



SERVIDORES WEB DE ALTAS PRESTACIONES

Práctica 6 - Servidor de disco NFS

César Muñoz Reinoso

Curso 2022-2023

1 de junio de 2023

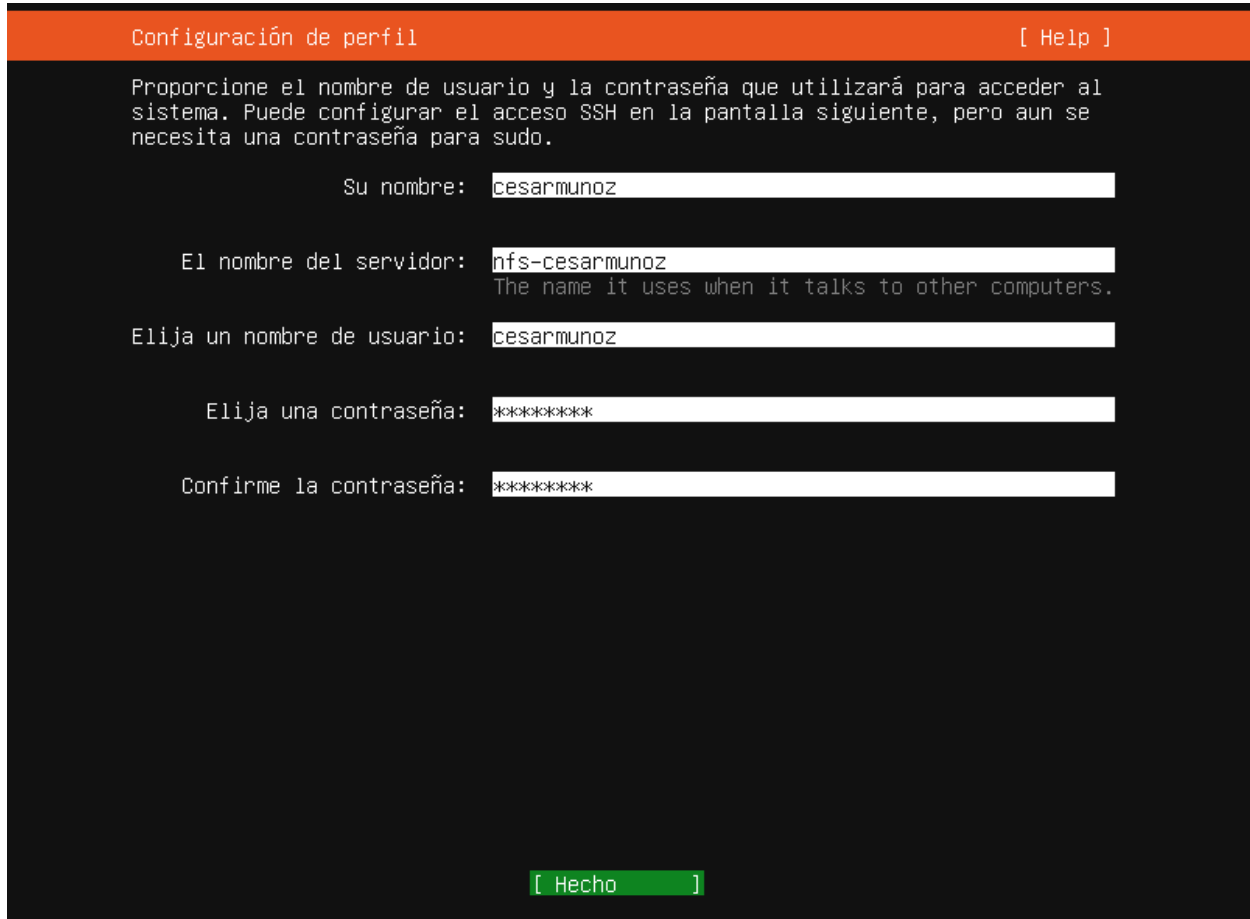
Índice

1. Configurar servidor disco NFS	2
1.1. Creación Máquina Virtual Servidor NFS	2
1.2. Montaje de carpeta exportada en clientes	4
1.3. Comprobación de acceso a archivos	5
1.4. Automontaje de carpeta al arranque del sistema	5
2. Seguridad de NFS	6
2.1. Configuración de mountd y nlockmgr	6
2.2. Configuración reglas IPTABLES	8

1. Configurar servidor disco NFS

1.1. Creación Máquina Virtual Servidor NFS

Al igual que en prácticas anteriores, creamos una máquina virtual nfs-cesarmunoz con 1GB de RAM y 10GB de disco duro. Instalamos Ubuntu Server 22.04.2 con usuario cesarmunoz y contraseña Swap1234. Configuramos además los adaptadores de red NAT y Solo-Anfitrión, en donde asignamos la ip 192.165.56.6.



Configuración de perfil [Help]

Proporcione el nombre de usuario y la contraseña que utilizará para acceder al sistema. Puede configurar el acceso SSH en la pantalla siguiente, pero aun se necesita una contraseña para sudo.

Su nombre: cesarmunoz

El nombre del servidor: nfs-cesarmunoz
The name it uses when it talks to other computers.

Elija un nombre de usuario: cesarmunoz

Elija una contraseña: ****

Confirme la contraseña: ****

[Hecho]

Para configurar el servicio de NFS instalamos las herramientas necesarias, creamos y asignamos permisos a las carpetas que utilizaremos para dicho servicio. Además para darle permiso de lectura y escritura a las máquinas M1 y M2, deberemos añadir las direcciones ip en el archivo de configuración /etc/exports. Reiniciamos y comprobamos el estado del servicio NFS para que esta totalmente configurado y listo para ser utilizado por los clientes.

```
sudo apt-get install nfs-kernel-server nfs-common rpcbind
```

```
sudo mkdir /datos
```

```
sudo mkdir /datos/compartido
```

```
sudo chown nobody:nogroup /datos/compartido
sudo chmod -R 777 /datos/compartido
```

(En /etc/exports)

```
/datos/compartido 192.168.56.2(rw) 192.168.56.3(rw)
```

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

```
cesarmunoz@nfs-cesarmunoz:~$ sudo apt-get install nfs-kernel-server nfs-common rpcbind
[sudo] password for cesarmunoz:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  keyutils libnfsidmap1
Paquetes sugeridos:
  watchdog
Se instalarán los siguientes paquetes NUEVOS:
  keyutils libnfsidmap1 nfs-common nfs-kernel-server rpcbind
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 521 kB de archivos.
Se utilizarán 1.973 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnfsidmap1 amd64 1:2.6.1-1ubuntu1.2 [42,9 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 rpcbind amd64 1.2.6-2build1 [46,6 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy/main amd64 keyutils amd64 1.6.1-2ubuntu3 [50,4 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nfs-common amd64 1:2.6.1-1ubuntu1.2 [241 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 nfs-kernel-server amd64 1:2.6.1-1ubuntu1.2 [140 kB]
Descargados 521 kB en 1s (499 kB/s)
Seleccionando el paquete libnfsidmap1:amd64 previamente no seleccionado.
(Leyendo la base de datos ... 74073 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libnfsidmap1_1%3a2.6.1-1ubuntu1.2_amd64.deb ...
Desempaquetando libnfsidmap1:amd64 (1:2.6.1-1ubuntu1.2) ...
Seleccionando el paquete rpcbind previamente no seleccionado.
Preparando para desempaquetar .../rpcbind_1.2.6-2build1_amd64.deb ...
Desempaquetando rpcbind (1.2.6-2build1) ...
Seleccionando el paquete keyutils previamente no seleccionado.
Preparando para desempaquetar .../keyutils_1.6.1-2ubuntu3_amd64.deb ...
```

```
cesarmunoz@nfs-cesarmunoz:~$ sudo mkdir /datos
cesarmunoz@nfs-cesarmunoz:~$ sudo mkdir /datos/compartido
cesarmunoz@nfs-cesarmunoz:~$ sudo chown nobody:nogroup /datos/compartido/
cesarmunoz@nfs-cesarmunoz:~$ sudo chmod -R 777 /datos/compartido/
```

```
GNU nano 6.2 /etc/exports *
# /etc/exports: the access control list for filesystems which may be exported
#                to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes      hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4       gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/datos/compartido 192.168.56.2(rw) 192.168.56.3(rw)
```

```
cesarmunoz@nfs-cesarmunoz:~$ sudo service nfs-kernel-server status
● nfs-server.service - NFS server and services
   Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled; vendor preset: enabled)
   Active: active (exited) since Tue 2023-05-30 08:42:35 UTC; 5s ago
     Process: 2187 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=1/FAILURE)
     Process: 2188 ExecStart=/usr/sbin/rpc.nfsd (code=exited, status=0/SUCCESS)
    Main PID: 2188 (code=exited, status=0/SUCCESS)
      CPU: 8ms

may 30 08:42:34 nfs-cesarmunoz systemd[1]: Starting NFS server and services...
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: exportfs: /etc/exports [2]: Neither 'subtree_check' or 'no_subtree_check'
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: Assuming default behaviour ('no_subtree_check').
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: NOTE: this default has changed since nfs-utils version 1.0.x
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: exportfs: /etc/exports [2]: Neither 'subtree_check' or 'no_subtree_check'
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: Assuming default behaviour ('no_subtree_check').
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: NOTE: this default has changed since nfs-utils version 1.0.x
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: exportfs: Failed to stat /datos/compartidos: No such file or directory
may 30 08:42:35 nfs-cesarmunoz exportfs[2187]: exportfs: Failed to stat /datos/compartidos: No such file or directory
may 30 08:42:35 nfs-cesarmunoz systemd[1]: Finished NFS server and services.
lines 1-18/18 (END)
```

1.2. Montaje de carpeta exportada en clientes

En las máquinas M1 y M2 instalamos las herramientas necesarias para actuar como clientes de NFS, creamos el punto de montaje en la que queremos montar la carpeta exportada, le otorgamos permisos y la montamos la carpeta remota creada en el servidor NFS.

```
sudo apt-get install nfs-common rpcbind
```

```
cd /home/cesarmunoz
```

```
sudo mkdir datos
```

```
sudo chmod -R 777 datos
```

```
sudo mount 192.168.56.6:/datos/compartido datos
```

```
cesarmunoz@m1-cesarmunoz:~$ sudo apt-get install nfs-common rpcbind
[sudo] password for cesarmunoz:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Se instalarán los siguientes paquetes adicionales:
  keyutils libnfsidmap1
Paquetes sugeridos:
  watchdog
Se instalarán los siguientes paquetes NUEVOS:
  keyutils libnfsidmap1 nfs-common rpcbind
0 actualizados, 4 nuevos se instalarán, 0 para eliminar y 30 no actualizados.
Se necesita descargar 381 kB de archivos.
Se utilizarán 1.447 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libnfsidmap1 amd64 1:2.6.1-1ubuntu1.2 [42,9 kB]
```

```
cesarmunoz@m1-cesarmunoz:~$ cd /home/cesarmunoz/
cesarmunoz@m1-cesarmunoz:~$ mkdir datos
cesarmunoz@m1-cesarmunoz:~$ chmod -R 777 datos
cesarmunoz@m1-cesarmunoz:~$ sudo mount 192.168.56.6:/datos/compartido datos
cesarmunoz@m1-cesarmunoz:~$
```

1.3. Comprobación de acceso a archivos

Comprobamos que efectivamente tenemos acceso a la carpeta remota y que los cambios realizados se actualizan. Para ello desde la máquina M1 creamos un archivo `archivo_prueba.txt` y añadimos una línea para ver los cambios en las diferentes máquinas. Desde M2 comprobamos que tenemos acceso a la carpeta y que se encuentra el archivo creado desde M1, volvemos a añadir una línea al final del archivo y desde la máquina servidor de NFS vemos que se han actualizado los datos.

```
cesarmunoz@m1-cesarmunoz:~$ cd datos/  
cesarmunoz@m1-cesarmunoz:~/datos$ touch archivo_prueba.txt  
cesarmunoz@m1-cesarmunoz:~/datos$ echo 'Escrito desde M1' > archivo_prueba.txt  
cesarmunoz@m1-cesarmunoz:~/datos$ cat archivo_prueba.txt  
Escrito desde M1  
cesarmunoz@m1-cesarmunoz:~/datos$
```

```
cesarmunoz@m2-cesarmunoz:~$ cd datos/  
cesarmunoz@m2-cesarmunoz:~/datos$ cat archivo_prueba.txt  
Escrito desde M1  
cesarmunoz@m2-cesarmunoz:~/datos$ echo 'Escrito desde M2' >> archivo_prueba.txt  
cesarmunoz@m2-cesarmunoz:~/datos$ cat archivo_prueba.txt  
Escrito desde M1  
Escrito desde M2  
cesarmunoz@m2-cesarmunoz:~/datos$
```

```
cesarmunoz@nfs-cesarmunoz:/datos/compartido$ cat archivo_prueba.txt  
Escrito desde M1  
Escrito desde M2  
cesarmunoz@nfs-cesarmunoz:/datos/compartido$
```

1.4. Automontaje de carpeta al arranque del sistema

Para no tener que ejecutar el comando de montaje cada vez que queramos acceder al servidor NFS, podemos realizar una configuración automática de montaje desde `/etc/fstab`. Añadiendo una línea en la que especificamos la carpeta remota, el punto de montaje y las opciones de montaje, este se montará automáticamente al arranque del sistema.

```
192.168.56.6:/datos/compartido /home/cesarmunoz/datos nfs auto,noatime, nolock,bg,  
nfsvers=3,intr,tcp,actime=1800 0 0
```

```

GNU nano 6.2 /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-y81Aaklmhfe7BBNopCjew6vHbiQ20FSKErHoqWELDjLmONv0kHDQS3m6n0q1Z4uD / ext4 defaults 0 1
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/c6fff2ec-cacc-43b9-a6c6-9f78ca77bf75 /boot ext4 defaults 0 1
/swap.img none swap sw 0 0

192.168.56.6:/datos/compartido /home/cesarmunoz/datos nfs auto,noatime,noexec,bg,nfsvers=3,intr,tcp,actimeo=1800 0 0

```

2. Seguridad de NFS

2.1. Configuración de mountd y nlockmgr

Para añadirle seguridad a nuestro sistema NFS abriremos los puertos para que solo los clientes puedan tener acceso a ellos. Los servicios mountd y nlockmgr utilizan puertos dinámicos, luego deberemos fijar dichos puertos para poder usar directivas IPTABLES. Para el servicio mountd, modificaremos el archivo `/etc/default/nfs-kernel-server` y especificaremos que use el puerto 2000. Para nlockmgr tendremos que crear el archivo `swap-nfs-ports.conf` en `/etc/sysctl.d/` y añadir los puertos 2001 y 2002 para los protocolos tcp y udp respectivamente. Lanzamos el archivo de configuración que hemos creado y reiniciamos el servidor NFS. Comprobamos los puertos asociados a cada servicio con `rpcinfo`.

```

(En /etc/default/nfs-kernel-server)
RPCMOUNTDOPTS="--manage gids -p 2000"

```

```

(En /etc/sysctl.d/swap-nfs-ports.conf)
fs.nfs.nlm_tcpport = 2001
fs.nfs.nlm_udpport = 2002

```

```

sudo sysctl --system
/etc/init.d/nfs-kernel-server restart

```

```

sudo rpcinfo -p localhost

```

```

GNU nano 6.2 /etc/default/nfs-kernel-server
# Number of servers to start up
RPCNFSDCOUNT=8

# Runtime priority of server (see nice(1))
RPCNFSDPRIORITY=0

# Options for rpc.mountd.
# If you have a port-based firewall, you might want to set up
# a fixed port here using the --port option. For more information,
# see rpc.mountd(8) or http://wiki.debian.org/SecuringNFS
# To disable NFSv4 on the server, specify '--no-nfs-version 4' here
RPCMOUNTDOPTS="--manage-gids -p 2000"

# Do you want to start the svcgssd daemon? It is only required for Kerberos
# exports. Valid alternatives are "yes" and "no"; the default is "no".
NEED_SVCSSD=""

# Options for rpc.svcgssd.
RPCSVCGSSDOPTS=""

```

```

GNU nano 6.2 /etc/sysctl.d/swap-nfs-ports.conf
fs.nfs.nlm_tcpport = 2001
fs.nfs.nlm_udpport = 2002

```

```

cesarmunoz@nfs-cesarmunoz:/datos/compartido$ sudo nano /etc/default/nfs-kernel-server
[sudo] password for cesarmunoz:
cesarmunoz@nfs-cesarmunoz:/datos/compartido$ sudo touch /etc/sysctl.d/swap-nfs-ports.conf
cesarmunoz@nfs-cesarmunoz:/datos/compartido$ sudo nano /etc/sysctl.d/swap-nfs-ports.conf
cesarmunoz@nfs-cesarmunoz:/datos/compartido$ sudo sysctl --system
* Applying /etc/sysctl.d/10-console-messages.conf ...
kernel.printk = 4 4 1 7
* Applying /etc/sysctl.d/10-ipv6-privacy.conf ...
net.ipv6.conf.all.use_tempaddr = 2
net.ipv6.conf.default.use_tempaddr = 2
* Applying /etc/sysctl.d/10-kernel-hardening.conf ...
kernel.kptr_restrict = 1
* Applying /etc/sysctl.d/10-magic-sysrq.conf ...

```

```

cesarmunoz@nfs-cesarmunoz:/datos/compartido$ /etc/init.d/nfs-kernel-server restart
Restarting nfs-kernel-server (via systemctl): nfs-kernel-server.service==== AUTHENTICATING FOR org.freedesktop.systemd1.
manage-units ====
Authentication is required to restart 'nfs-server.service'.
Authenticating as: cesarmunoz
Password:
==== AUTHENTICATION COMPLETE ====
.

```



```
cesarmunoz@nfs-cesarmunoz:/datos/compartido$ sudo rpcinfo -p localhost
```

program	vers	proto	port	service
100000	4	tcp	111	portmapper
100000	3	tcp	111	portmapper
100000	2	tcp	111	portmapper
100000	4	udp	111	portmapper
100000	3	udp	111	portmapper
100000	2	udp	111	portmapper
100024	1	udp	39015	status
100024	1	tcp	46465	status
100005	1	udp	35939	mountd
100005	1	tcp	39675	mountd
100005	2	udp	53500	mountd
100005	2	tcp	37113	mountd
100005	3	udp	54513	mountd
100005	3	tcp	50269	mountd
100003	3	tcp	2049	nfs
100003	4	tcp	2049	nfs
100227	3	tcp	2049	
100021	1	udp	2002	nlockmgr
100021	3	udp	2002	nlockmgr
100021	4	udp	2002	nlockmgr
100021	1	tcp	2001	nlockmgr
100021	3	tcp	2001	nlockmgr
100021	4	tcp	2001	nlockmgr

2.2. Configuración reglas IPTABLES

Al igual que en practicas anteriores, creamos un script con todos los comandos de iptables necesarios, además como en cada reinicio de las máquinas la configuración de iptables se restablece, crearemos un daemon que se ejecutará en cada arranque del sistema.

Abriremos con el protocolo TCP los puertos 111 para el servicio portmapper, 2000 para mountd, 2001 nlockmgr y 2049 para nfs, además con protocolo UDP los puertos los puertos 111 para el servicio portmapper, 2000 para mountd, 2002 nlockmgr y 2049 para nfs.

```
iptables -A INPUT -s 192.168.56.2,192.168.56.3 - p tcp -m multiport --ports 111,2000,2001,2049 -j ACCEPT
```

```
iptables -A INPUT -s 192.168.56.2,192.168.56.3 - p udp -m multiport --ports 111,2000,2002,2049 -j ACCEPT
```

```

GNU nano 6.2                                iptables_script.sh

#Eliminar al mismo tiempo todas sus reglas

iptables -F
iptables -X
iptables -Z
iptables -t nat -F

#Denegar todo el trafico

iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

#Abrir puertos para acceder a NFS

iptables -A INPUT -s 192.168.56.2,192.168.56.3 -p tcp -m multiport --ports 111,2000,2001,2049 -j ACCEPT
iptables -A INPUT -s 192.168.56.2,192.168.56.3 -p udp -m multiport --ports 111,2000,2002,2049 -j ACCEPT

```

```

GNU nano 6.2                                /etc/systemd/system/iptables_daemon.service

[Unit]
After=syslog.target

[Service]
ExecStart=/home/cesarmunoz/iptables_script.ssh

[Install]
RequiredBy=multi-user.target

```

```

cesarmunoz@nfs-cesarmunoz:~$ sudo nano /etc/systemd/system/iptables_daemon.service
cesarmunoz@nfs-cesarmunoz:~$ systemctl daemon-reload
==== AUTHENTICATING FOR org.freedesktop.systemd1.reload-daemon ====
Authentication is required to reload the systemd state.
Authenticating as: cesarmunoz
Password:
==== AUTHENTICATION COMPLETE ====
cesarmunoz@nfs-cesarmunoz:~$ systemctl enable iptables_daemon.service
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-unit-files ====
Authentication is required to manage system service or unit files.
Authenticating as: cesarmunoz
Password:
==== AUTHENTICATION COMPLETE ====
Created symlink /etc/systemd/system/multi-user.target.requires/iptables_daemon.service → /etc/systemd/system/iptables_daemon.service.
==== AUTHENTICATING FOR org.freedesktop.systemd1.reload-daemon ====
Authentication is required to reload the systemd state.
Authenticating as: cesarmunoz
Password:
==== AUTHENTICATION COMPLETE ====

```