

# Tema 09: Gestión de las comunicaciones.

## Tutorial NFS

Javier Sánchez Monedero

17 de mayo de 2024

### Tabla de contenidos

<b>1 Receta NFS</b>	<b>1</b>
<b>2 Configuración red NAT en VirtualBox</b>	<b>2</b>
<b>3 Configuración NFS</b>	<b>6</b>
<b>4 Referencias</b>	<b>8</b>

### 1 Receta NFS

Escenario:

- VirtualBox como proveedor de red
- Debian como servidor NFS (necesitaríamos NIS/LDAP, pero para este tutorial podemos obviarlo)
- Otro Linux (Debian/Mint/...) como cliente NFS. Puede simplemente clonar la máquina primera generando nuevas MAC o descargar una preinstalada de <https://www.osboxes.org/debian/>.

Precauciones: en la práctica un servicio y red NFS debe diseñarse con cuidado y en conjunto con otras herramientas como firewalls. Aquí vamos a enseñar un ejemplo básico.

## 2 Configuración red NAT en VirtualBox

En VirtualBox añadimos una red NAT en archivos->preferencias. **Nota:** Esto es una red a nivel de VirtualBox que nos permitirá que puedan comunicarse las máquinas entre ellas. Podemos usar la subred 10.0.3.\* para diferenciar de la anterior.

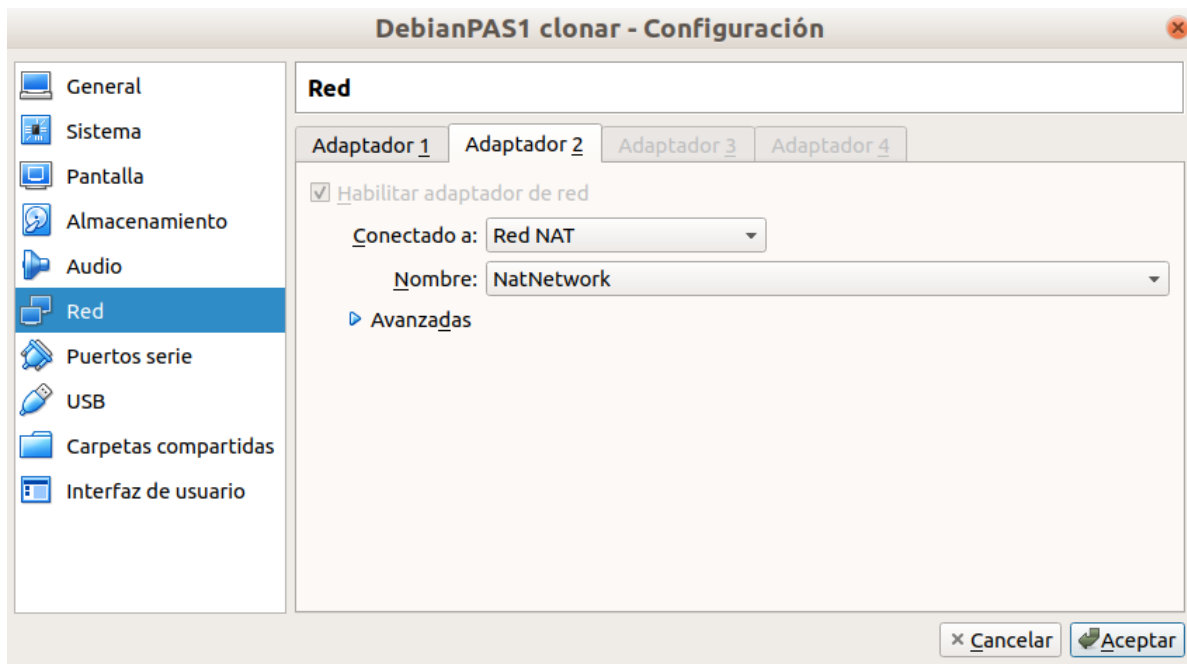


Recuerda añadir reenvío de puertos para poder conectarnos por SSH a nuestro servidor Debian si no lo has hecho:



→

Configuramos tanto la máquina servidor (Debian) como la cliente (Linux Mint) para que use la red NAT que hemos configurado. Para ello añadimos otro interfaz de red.



Iniciamos la máquina Debian. Comprobamos que tenemos el nuevo adaptador de red:

```
ip a
```

Deben aparecer 3: lo, enp0s3 y enp0s8.

Vamos a configurar una IP fija para el servidor (10.0.3.3) y cliente (10.0.3.4) en la interfaz de la red NAT.

```
$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp0s3
iface enp0s3 inet dhcp

# IP secundaria red NAT
```

```
auto enp0s8
iface enp0s8 inet static
    address 10.0.3.3
    netmask 255.255.255.0
```

Puedes hacer el cambio sin que sea permanente con:

```
sudo ip addr add 10.0.3.3/24 dev enp0s8
```

Si queremos conectarnos por SSH tendremos que configurar el reenvío de puertos de la red NAT para esta IP o mantener la interfaz de red anterior (10.0.2.\*)

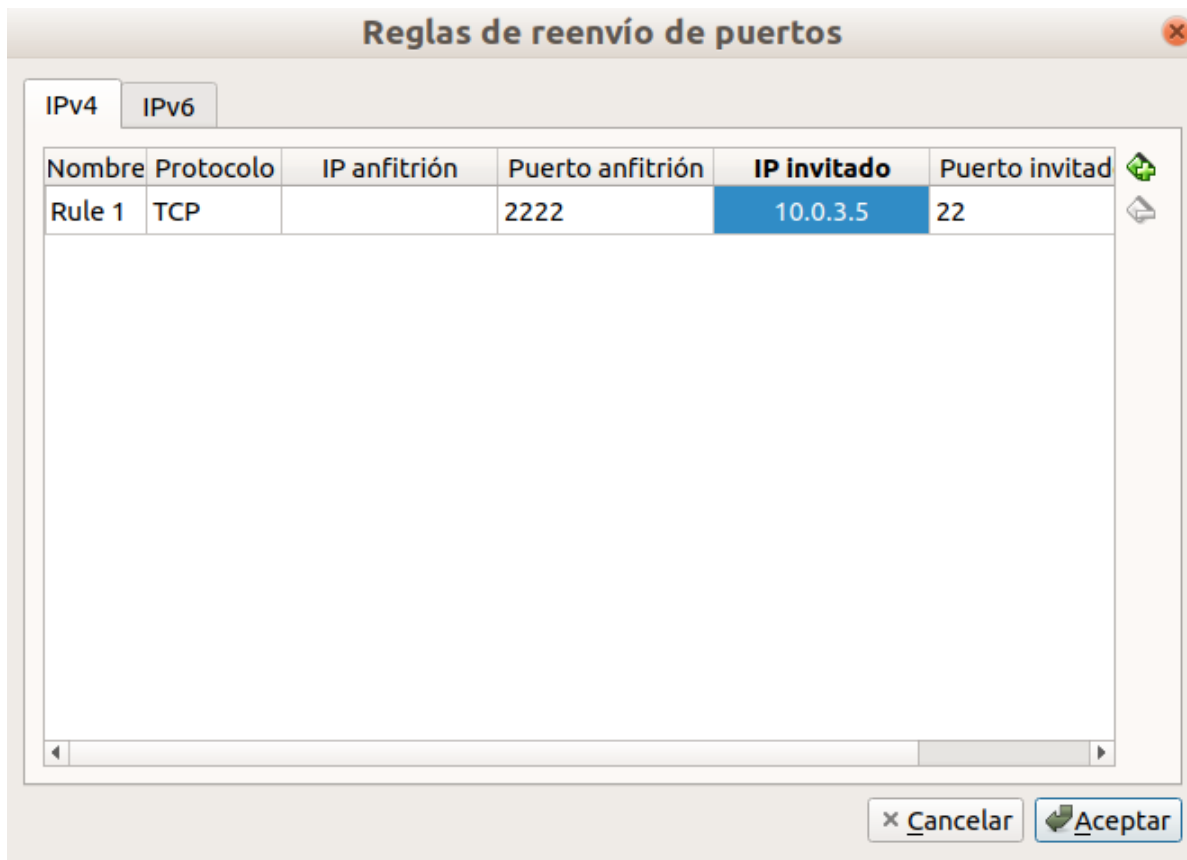
**Importante:** Tanto la conexión NAS como la Red NAS implementan por defecto un servidor DHCP. En el ejemplo anterior la una red sigue con DHCP, **habrá que iniciar primero siempre la máquina del servidor** para que su IP no la coja otra máquina.

```
Debian GNU/Linux 11 debianpas2 tty1

debianpas2 login: pas
Password:
Linux debianpas2 5.10.0-21-amd64 #1 SMP Debian 5.10.162-1 (2023-01-21) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu May 11 18:16:37 CEST 2023 from 10.0.2.2 on pts/0
You have new mail.
pas@debianpas2:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    link/ether 08:00:27:f9:6e:96 brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.5/24 brd 10.0.3.255 scope global dynamic enp0s3
        valid_lft 1020sec preferred_lft 1020sec
    inet6 fe80::a00:27ff:fef9:6e96/64 scope link
        valid_lft forever preferred_lft forever
pas@debianpas2:~$ _
```



->

### 3 Configuración NFS

Comprueba las ips en cada máquina y que pueden “verse” usando ping.

En mi caso: - host\_ip: 10.0.3.3 - client\_ip: 10.0.3.5

#### En el servidor

Instalamos el paquete de soporte NFS:

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

Vamos a exportar el /home. En /etc/exports añadimos:

```
/home client_ip(rw, sync, no_root_squash, no_subtree_check)
```

Echa un vistazo a lo que significa cada una de estas opciones:

- **rw**: Esta opción da al ordenador cliente acceso de lectura y escritura al volumen.
- **sync**: Esta opción obliga a NFS a escribir los cambios en el disco antes de responder. El resultado es un entorno más estable y consistente, ya que la respuesta refleja el estado real del volumen remoto. Sin embargo, también reduce la velocidad de las operaciones de archivo.
- **no\_subtree\_check**: Esta opción evita la comprobación de subárboles, que es un proceso en el que el host debe comprobar si el archivo sigue realmente disponible en el árbol exportado para cada solicitud. Esto puede causar muchos problemas cuando se cambia el nombre de un archivo mientras el cliente lo tiene abierto. En casi todos los casos, es mejor desactivar la comprobación de subárboles.
- **no\_root\_squash**: Por defecto, NFS traduce las peticiones de un usuario root de forma remota a un usuario sin privilegios en el servidor. Esto está pensado como una característica de seguridad para evitar que una cuenta root en el cliente utilice el sistema de archivos del host/servidor como root. **no\_root\_squash** desactiva este comportamiento para ciertos recursos compartidos.

**Pregunta:** ¿Activamos o no **no\_root\_squash**?

```
sudo systemctl restart nfs-kernel-server
```

## En el cliente

Instalamos el paquete de funcionalidades NFS para el cliente:

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

Creación de puntos de montaje y montaje de directorios en el cliente.

**Nota:** Si hay archivos y directorios en su punto de montaje, se ocultarán en cuanto monte el recurso compartido NFS. Para evitar la pérdida de archivos importantes, asegurate de que, si monta en un directorio que ya existe, el directorio esté vacío.

```
# Creamos directorio local
sudo mkdir -p /srv/home
# Montamos /home remoto en el local
sudo mount host_ip:/home /srv/home
# Comprobamos que se ha añadido a nuestro sistema de ficheros
df -h
```

Comprobación de que podemos escribir:

```
sudo touch /srv/home/home.test  
ls -l /srv/home/home.test
```

Para hacer que el directorio NFS se monte al arrancar:

```
# /etc/fstab  
host_ip:/home          /srv/home      nfs auto,nofail,noatime,nolock,intr,tcp,actimeo=1
```

## 4 Referencias

<https://www.digitalocean.com/community/tutorials/how-to-set-up-an-nfs-mount-on-debian-11>

<https://www.howtoforge.com/tutorial/install-nfs-server-and-client-on-debian/>