterior, es decir, los accesos realizados como root desde el cliente serán también de root en el servidor NFS.
 - **all_squash** indica que todos los clientes, incluido root, tendrán acceso al directorio con privilegios de un usuario anónimo. No se mantienen los UID y GID de ningún usuario.
 - Si se utiliza alguna de las opciones squash podemos indicar con qué UID y GID de usuario se quiere que se acceda, en lugar del anónimo. En este caso hemos de indicar a continuación de la opción squash lo siguiente:
 - (rw,all_squash,anonuid=1002,anongid=1002)
 - Y significa que la conexión del cliente NFS se hará con los UID y GID 1002.

Veamos algunas posibles configuraciones:

```
/home/david/compartida (rw) #Comparte con cualquier equipo  
/home/david/compartida 192.168.10.0/24(ro) #Sólo lectura
```

Una vez configurado el servidor, es necesario reiniciar el servicio para que funcione correctamente,

```
sudo service nfs-kernel-server restart
```

El cliente

Ahora nos toca instalar y configurar el cliente en nuestro equipo Ubuntu. Esto es tan sencillo, como ejecutar la siguiente orden en un terminal:

```
sudo apt install nfs-common
```

Por último, nos queda montar el directorio remoto en nuestro equipo, así por ejemplo, si el servidor se encuentra en la IP 192.168.10.126, y el directorio que queremos montar es **/home/david/compartida**, lo primero que debemos hacer es crear el directorio donde va a ser montado en nuestro equipo o utilizar uno existente:

```
sudo mkdir -P /media/partner
```

A continuación solo nos queda ejecutar la siguiente orden en un emulador de terminal, para poder acceder al recurso remoto:

```
sudo mount -t nfs 192.168.10.126:/home/david/compartida /media/partner
```

A partir de este momento, podemos comprobar que nuestra carpeta **/media/partner** contiene la información de la carpeta **/home/david/compartida** del servidor. Si disponemos de permisos de lectura y escritura, podemos incluso crear o modificar los archivos dentro de nuestra carpeta y los cambios se estarán guardando realmente en la carpeta del servidor.

Para realizar el montaje, debemos hacerlo sobre una carpeta existente en nuestro sistema, si dicha carpeta de nuestro sistema contiene archivos, éstos no estarán accesibles ya que la carpeta nos mostrará los archivos remotos.

Si al intentar montar la carpeta NFS, obtenemos algún error, puede ser por una de estas tres razones: por un problema en la red, un problema en el servidor o un problema en el cliente.

Para averiguar si el problema es del servidor o no, podemos intentar montar por NFS la carpeta en el propio servidor, usando la IP 127.0.0.1. Si funciona entonces el problema estará en la red o en el cliente. Si hacemos ping del servidor al cliente y no hay cortafuegos, el problema será en el cliente.

Para desmontar una carpeta, tan solo debemos ejecutar el comando `umount` seguido del nombre de la carpeta en la que está montada, ejemplo:

```
sudo umount /media/partner
```

Si deseamos que nuestro PC monte siempre de forma automática una carpeta compartida por NFS cuando iniciemos nuestro Linux, existe la posibilidad de añadir en el archivo `/etc/fstab` una línea, como por ejemplo:

```
# Montaje automático al iniciar el PC  
#Añadir en /etc/fstab  
ip-del-servidor:/fotos /fotos-servidor nfs
```

De esta manera, cuando arranquemos nuestro PC, la carpeta `/fotos` del servidor quedará automáticamente montada sobre nuestra carpeta `/fotos-servidor` y no tendremos que ejecutar el comando `mount` para nada.

Ver recursos compartidos de una máquina

Si queremos ver qué directorios tiene compartidos una máquina vía NFS, ejecutamos el comando `showmount` en la máquina cliente con el parámetro `-e`:

```
$ showmount -e IPSERVIDOR
```

Por ejemplo, si tenemos un servidor NFS cuya IP es `192.168.1.2` y queremos ver, desde nuestra máquina cliente, qué directorios tiene exportados, ejecutaremos el comando:

```
$ showmount -e 192.168.1.2
```

NOTA

Los permisos de compartición por NFS no excluyen a los permisos del sistema unix sino que prevalecen los más restrictivos. Si una carpeta está compartida con permiso NFS de lectura y escritura pero en los permisos del sistema solo disponemos de permiso de lectura, no podremos escribir. Si una carpeta está compartida con permisos NFS de lectura y disponemos de permisos de lectura y escritura en el sistema, tampoco podremos escribir. Para poder escribir necesitaremos disponer permiso de lectura y escritura tanto en los permisos del sistema como en los permisos de compartición NFS. De igual forma, si compartimos la carpeta `/home` con permisos de lectura y escritura pero el usuario pepe solo tiene acceso a la carpeta `/home/pepe`, no podrá acceder a ninguna otra carpeta dentro de `/home` ya que los permisos del sistema se lo impedirán.