



UNIVERSIDAD
DE GRANADA

PRÁCTICA 4: ASEGURAR LA GRANJA WEB

ANA BUENDÍA RUIZ-AZUAGA

Correo electrónico

anabuenrúa@correo.ugr.es

E.T.S. INGENIERÍAS INFORMÁTICA Y DE TELECOMUNICACIÓN

Granada, a 25 de mayo de 2022

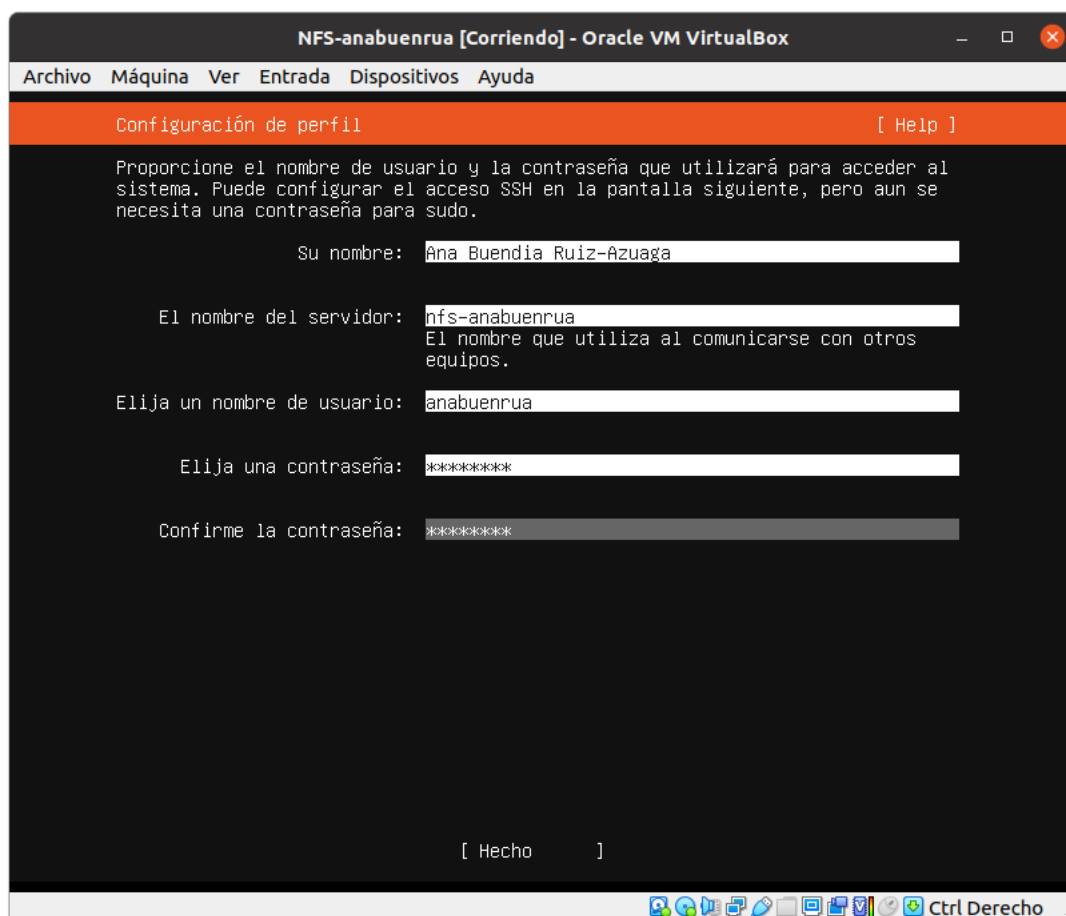
ÍNDICE GENERAL

| | | |
|------|-----------------------------------|----|
| 1. | INSTALACIÓN DE LA MÁQUINA VIRTUAL | 3 |
| 2. | CONFIGURAR SERVIDOR DE DISCO NFS | 5 |
| 2.1. | Opciones avanzadas | 7 |
| 3. | SEGURIDAD EN NFS | 10 |
| 3.1. | Opciones avanzadas | 11 |
| 4. | BIBLIOGRAFÍA | 14 |

INSTALACIÓN DE LA MÁQUINA VIRTUAL

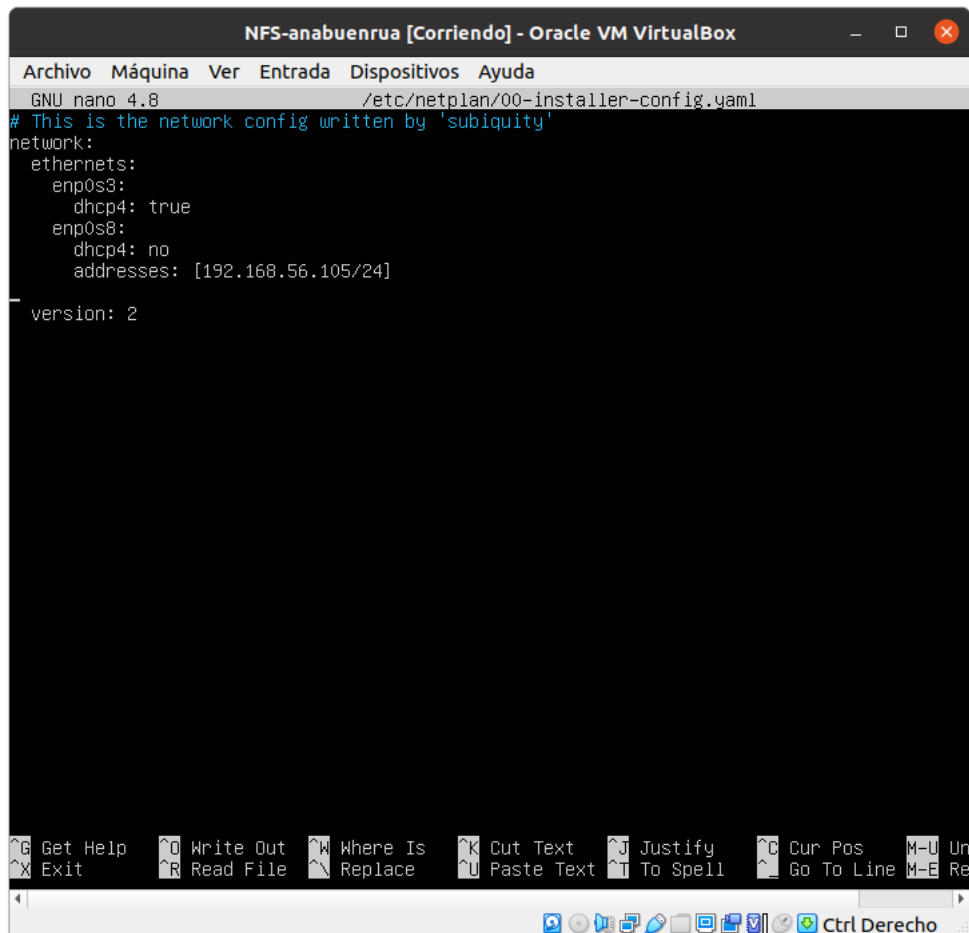
Para realizar la práctica creamos la máquina virtual NFS-anabuenrúa, que tendrá, al igual que las otras, 1GB de RAM y 10GB de disco duro dinámico. Además, se ha añadido antes de lanzar la máquina virtual el adaptador de red solo-anfitrión. Introducimos los datos como puede verse en (1).

Figura 1: Introducimos los datos de la máquina virtual durante la instalación.



Ahora asignamos una IP estática a `nfs-anabuenrúa`, se ha escogido `192.168.56.105` editando `/etc/netplan/00-installer-config.yaml`, por tanto, el fichero quedaría como en (2).

Figura 2: Fichero `/etc/netplan/00-installer-config.yaml`

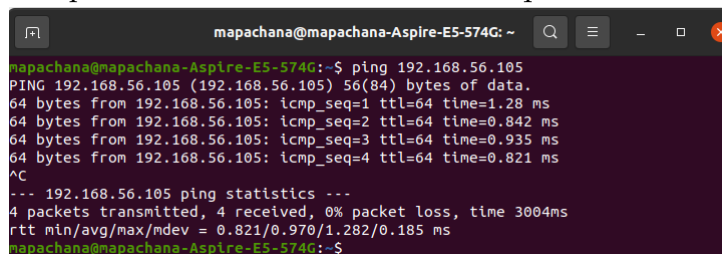


```

GNU nano 4.8 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      addresses: [192.168.56.105/24]
  version: 2
  
```

Hacemos los cambios efectivos ejecutando `sudo netplan apply` como en las prácticas anteriores y comprobamos que hay conexión en (3).

Figura 3: Comprobación de la conexión de la máquina con otras máquinas.



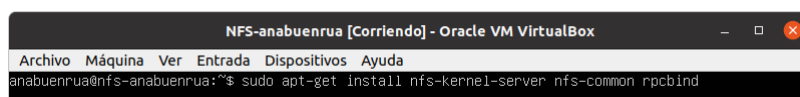
```

mapachana@mapachana-Aspire-E5-574G: ~
mapachana@mapachana-Aspire-E5-574G:~$ ping 192.168.56.105
PING 192.168.56.105 (192.168.56.105) 56(84) bytes of data:
64 bytes from 192.168.56.105: icmp_seq=1 ttl=64 time=1.28 ms
64 bytes from 192.168.56.105: icmp_seq=2 ttl=64 time=0.842 ms
64 bytes from 192.168.56.105: icmp_seq=3 ttl=64 time=0.935 ms
64 bytes from 192.168.56.105: icmp_seq=4 ttl=64 time=0.821 ms
^C
--- 192.168.56.105 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.821/0.970/1.282/0.185 ms
mapachana@mapachana-Aspire-E5-574G:~$
  
```

CONFIGURAR SERVIDOR DE DISCO NFS

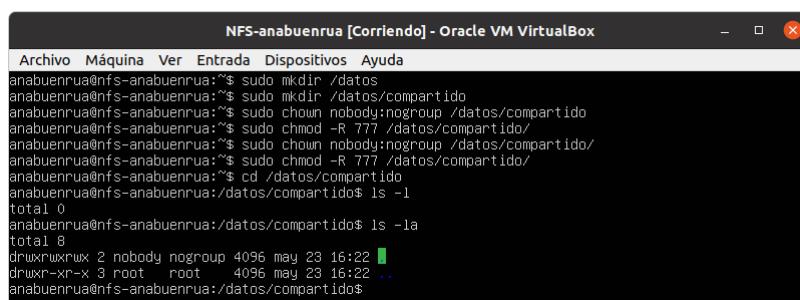
Comenzamos la práctica trabajando en `nfs-anabuenrúa`. Primero instalamos las herramientas básicas que vamos a necesitar ejecutando el comando de (4).

Figura 4: Instalación de herramientas en `nfs-anabuenrúa`



Ahora creamos la carpeta `/datos/compartido` donde vamos a tener los ficheros que se van a compartir entre las máquinas virtuales, cambiamos su propietario y grupo y asignamos permisos, como mostramos en (5).

Figura 5: Creación de carpeta `/datos/compartido`, cambio de propietarios y asignación de permisos.



A continuación editamos `/etc/exports` para dar permisos de accesos a `m1` y `m2`, como se ve en (6).

Finalmente relanzamos el servicio y comprobamos su estado. Esto puede consultarse en (7).

Ahora vamos a configurar `m1` y `m2`. Para no ser repetitivo, se va a realizar y mostrar la configuración solamente en `m1`, ya que en `m2` se haría la misma.

Comenzamos instalando las herramientas que vamos a usar, como se ve en (8)

Figura 6: Fichero /etc/exports.

```

GNU nano 4.8 /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/ 192.168.56.101(rw) 192.168.56.102(rw)

```

Figura 7: Relanzar servicio y comprobar su estado.

```

NFS-anabuenrúa:/datos/compartido$ sudo service nfs-kernel-server restart
anabuenrúa@nfs-anabuenrúa:/datos/compartido$ 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 Mon 2022-05-23 16:33:03 UTC; 3s ago
     Process: 2696 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=0/SUCCESS)
     Process: 2697 ExecStart=/usr/sbin/rpc.nfsd $RPCNFSDARGS (code=exited, status=0/SUCCESS)
    Main PID: 2697 (code=exited, status=0/SUCCESS)

may 23 16:33:02 nfs-anabuenrúa systemd[1]: Starting NFS server and services...
may 23 16:33:02 nfs-anabuenrúa exportfs[2696]: exportfs: /etc/exports [2]: Neither 'subtree_check'
may 23 16:33:02 nfs-anabuenrúa exportfs[2696]: Assuming default behaviour ('no_subtree_check').
may 23 16:33:02 nfs-anabuenrúa exportfs[2696]: NOTE: this default has changed since nfs-utils ver
may 23 16:33:02 nfs-anabuenrúa exportfs[2696]: exportfs: /etc/exports [2]: Neither 'subtree_check'
may 23 16:33:02 nfs-anabuenrúa exportfs[2696]: Assuming default behaviour ('no_subtree_check').
may 23 16:33:02 nfs-anabuenrúa exportfs[2696]: NOTE: this default has changed since nfs-utils ver
may 23 16:33:03 nfs-anabuenrúa systemd[1]: Finished NFS server and services.
lines 1-15/15 (END)

```

Figura 8: Instalación de herramientas en m1

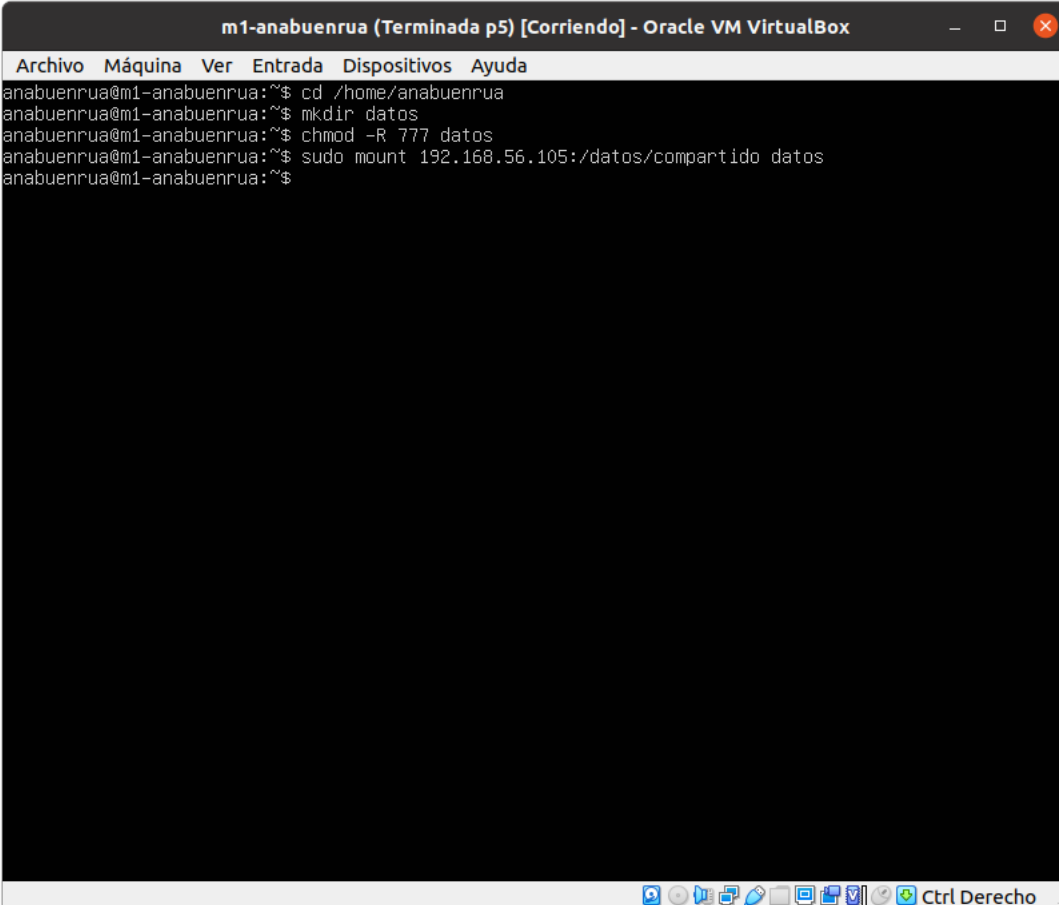
```

m1-anabuenrúa (Terminada p5) [Corriendo] - Oracle VM VirtualBox
anabuenrúa@m1-anabuenrúa:~$ sudo apt-get install nfs-common rpcbind

```

A continuación creamos el punto de montaje datos en /home/anabuenrúa, le asignamos los permisos y montamos la carpeta remota, como se muestra en (9).

Figura 9: Creación del punto de montaje, asignación de permisos y montaje del directorio remoto.



```
m1-anabuenrúa (Terminada p5) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@m1-anabuenrúa:~$ cd /home/anabuenrúa
anabuenrúa@m1-anabuenrúa:~$ mkdir datos
anabuenrúa@m1-anabuenrúa:~$ chmod -R 777 datos
anabuenrúa@m1-anabuenrúa:~$ sudo mount 192.168.56.105:/datos/compartido datos
anabuenrúa@m1-anabuenrúa:~$
```

Finalmente comprobamos que funciona, pues al crear un archivo en la carpeta datos de m1 se muestra en m2 y nfs. Esto puede verse en (10).

2.1 OPCIONES AVANZADAS

Para comenzar, vamos a hacer que el montaje de la carpeta remota se realice de forma automática al arrancar la máquina virtual tanto en m1 como m2. Para ello, vamos a editar el fichero /etc/fstab añadiendo la línea siguiente:

```
192.168.56.105:/datos/compartido /home/anabuenrúa/datos/ nfsauto,noatime,
nolock,bg,nfsvers=3,intr,tcp,actimeo=1800 0 0
```

Pese a que hemos tenido que mostrarla en 2 líneas, esta es una sola línea. El archivo editado puede verse en (11).

Figura 10: Comprobación del correcto funcionamiento.

The figure consists of three terminal windows from Oracle VM VirtualBox, each showing the output of the 'ls -la' command on a specific NFS mount.

Window 1: m1-anabuenrue (Terminada p5) [Corriendo] - Oracle VM VirtualBox

```

anabuenrue@m1-anabuenrue:~$ ls -la datos/
total 8
drwxrwxrwx 2 nobody nogroup 4096 may 23 16:22 .
drwxr-xr-x 9 anabuenrue anabuenrue 4096 may 23 16:45 ..
anabuenrue@m1-anabuenrue:~$ touch datos/archivo.txt
anabuenrue@m1-anabuenrue:~$ ls -la datos/
total 8
drwxrwxrwx 2 nobody nogroup 4096 may 23 17:08 .
drwxr-xr-x 9 anabuenrue anabuenrue 4096 may 23 16:45 ..
-rw-rw-r-- 1 anabuenrue anabuenrue 0 may 23 17:08 archivo.txt
anabuenrue@m1-anabuenrue:~$ _

```

Window 2: m2-anabuenrue (terminada p5) [Corriendo] - Oracle VM VirtualBox

```

anabuenrue@m2-anabuenrue:~$ ls -la datos/
total 8
drwxrwxrwx 2 nobody nogroup 4096 may 23 16:22 .
drwxr-xr-x 9 anabuenrue anabuenrue 4096 may 23 17:03 ..
anabuenrue@m2-anabuenrue:~$ ls -la datos/
total 8
drwxrwxrwx 2 nobody nogroup 4096 may 23 17:08 .
drwxr-xr-x 9 anabuenrue anabuenrue 4096 may 23 17:03 ..
-rw-rw-r-- 1 anabuenrue anabuenrue 0 may 23 17:08 archivo.txt
anabuenrue@m2-anabuenrue:~$ _

```

Window 3: NFS-anabuenrue [Corriendo] - Oracle VM VirtualBox

```

anabuenrue@nfs-anabuenrue:/datos/compartido$ ls -la
total 8
drwxrwxrwx 2 nobody nogroup 4096 may 23 16:22 .
drwxr-xr-x 3 root root 4096 may 23 16:22 ..
anabuenrue@nfs-anabuenrue:/datos/compartido$ ls -la
total 8
drwxrwxrwx 2 nobody nogroup 4096 may 23 17:08 .
drwxr-xr-x 3 root root 4096 may 23 16:22 ..
-rw-rw-r-- 1 anabuenrue anabuenrue 0 may 23 17:08 archivo.txt
anabuenrue@nfs-anabuenrue:/datos/compartido$ _

```

Figura 11: Fichero /etc/fstab

The figure shows a terminal window titled 'm1-anabuenrue (Terminada p5) [Corriendo] - Oracle VM VirtualBox' with the following commands and output:

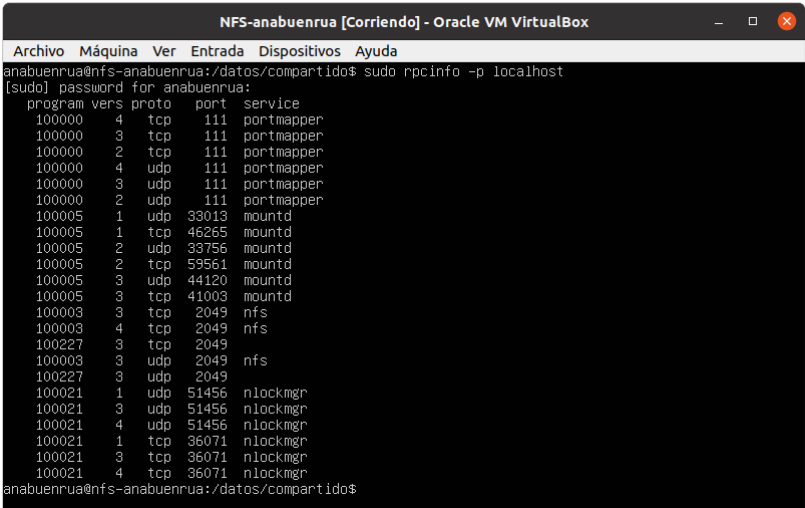
```

anabuenrue@m1-anabuenrue:~$ cd /home/anabuenrue
anabuenrue@m1-anabuenrue:~$ mkdir datos
anabuenrue@m1-anabuenrue:~$ chmod -R 777 datos
anabuenrue@m1-anabuenrue:~$ sudo mount 192.168.56.105:/datos/compartido datos
anabuenrue@m1-anabuenrue:~$

```

Además, en la máquina `nfs-anabuenrue` podemos comprobar qué puertos se están usando como se ve en (12).

Figura 12: Puertos asignados.

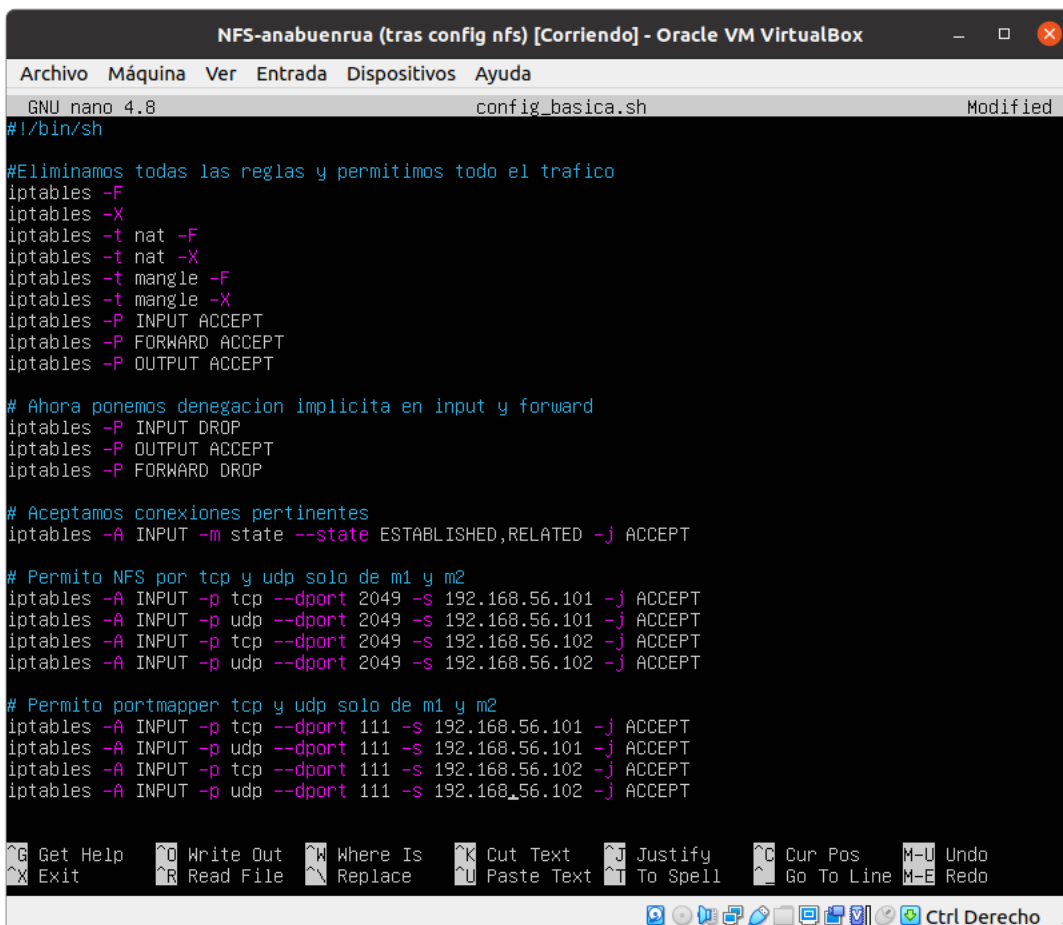


SEGURIDAD EN NFS

Comenzamos creando en nfs-anabuenrúa en /home/anabuenrúa una carpeta scripts_iptable para almacenar los ficheros de configuración de iptables como en las otras máquinas.

Vamos a crear un fichero en esta carpeta que contenga las reglas de seguridad por defecto de esta máquina. Este fichero puede consultarse en (13).

Figura 13: Fichero de configuración básica de iptables.



```
NFS-anabuenrúa (tras config nfs) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 4.8                                config_basica.sh  Modified
#!/bin/sh

#Eliminamos todas las reglas y permitimos todo el trafico
iptables -F
iptables -X
iptables -t nat -F
iptables -t nat -X
iptables -t mangle -F
iptables -t mangle -X
iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT

# Ahora ponemos denegacion implicita en input y forward
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP

# Aceptamos conexiones pertinentes
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Permito NFS por tcp y udp solo de m1 y m2
iptables -A INPUT -p tcp --dport 2049 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p udp --dport 2049 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p tcp --dport 2049 -s 192.168.56.102 -j ACCEPT
iptables -A INPUT -p udp --dport 2049 -s 192.168.56.102 -j ACCEPT

# Permito portmapper tcp y udp solo de m1 y m2
iptables -A INPUT -p tcp --dport 111 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p udp --dport 111 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p tcp --dport 111 -s 192.168.56.102 -j ACCEPT
iptables -A INPUT -p udp --dport 111 -s 192.168.56.102 -j ACCEPT

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos   M-U Undo
^X Exit      ^R Read File  ^N Replace   ^U Paste Text ^T To Spell   ^_ Go To Line M-E Redo
Ctrl Derecho
```

3.1 OPCIONES AVANZADAS

Como mountd y nlockmgr usan puertos dinámicos para poder escribir reglas de iptables primero vamos a fijar los puertos de estos servicios.

Comenzamos fijando el puerto de mountd, para lo que editamos el fichero

/etc/default/nfs-kernel-server editando la línea correspondiente para fijar el puerto a 2000 (por ejemplo, podríamos asignar otro). Este fichero puede verse en (14).

Figura 14: Fichero /etc/default/nfs-kernel-server.

```

GNU nano 4.8 /etc/default/nfs-kernel-server Modified
# 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_SVCGSSD=""

# Options for rpc.svcgssd.
RPCSVCGSSDOPTS=""

```

Para fijar el puerto de nlockmgr creamos el archivo swap-nfs-ports.conf en /etc/sysctl.d/ con el contenido que se muestra en (15) para fijar los puertos de tcp y udp a 2001 y 2002 respectivamente.

Ahora reiniciamos el sistema especificando este archivo de configuración. En (16) se muestran los comandos empleados. Se omite la salida del primer comando debido a que es muy larga.

Y comprobamos finalmente los puertos una vez realizada esta configuración. Esto puede verse en (17)

Figura 15: Fichero /etc/sysctl.d/swap-nfs-ports.conf.

```

NFS-anabuenrúa (tras config nfs) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 4.8 swap-nfs-ports.conf Modified
# Configuración de puertos de nlockmgr

fs.nfs.nlm_tcpport = 2001
fs.nfs.nlm_udpport = 2002

```

Figura 16: Reinicio del sistema especificando el archivo de configuración.

```

NFS-anabuenrúa (tras config nfs) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@nfs-anabuenrúa:/$ sudo systemctl --system_

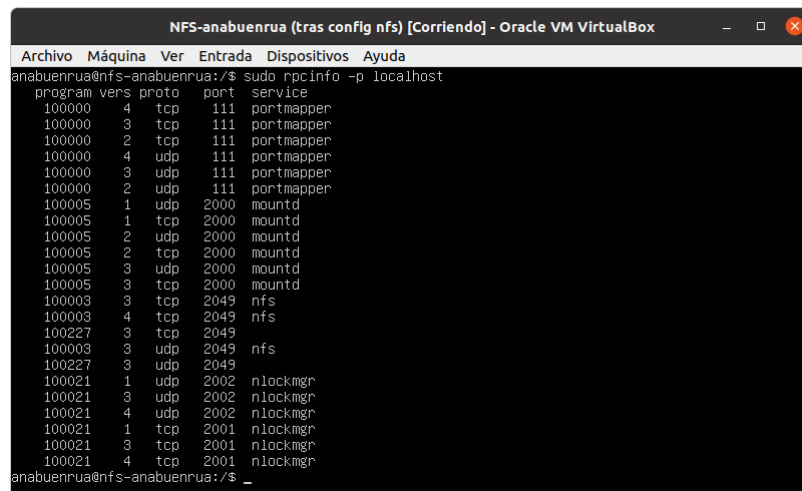
NFS-anabuenrúa (tras config nfs) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrúa@nfs-anabuenrúa:/$ /etc/init.d/nfs-kernel-server restart
Restarting nfs-kernel-server (via systemctl): nfs-kernel-server.service==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ===
Authentication is required to restart 'nfs-server.service'.
Authenticating as: Ana Buendia Ruiz-Azuaga (anabuenrúa)
Password:
==== AUTHENTICATION COMPLETE ===
anabuenrúa@nfs-anabuenrúa:/$

```

Ahora que hemos fijado los puertos de estos servicios, podemos definir reglas para abrir los puertos correspondientes a estos. Partiendo del fichero de configuración ya presentado, lo editamos como se ve en (18).

Finalmente comprobamos que todo funciona correctamente repitiendo la prueba que puede verse en (10).

Figura 17: Comprobación de puertos asignados.

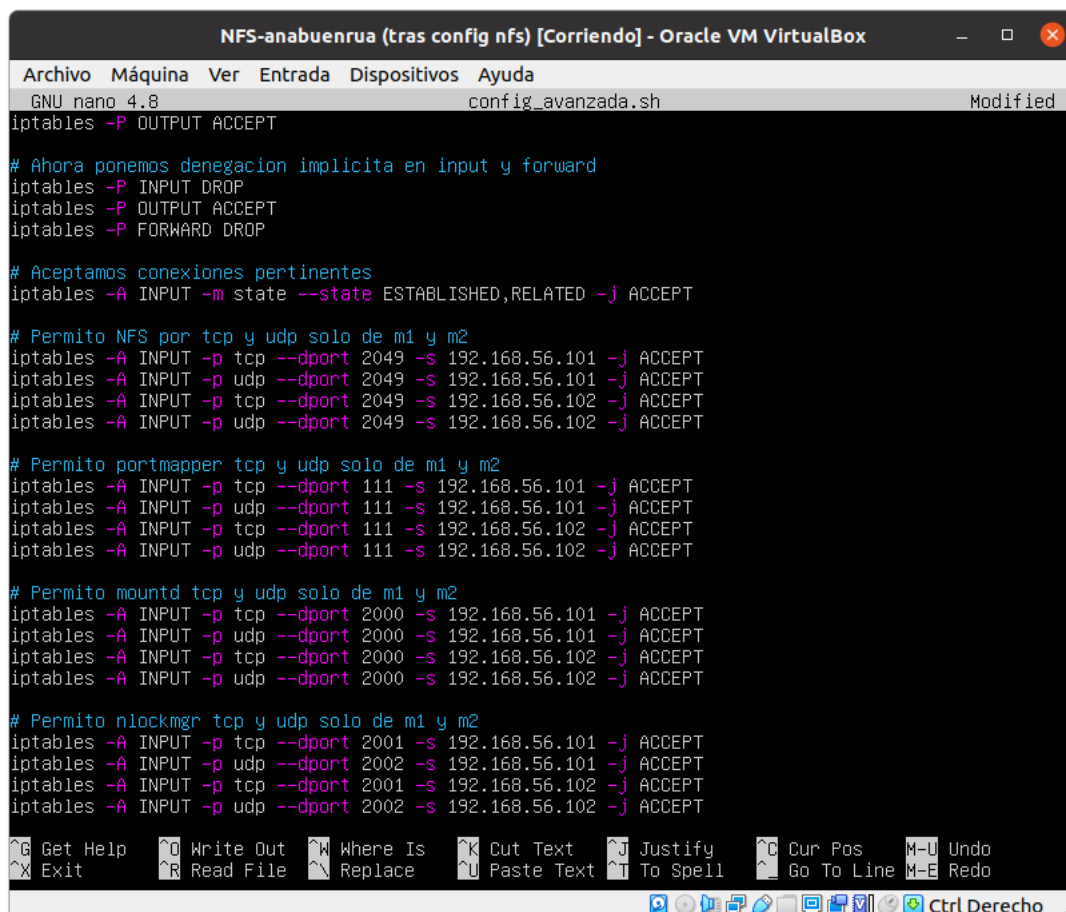


```

NFS-anabuenrue (tras config nfs) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
anabuenrue@nfs-anabuenrue:/$ sudo nmap -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
100005 1 udp 2000 mountd
100005 1 tcp 2000 mountd
100005 2 udp 2000 mountd
100005 2 tcp 2000 mountd
100005 3 udp 2000 mountd
100005 3 tcp 2000 mountd
100003 3 tcp 2049 nfs
100003 4 tcp 2049 nfs
100227 3 tcp 2049
100003 3 udp 2049 nfs
100227 3 udp 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
anabuenrue@nfs-anabuenrue:/$

```

Figura 18: Fichero de configuración avanzada de iptables



```

NFS-anabuenrue (tras config nfs) [Corriendo] - Oracle VM VirtualBox
GNU nano 4.8 config_avanzada.sh Modified
iptables -P OUTPUT ACCEPT

# Ahora ponemos denegacion implicita en input y forward
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP

# Aceptamos conexiones pertinentes
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Permito nfs por tcp y udp solo de m1 y m2
iptables -A INPUT -p tcp --dport 2049 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p udp --dport 2049 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p tcp --dport 2049 -s 192.168.56.102 -j ACCEPT
iptables -A INPUT -p udp --dport 2049 -s 192.168.56.102 -j ACCEPT

# Permito portmapper tcp y udp solo de m1 y m2
iptables -A INPUT -p tcp --dport 111 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p udp --dport 111 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p tcp --dport 111 -s 192.168.56.102 -j ACCEPT
iptables -A INPUT -p udp --dport 111 -s 192.168.56.102 -j ACCEPT

# Permito mountd tcp y udp solo de m1 y m2
iptables -A INPUT -p tcp --dport 2000 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p udp --dport 2000 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p tcp --dport 2000 -s 192.168.56.102 -j ACCEPT
iptables -A INPUT -p udp --dport 2000 -s 192.168.56.102 -j ACCEPT

# Permito nlockmgr tcp y udp solo de m1 y m2
iptables -A INPUT -p tcp --dport 2001 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p udp --dport 2002 -s 192.168.56.101 -j ACCEPT
iptables -A INPUT -p tcp --dport 2001 -s 192.168.56.102 -j ACCEPT
iptables -A INPUT -p udp --dport 2002 -s 192.168.56.102 -j ACCEPT

```

Tras comprobar que la granja funciona correctamente y está bien protegida, usamos iptables-persistent como hicimos en la práctica 4 (y se usó de nuevo en la 5) para hacer las reglas persistentes al inicio.

BIBLIOGRAFÍA

- Diapositivas y gui3n de la pr3ctica.
- <https://www.mysqltutorial.org/mysql-not-null-constraint/>
- https://www.w3schools.com/sql/sql_unique.asp
- https://www.w3schools.com/sql/sql_primarykey.ASP
- <https://www.mysqltutorial.org/mysql-unique-constraint/>
- <http://pwet.fr/man/linux/commandes/mysqldump/>
- <https://stdworkflow.com/927/2061-authentication-plugin-caching-sha2-password-reported-error-authentication-require-secure-connection>
- <https://dev.mysql.com/doc/refman/5.7/en/replication-administration-status.html>