



## UD 10. LINUX: REDES

Sistemas informáticos  
CFGS DAW

Borja Salom

[b.salomsantamaria@edu.gva.es](mailto:b.salomsantamaria@edu.gva.es)

2022/2023

Versión:230217.1050

## Licencia



**Atribución - No comercial - Compartir igual**  
**(por-nc-sa):** No se permite el uso comercial de la obra original ni de ninguna obra derivada. cuya distribución debe realizarse bajo una licencia igual a la que rige la obra original.

## Nomenclatura

A lo largo de esta unidad se utilizarán diferentes símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

☐ Importante

☐ Atención

☐ Interesante

# ÍNDICE DE CONTENIDO

<b>1. Comandos de red de Linux.....</b>	<b>4</b>
1.1 Comando Ifconfig.....	4
1.2 Comando ip.....	5
1.3 Comando ping.....	8
1.4 Antiguos comandos frente a nuevos comandos.....	9
<b>2. SSH.....</b>	<b>9</b>
2.1 Intall Servidor SSH .....	10
2.2 Puertos SSH.....	10
2.3 Comandos SSH .....	10
<b>3. SFTP.....</b>	<b>11</b>
3.1 Conectarse a SFTP.....	12
3.2 Transferir archivos con SFTP .....	12
3.3 Manipulaciones sencillas de archivos con SFTP .....	13

## UD10. LINUX: REDES

### 1. SAMBA

Samba es un protocolo de comunicación para conectar windows con el sistema de archivos linux y así ver los discos o carpetas linux como unidades de red en windows.

Instalar y configurar samba es bastante sencillo, hay que instalarlo desde la línea de comandos:

```
apt-get install samba
```

Ahora tienes que configurarlo para acceder a las carpetas de linux desde windows. Para configurarlo tienes que editar el archivo

```
:[profiles]
; comentario = Perfiles de usuarios
; ruta = /home/samba/profiles
invitado ok = no
browseable = no
; crear máscara = 0600
; máscara de directorio = 0700
```

Lo copiamos y modificamos tal y como queremos que se muestre nuestro directorio en windows:

```
[test_samba]
comment = directorio de prueba
samba path =
/home/user/samba_test guest ok =
no
navegable = sí crear
máscara = 0600
directorio máscara =
```

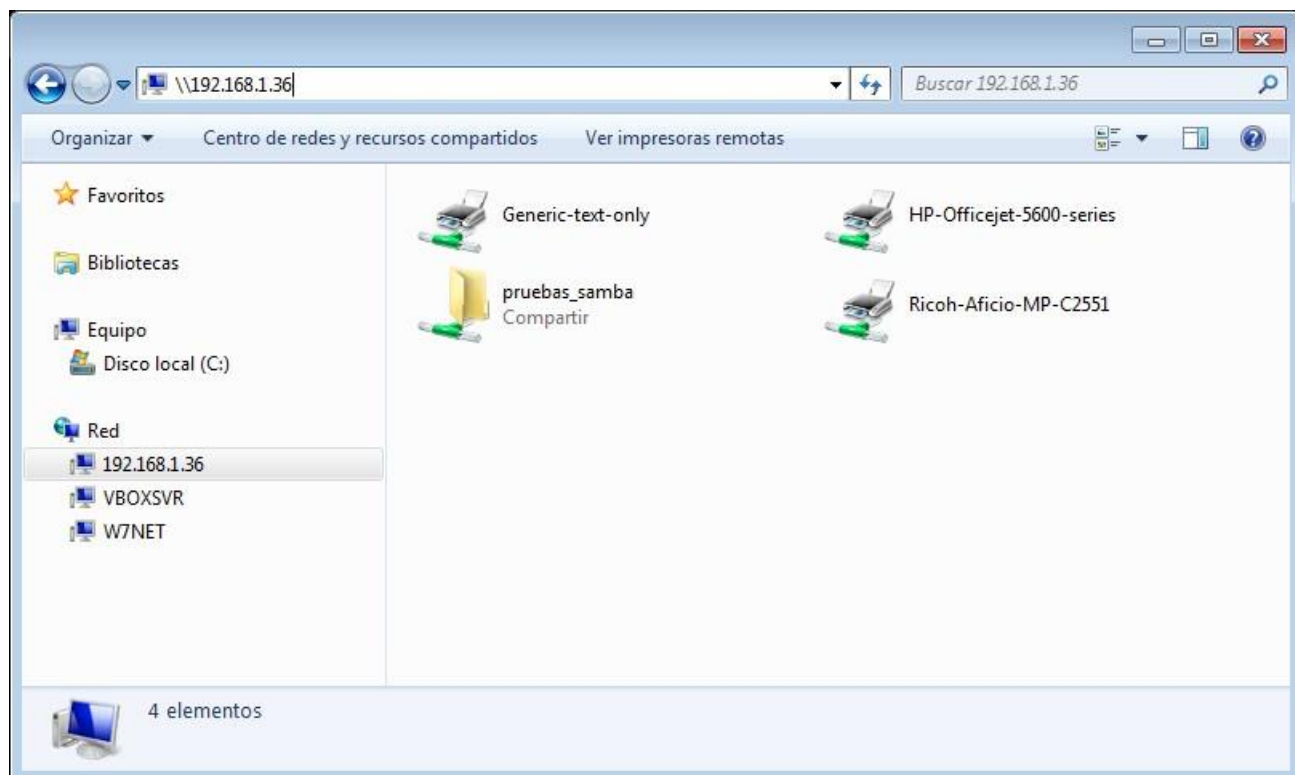
Luego ponemos una contraseña samba al usuario que queremos que tenga acceso al directorio

```
sudo smbpasswd -L -a  
nombredeusuario # ponemos una  
contraseña  
sudo smbpasswd -l -e nombredeusuario
```

Habilitamos el usuario para samba y reiniciamos el servicio

```
sudo /etc/init.d/smbd restart
```

Ahora vamos a windows y escribimos nuestra IP en el explorador de windows.



Ahora probaremos que podemos escribir en el directorio para que funcione.

Si estamos compartiendo directorios web en /var/www o algo así, puede que tengas problemas de acceso pero esto se puede solucionar poniendo las siguientes directivas en el fichero de configuración:

```
forzar usuario = root  
forzar grupo = root
```

## 2. SERVIDOR NFS

Un servidor NFS (Network File System) es un sistema que permite a los clientes acceder a sistemas de archivos remotos a través de la red.

El servidor NFS comparte uno o varios sistemas de archivos con los clientes mediante el protocolo NFS, que es un protocolo estándar de Internet para compartir archivos entre sistemas Unix/Linux.

Los clientes NFS pueden montar los sistemas de archivos remotos desde el servidor como si fueran sistemas de archivos locales, lo que les permite acceder a los archivos y directorios remotos de forma transparente como si estuvieran en su propio sistema.

El servidor NFS se utiliza ampliamente en entornos informáticos en red Unix/Linux para compartir recursos de almacenamiento de archivos y permitir el acceso a datos a través de la red de forma eficiente.

### 2.1 Instalación del servidor NFS en Linux

Lo que vamos a hacer primero es actualizar nuestros servidores:

```
$ sudo apt update && apt dist-upgrade
```

Una vez que esté actualizado y reiniciemos, vamos a instalar NFS.

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

El siguiente paso es crear la carpeta compartida, en mi caso, tendría este aspecto.

```
$ sudo mkdir -p /mnt/data
```

Especifiquemos algunos permisos para que "todos" los clientes puedan escribir en ese directorio.

```
$ sudo chown nobody:nogroup /mnt/data  
$ sudo chmod 777 /mnt/data
```

Lo siguiente que vamos a hacer es configurar NFS para que sólo acepte conexiones de nuestros clientes. Si no se hace esto, cuando queramos conectarnos desde un cliente, NFS nos dará acceso denegado.

Para ello, vamos a editar el archivo de "exportación

```
$ sudo nano /etc/exports
```

Lo siguiente es añadir la carpeta que queremos compartir y especificamos cuales serán las direcciones IP que pueden escribir en esa carpeta, para mi ejemplo quedaría así:

```
/mnt/datos      192.168.1.80(rw,sync,no_subtree_check)
s               192.168.1.81(rw,sync,no_subtree_check)
/mnt/datos      192.168.1.82(rw,sync,no_subtree_check)
```

Y el archivo completo tendría este aspecto:

```
# /etc/exports: la lista de control de acceso para sistemas de archivos que
# pueden ser exportados #a clientes NFS. Ver exports(5).
#
# Ejemplo para NFSv2 y NFSv3:
# /srv/homeshostname1 (rw,sync,no_subtree_check)
# hostname2(ro,sync,no_subtree_check)
#
# Ejemplo para NFSv4:
# /srv/nfs4gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check) #
# /srv/nfs4/homes gss/krb5i(rw,sync,no_subtree_check)
#
/mnt/datos      192.168.1.80(rw,sync,no_subtree_check)
s               192.168.1.81(rw,sync,no_subtree_check)
/mnt/datos      192.168.1.82(rw,sync,no_subtree_check)
```

Básicamente lo que estoy haciendo con esto es permitir que mi clúster docker swarm pueda acceder y escribir en esa carpeta compartida.

Una vez que tengamos esto listo, vamos a "publicar" estas exportaciones:

```
$ sudo exportfs -a
```

Lo siguiente es reiniciar el servidor NFS de esta forma para que tome esos cambios:

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

Como paso opcional, si hay un cortafuegos activado en los servidores, recuerda hacer excepciones para que el tráfico entre clientes y servidor fluya.

## 2.2 Configurar los clientes

Algo que no mencioné es que la dirección IP que asigné a mi servidor NFS es la IP 192.168.1.83, dicho esto, vamos a instalar el cliente NFS en nuestros nodos.

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

Una vez instalado, vamos a crear una carpeta local, para montar la carpeta remota. En este caso, para evitar confusiones, la voy a llamar "cliente1" pero me gusta que tengan el mismo nombre para poder estandarizar configuraciones.

```
$ sudo mkdir -p /mnt/cliente1
```

Una vez tengamos la carpeta, vamos a montar el directorio remoto en esta que acabamos de crear, quedaría así:

```
$ sudo mount 192.168.1.83:/mnt/data /mnt/cliente1
```

A partir de ese momento, cuando naveguen a `"/mnt/cliente1"` verán el contenido de `"/mnt/datos"` desde el servidor NFS.

Finalmente, seguramente viene la pregunta, ¿cómo hago para que este cambio sea persistente y permanezca a través de los reinicios? y yo te respondo, que buena pregunta.

Para que el montaje se haga automáticamente al arrancar, tenemos que tocar el botón `/etc/fstab`:



```
$ sudo nano /etc/fstab
```

En este fichero veremos el disco o discos que tenemos y añadiremos una línea que es nuestro NFS. La línea debe ser algo como esto:

```
192.168.1.83:/mnt/data /mnt/data nfs por defecto 0 1
```

Y el archivo es, en mi caso, así:

```
# /etc/fstab: información estática del
sistema de archivos. #
# Utilice 'blkid' para imprimir el identificador Único universal de un
# dispositivo; esto puede usarse con UUID= como una forma más robusta de
# nombrar dispositivos # que funciona incluso si se añaden y quitan discos.
Ver fstab(5).
#
# <file system> <mount point> <type><options> <dump><pass> #
/ estaba en /dev/sda2 durante la instalación de curtin.
/dev/disk/by-uuid/726855df-27f8-4d31-9382-e71914e3bafb / ext4 por defecto 0 0
/swap.img          ninguno swap      sw          0          0

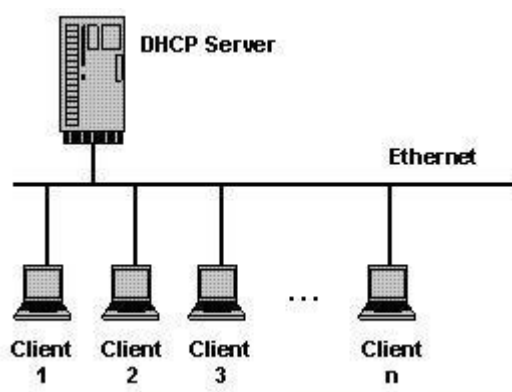
192.168.1.83:/mnt/data /mnt/data nfs por defecto 0 1
```

Guardamos, cerramos y reiniciamos. Cuando iniciemos sesión, vamos a /mnt/client1 y deberíamos ver la carpeta remota montada.

## 3. PROTOCOLO DHCP

### 3.1 ¿Qué es DHCP?

Un servidor de *Protocolo de Configuración Dinámica de Host* ( *DHCP* ) asigna dinámicamente direcciones IP y otros ajustes para una red dada a otros ordenadores cliente que están conectados a la red. Esto simplifica la administración de la red y facilita la conexión de nuevos equipos a la red.



Todas las direcciones IP de todos los ordenadores se almacenan en una base de datos que reside en un servidor.

Un servidor DHCP puede proporcionar ajustes de configuración utilizando dos métodos.

#### **Direcciones**

Este método se basa en la definición de un pool de direcciones IP para clientes DHCP ( también llamado pool de direcciones IP ) que suministran sus propiedades de configuración dinámicamente según lo soliciten los ordenadores cliente. Cuando un cliente DHCP deja de estar en la red durante un periodo determinado, la configuración caduca y la dirección IP sondeada se libera para ser utilizada por otros clientes DHCP.

#### **Dirección MAC**

Este método se basa en utilizar el protocolo DHCP para identificar la dirección de hardware única de cada tarjeta de red conectada a la red y, a continuación, se asigna una configuración constante así como la misma dirección IP cada vez que la configuración DHCP del cliente realiza una solicitud. al servidor DHCP desde el mismo dispositivo de red.

### 3.2 Instalar un servicio DHCP en Ubuntu y Debian

Para instalar el servidor de asignación automática de direcciones IP, ejecutamos el comando:

```
sudo apt-get install dhcp3-server
```

Este sencillo paso instala el servidor en nuestro linux.

### 3.3 Configuración del servidor DHCP

En el caso de que tengamos dos interfaces de red ( NICs ) en nuestro servidor Linux, tenemos que seleccionar cual vamos a utilizar para escuchar las peticiones DHCP. Para configurar el servicio, editamos el fichero `/etc/default/dhcp3-server` , y cambiamos `INTERFACES="eth0"` por la tarjeta de red interna.

Es necesario hacer una copia de seguridad del archivo de configuración:

```
cp /etc/dhcp3/dhcpd.conf /etc/dhcp3/dhcpd.conf.back
```

Configurar utilizando el método de rango de direcciones (IP pool) Editamos la configuración tecleando:

```
sudo nano /etc/dhcp3/dhcpd.conf
```

Y en este archivo cambiamos las siguientes secciones

```
default-lease-time 600;
max-lease-time 7200;
opción máscara de subred 255.255.255.0;
opción broadcast-address 192.168.1.255;
opción routers 192.168.1.1;
option nombre-dominio-servidores 192.168.1.9,
192.168.1.10; option nombre-dominio "guatewireless.org";
subred 192.168.1.0 máscara de red 255.255.255.0 {
rango 192.168.1.10 192.168.1.200;
}
```

Guardamos y salimos del fichero. El texto anterior configura el servidor DHCP con los siguientes parámetros:

- Asignación a los clientes de direcciones IP en el rango de 192.168.1.10 a

192.168.1.200

- Proporcionará la dirección IP durante un mínimo de 600 segundos y un máximo de 7200 segundos.
- Establezca la máscara de subred en 255.255.255.0
- Dirección de difusión 192.168.1.255
- Como puerta de enlace/puerta de red/enrutador la dirección 192.168.1.1
- Y los servidores 192.168.1.9 y 10 como sus servidores DNS

### Reservas de IP para determinadas máquinas

Para ello, al final del archivo añadimos el siguiente bloque

```
# SERVIDOR
host SERVIDOR {
    hardware ethernet xx:xx:xx:xx:xx:xx;
    dirección fija 192.168.1.210;
    opción broadcast-address 192.168.0.255;
}
```

Donde SERVIDOR es el nombre de la máquina, xx:xx:xx:xx:xx:xx es la dirección mac de la máquina, 192.168.1.210 es la IP que queremos reservar y 192.168.0.255 es la dirección broadcast

Guardamos el archivo con el que reiniciamos el servidor dhcp.

```
etc/init.d/dhcp3-server restart
```

### Configurar utilizando el método de dirección MAC

Con este método se pueden reservar algunas o todas las direcciones IP de nuestra red para determinadas máquinas. Como puedes ver, la configuración es muy similar a la anterior, con la salvedad de que para reservar la asignación de una IP a una determinada NIC ( interfaz de tarjeta de red ) debemos utilizar la etiqueta host

```
default-lease-time 600;
max-lease-time 7200;
opción máscara de subred 255.255.255.0;
opción broadcast-address 192.168.1.255;
opción routers 192.168.1.1;
```

```
option nombre-dominio-servidores 192.168.1.9,  
192.168.1.10; option nombre-dominio "guatewireless.org";  
subred 192.168.1.0 máscara de red 255.255.255.0 {  
rango 192.168.1.10 192.168.1.200;  
}  
host oracle{  
hardware ethernet 00:03:47:31:e1:7f;  
dirección fija 192.168.1.20;  
}  
impresora host {  
hardware ethernet 00:03:47:31:e1:b0;  
dirección fija 192.168.1.21;  
}
```

Ahora reiniciamos el servidor dhcp ejecutando el siguiente comando:

```
sudo /etc/init.d/dhcp3-server restart
```

### 3.4 Configuración del cliente DHCP en Linux Ubuntu

Si quieres configurar un escritorio o máquina con linux como cliente DHCP, sigue estos pasos:

- Editamos el archivo de interfaces de red

```
sudo vi /etc/network/interfaces
```

- Debemos tener las siguientes líneas, teniendo en cuenta que eth0 es un ejemplo

```
auto lo eth0  
iface eth0 inet dhcp  
iface lo inet loopback
```

- Guardamos y salimos del archivo
- Reiniciamos los servicios de red de Linux Ubuntu

```
sudo /etc/init.d/networking restart
```

Para conocer las direcciones asignadas a las máquinas cliente

```
tail -n 15 /var/lib/dhcp3/dhclient.*.leases
```