

## P10. Servidor NFS<sup>1</sup>

**NFS, (Network File System)** o Sistema de Ficheros en Red permite acceder a los ficheros remotos exactamente igual que si fueran locales. El acceso a los ficheros es totalmente transparente al cliente, funcionando con muchas arquitecturas de servidores.

Se recomienda utilizar **NFS dentro de una red local** detrás de un corta-fuegos que permita los accesos solo a equipos que integren la red local, nunca para compartir sistemas de archivos a través de Internet. Al no contar con un sistema de autenticación por contraseñas, es un servicio susceptible del ataque a diferencia de SAMBA que si exige la autenticación por contraseñas. Las ventajas de un servidor **NFS** son:

- Los datos accedidos por todo tipo de usuarios pueden mantenerse en un nodo central, con clientes que montan los directorios en el momento de arrancar. Por ejemplo, puede mantener todas las cuentas de usuario en una máquina, y hacer que las demás monten dichas cuentas en su directorio /home por **NFS**. Si además se instala **NIS (Network Information Service)**, los usuarios podrían entrar y trabajar de forma transparente en cualquiera de las máquinas.
- Los datos que consumen grandes cantidades de espacio de disco pueden mantenerse en un nodo.
- Los datos de administración pueden también mantenerse en un solo nodo. Ya no será necesario usar rcp para instalar el mismo fichero en 20 máquinas distintas.

### Desarrollo

Crearemos un **clon enlazado** de la plantilla de Debian que llamaremos **Debian-NFS** y un cliente que sería un **clón enlazado** llamado **Windows7-NFS** (o Windows 10 pero debe ser Ultimate)

Se debe **configurar un servidor NFS en Debian**. A continuación los usuarios del sistema cuando se conecten desde un equipo cliente en Windows deberán montar su directorio de Debian en el equipo desde el cual acceden de forma transparente con permisos **rw**.

### Pasos a seguir:

- Los paquetes a instalar para las dos versiones son los mismos: *nfs-kernel-server* y *nfs-common* (este último suele estar instalado por defecto)

---

<sup>1</sup> Se debe configurar las MV en VirtualBox como red interna.

- El fichero básico de configuración es el mismo en las dos versiones: */etc/exports*

#### Configuración de directorios y fichero export

- Los exports de NFSv4 deben residir en un pseudodirectorio, donde los directorios reales a exportar se montan con la opción `--bind`, por ejemplo para exportar `/home`

```
# mkdir /export
```

```
# mkdir /export/home
```

```
# mount --bind /home /export/home
```

- La opción `bind` permite remontar un directorio en otro sitio. Para que este montado permanezca, añadir al ***fstab*** la siguiente línea:

```
/home /export /home none bind 0 0
```

- Fichero `/etc/exports` en NFSv4

```
/export 193.144.84.0/24(rw,fsid=0,crossmnt,no_subtree_check,sync)
```

```
/export/home 193.144.84.0/24(rw,no_subtree_check,root_squash,sync)
```

Iniciar el demonio y comprobar que funciona:

```
#service nfs-kernel-server start
```

- Comprobar los directorios exportados con *showmount*

```
# showmount --exports localhost
```

***showmount*** muestra información de un servidor NFS: directorios que exporta, directorios montados por algún cliente y clientes que montan los directorios

- Podemos ver las estadísticas del servidor NFS con ***nfsstat***

#### Como es conecta un cliente Windows7

Las versiones Enterprise y Ultimate de Windows 7 soportan los servicios NFS. Si bien es una característica que no se instala por defecto y que tendremos que agregar. Esto lo podemos hacer desde **Panel de Control > Programas > Activar o desactivar características de Windows.**

En la lista de características que nos saldrá, nos desplazamos hasta encontrar Servicios NFS .



Iniciamos el servicio NFS tecleando en un símbolo de sistema con elevación de privilegios la orden.

***nfsadmin client start***

Si queremos ver la lista de recursos compartidos en el servidor remoto tendremos que teclear en un símbolo de sistema la orden

***showmount -e servidor***

Tendremos que sustituir "servidor" por la ip o el nombre equipo de la máquina que esté haciendo las veces de servidor nfs.

Finalmente, montamos el directorio en nuestro Windows 7.

***mount -o anon,fileaccess=777,casesensitive \\servidor\home\hat100 \****

En nuestro caso el uso de los parametros es el siguiente:

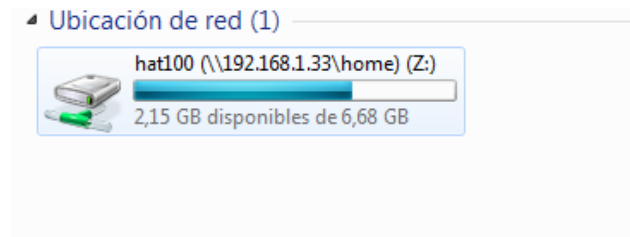
- **anon**= acceso anonimo al recurso compartido
- **fileaccess**= permisos de lectura, escritura y ejecución, la numeración se hereda de los permisos que se otorgan en Linux/unix a las carpetas y archivos.
- **casesensitive**= Especificamos si deseamos distinguir entre mayusculas y minisculas.a
- El **asterisco \*** indica a Windows que monte la carpeta en la siguiente unidad de Red que esté disponible. En nuestro caso están todas las letras libres, así que asigna la letra Z:

Podeís guardar el comando especificado en un archivo **cmd**, y ejecutarlo cada vez que iniciéis sesión utilizando para ello la utilidad **gpedit.msc**

La ruta la tenéis en

***Configuración de usuario > Configuración de Windows > Scripts de inicio de sesión o cierre.***

Sea como sea, al final dispondréis del recurso compartido como una unidad de red más.



## Actividades

Configurar en Debian un servidor NFS una carpeta ***/taller*** en la cual sea accesible desde el cliente Windows. Configúrala para que se monte cada vez que se inicie el equipo. Comprobar que podemos dejar un archivo de texto modificarlo guardarlo apagar el cliente volver a encenderlo y ver que el archivo podremos seguir trabajando con él.