

Francisco Joaquín Murcia Gómez
48734281H
Grado en ingeniería informática UA

ADMINISTRACION DE SISTEMAS OPERATIVOS Y DE REDES DE COMPUTADORES

Hito 2



índice

Licencias.....	3
CentOS	4
Particionado.....	4
Administración remota	6
SSH/SFTP/SCP	6
RDP	9
VNC	10
Networking	11
DHCP	11
Servidor OwnCloud	12
Servidor GIT.....	15
File Server	15
NFS	15
SAMBA	16
FreeNAS.....	17
Impresión remota	18
CUPS	18
Base de datos y web services.....	20
Base de datos	20
FreeBSD.....	23
Particionado.....	23
Administración remota	25
SSH/SFTP/SCP	25
Networking	26
DHCP	26
Git server	28
OwnCloud server.....	29
Servicios de impresión.....	29
Cups	29
Bases de datos y web services	30

Base de datos	30
File server.....	32
FreeNAS.....	32
Windows.....	34
Partición	34
Administración remota.....	34
SSH/SFTP/SCP	34
VNC	35
RDP.....	35
Networkig.....	36
DHCP	36
File server.....	38
FreeNAS.....	38
NFS/SAMBA	39

Licencias

Una licencia es un contrato mediante el cual una persona recibe de otra el derecho de uso, de copia, de distribución, de estudio y de modificación.

En el caso del software, serian todos los permisos que los desarrolladores otorgan a los usuarios, en estos se especifica:

- Los límites en la responsabilidad por fallos.
- El plazo de cesión de los derechos.
- Cesión del programa a terceros
- Reinstalación del programa en equipos distintos al original

En los sistemas operativos a analizar, Windows Server usa EULA, CentOS usa GPL GNU y FreeBSD, BSD

EULA:

- De pago, con precios entre 6€ y 800€
- Prohíbe la copia y modificación.
- Las actualizaciones del sistema pueden modificar la licencia
- Se conceden permisos a la distribuidora para realizar cambios en nuestro sistema
- Se conceden permisos a la distribuidora para utilizar nuestra información
- Podría dejar de funcionar si se realizan cambios de hardware

GNU GPL:

- Gratuito
- Permite la copia, modificación y redistribución del software.
- Puede ser vendido y se puede cobrar por los servicios sobre el software
- Se debe proveer con el código fuente
- Si hay un cambio en la licencia, los términos generales de la licencia existente se mantienen.

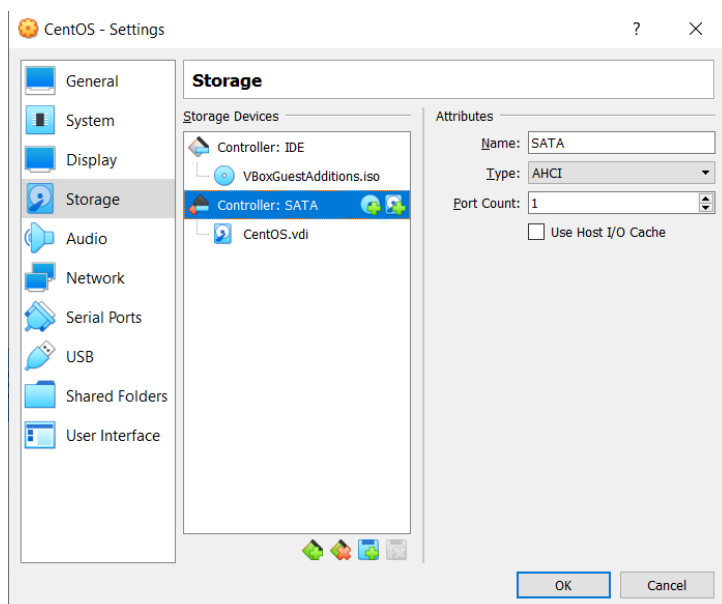
BSD

- Gratuito
- Permite la copia, modificación y redistribución del software
- BSD permite el cobro por la distribución
- El desarrollador no está en la obligación de incluir el código fuente

CentOS

Particionado

El apartado configuración de la máquina virtual, en el apartado de almacenamiento, clicamos en añadir un disco en el controlador SATA,



Una vez creado el disco para habilitarlo introducimos `fdisk /dev/sdb`, con el comando `p` nos muestra los discos listos para instalar.

```
[root@localhost ~]# fdisk /dev/sdb

Bienvenido a fdisk (util-linux 2.32.1).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

El dispositivo no contiene una tabla de particiones reconocida.
Se ha creado una nueva etiqueta de disco DOS con el identificador de disco 0xbb31abab.

Orden (m para obtener ayuda): p
Disco /dev/sdb: 8 GiB, 8589934592 bytes, 16777216 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0xbb31abab
```

Con el comando `n` introducimos el nuevo disco, después nos pedirá una serie de requisitos si la queremos primaria introducimos `p`, el número de partición `1` y los sectores lo dejamos que el sistema elija.

```

Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (0 primaria(s), 0 extendida(s), 4 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (1-4, valor predeterminado 1): 1
Primer sector (2048-16777215, valor predeterminado 2048):
Último sector, +sectores o +tamaño{K,M,G,T,P} (2048-16777215, valor predeterminado 16777215):

Crea una nueva partición 1 de tipo 'Linux' y de tamaño 8 GiB.

```

Para definir el tipo de partición tecleamos t, e introducimos el código del tipo de partición seleccionado, para saber los tipos, introducimos l, en el caso de Linux, es 83, así que lo introducimos.

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
f  W95 Ext'd (LBA) 54  OnTrackDM6  a6  OpenBSD  ee  GPT
10 OPUS 55 EZ-Drive a7 NeXTSTEP ef EFI (FAT-12/16/
11 FAT12 oculta 56 Golden Bow a8 UFS de Darwin f0 inicio Linux/PA
12 Compaq diagnost 5c Priam Edisk a9 NetBSD f1 SpeedStor
14 FAT16 oculta <3 61 SpeedStor ab arranque de Dar f4 SpeedStor
16 FAT16 oculta 63 GNU HURD o SysV af HFS / HFS+ f2 DOS secondary
17 HPFS/NTFS ocult 64 Novell Netware b7 BSDI fs fb VMFS de VMware
18 SmartSleep de A 65 Novell Netware b8 BSDI swap fc VMKCORE de VMwa
1b FAT32 de W95 oc 70 DiskSecure Mult bb Boot Wizard hid fd Linux raid auto
1c FAT32 de W95 (L 75 PC/IX bc Acronis FAT32 L fe LANstep
1e FAT16 de W95 (L 80 Minix antiguo be arranque de Sol ff BBT

Orden (m para obtener ayuda): t

Se ha seleccionado la partición 1
Código hexadecimal (escriba L para ver todos los códigos): 83
Se ha cambiado el tipo de la partición 'Linux' a 'Linux'.

```

Para finalizar de montar el disco tenemos que crear la carpeta data, para ello, usaremos mkdir /data y para montarla en el disco, mount /dev/sdb1 /data

```

[root@localhost ~]# mkdir /data
mkdir: no se puede crear el directorio «/data»: El fichero ya existe
[root@localhost ~]# mount /dev/sdb1 /data
[root@localhost ~]# mount
sysfs on /sys type sysfs (rw, nosuid, nodev, noexec, relatime, seclabel)

```

Para conformar ejecutamos mount y al final del todo ha de aparecer la carpeta data en el disco.

```

liet,check=s,map=n,blocksize=2048,uid=1000,gid=1000,
)
/dev/sdb1 on /data type ext4 (rw,relatime,seclabel)
[root@localhost ~]#

```

Administración remota

SSH/SFTP/SCP

SSH viene instalado en CentOS, para configurarlo hay que acceder al fichero /etc/ssh/sshd_config, en el aparece toda la configuración de SSH, en el añadimos lo siguiente:

- Añadir: AllowUsers <usuarios permitidos>
- Añadir PubkeyAuthentication yes
- Modificar: UsePAM yes
- Modificar: PermitRootLogin no

Una vez hecho eso para poder conectarse hay que cambiar la configuración del kernel modificándolo en el fichero /etc/sysconfig/selinux, SELINUX estará en enable, pues lo ponemos en disabled.

```
#      enforcing - SELinux security policy is enforcing
#      permissive - SELinux prints warnings without enforcing
#      disabled - No SELinux policy is loaded
SELINUX=disabled
# SELINUXTYPE= can take one of the following values:
#      targeted - Targeted policy
#      minimum - Modification of the base policy
#      mls - Multi Level Security
SELINUXTYPE=targeted
```

Reiniciamos y ya en el host, introducimos ssh <usuario>@<ip servidor> nos pedirá la contraseña del usuario y ya estaríamos conectados al servidor

```
C:\Users\FRAN>ssh -p 22 fran@169.254.0.9
fran@169.254.0.9's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Wed Nov  4 13:30:20 2020 from 169.254.0.4
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Wed Nov  4 13:30:20 2020 from 169.254.0.4
[fran@localhost ~]$
```

Para generar la clave publica, usaremos el comando ssh-keygen -t rsa en el host

```

C:\Users\FRAN> ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\FRAN/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\FRAN/.ssh/id_rsa.
Your public key has been saved in C:\Users\FRAN/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:/wRs8x1xBH0yWz7HpajjyN4rlcaWW4Lr4NL9qGcP1Vw fran@DESKTOP-FB4AP4E
The key's randomart image is:
+---[RSA 2048]---+
|                 |
|      +.o        |
|      . *        |
|      = E .      |
|      . + B o    |
|      So*oB +    |
|      .+0=.=     |
|      ....=+++   |
|      ...O**+.+  |
|      .O**+*..   |
|                 |
+---[SHA256]---+

C:\Users\FRAN>

```

En la Windows powershell introducimos la siguiente sentencia para asignarla al servidor:

```
cat ~/.ssh/id_rsa.pub | ssh <usuario>@<ip servidor> "mkdir ~/.ssh; cat >> ~/.ssh/authorized_keys"
```

después en el servidor, signamos los permisos adecuados a la carpeta .ssh y el fichero que contiene las claves

```

Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[fran@localhost .ssh]$ chmod 700 ~/.ssh
[fran@localhost .ssh]$ chmod 600 authorized_keys
[fran@localhost .ssh]$ s

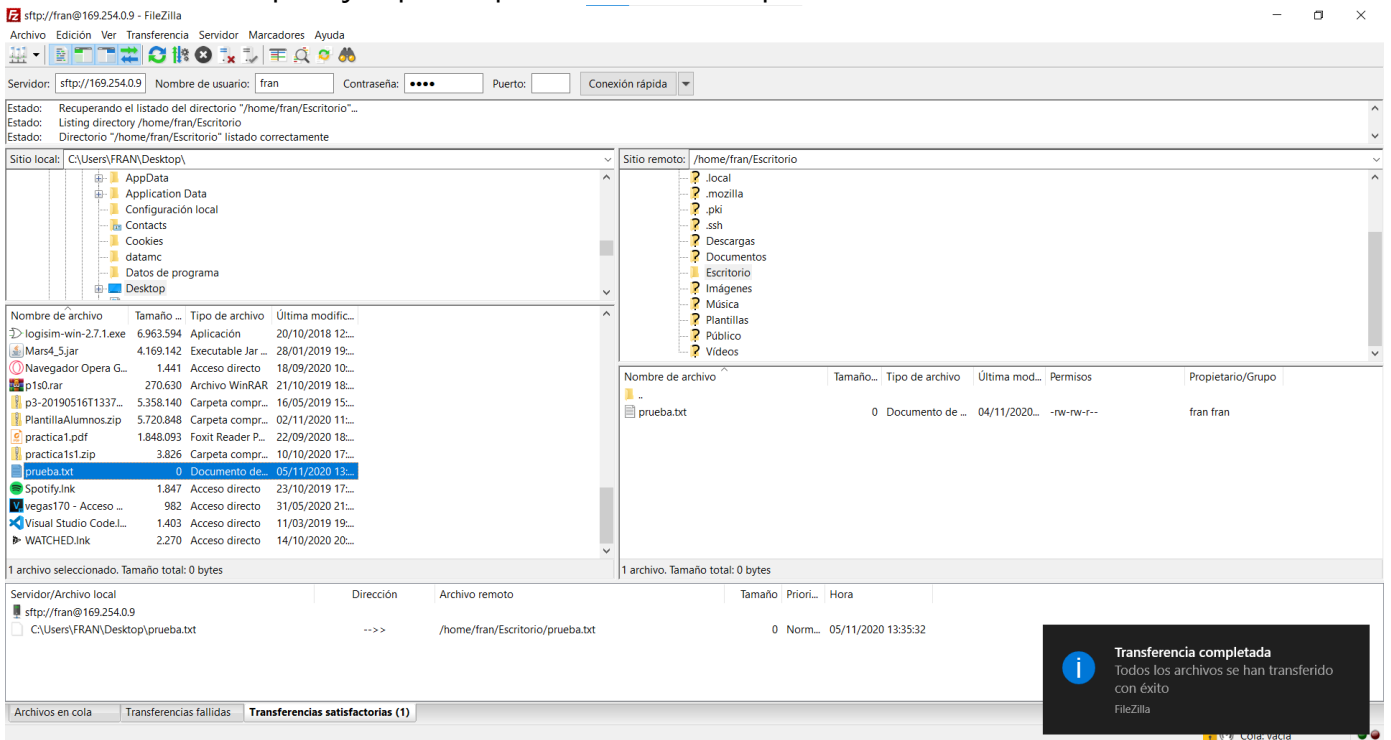
```

Para realizar la conexión SFTP, se ha utilizado el programa de gestión de archivos filezilla.

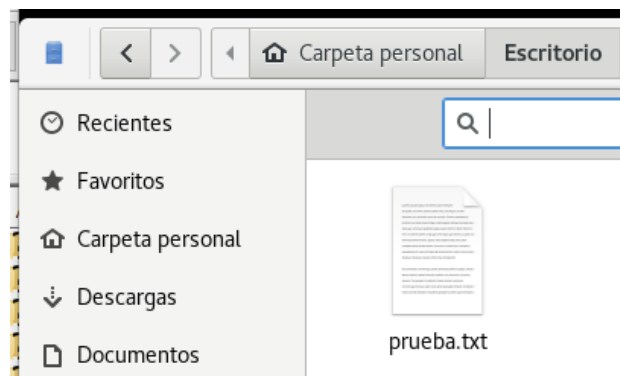
Nos conectamos a nuestro servidor con la ip usuario y contraseña

The screenshot shows the FileZilla connection window. The 'Servidor:' field contains '169.254.0.9', the 'Nombre de usuario:' field contains 'fran', and the 'Puerto:' field contains '22'. There is a 'Conexión rápida' button and a dropdown arrow next to it. The 'Contraseña:' field is masked with dots.

Y para pasar archivos del host al servidor o viceversa solo hay que pinchar y arrastrar, por ejemplo, he pasado un archivo prueba al servidor



Si nos vamos al escritorio que es a donde lo he pasado, vemos que el archivo aparece.



Para comprobar el servicio SCP he enviado un documento de prueba

`scp -p <puerto> <archivo> <usuario>@<ip>:<ruta destino>`

```
C:\Users\FRAN\Desktop>scp -p 22 scpPrueba.txt fran@169.254.0.9:/home/fran/Escritorio
22: No such file or directory
scpPrueba.txt                                     100%   0   0.0KB/s   00:00

C:\Users\FRAN\Desktop>
```

Si hacemos un ls en escritorio, observamos que el archivo se ha enviado

```
[fran@localhost Escritorio]$ ls
memoria prueba.txt scpPrueba.txt
[fran@localhost Escritorio]$
```

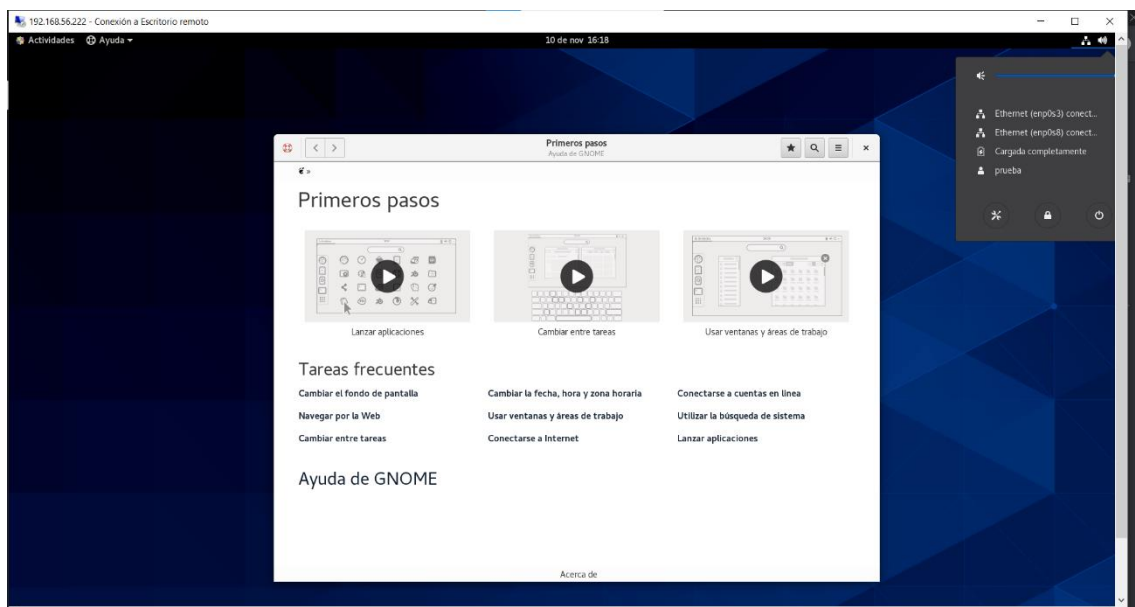
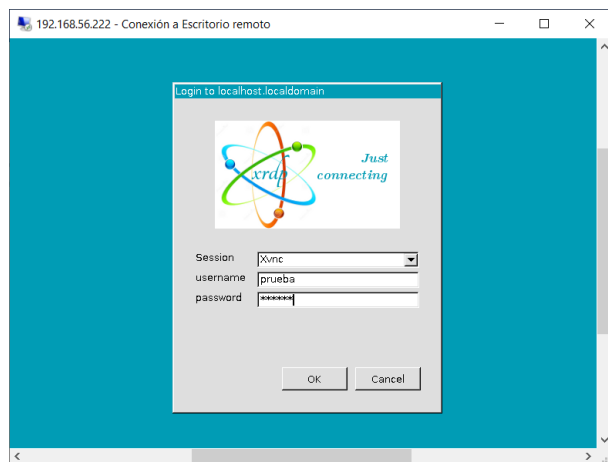
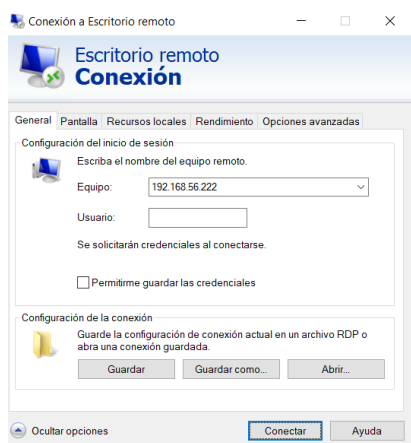
RDP

Para activar el servicio RDP necesitamos instalar el xrdp, para ello previamente se instalara epel-release para que deje instalar xrdp

En segundo paso, activamos el servicio con el comando `systemctl start xrdp` y lo comprobamos con `systemctl status xrdp`.

```
[root@localhost ~]# systemctl status xrdp
● xrdp.service - xrdp daemon
   Loaded: loaded (/usr/lib/systemd/system/xrdp.service; disabled; vendor preset: disabled)
   Active: active (running) since Tue 2020-11-10 15:14:15 CET; 55min ago
     Docs: man:xrdp(8)
           man:xrdp.ini(5)
  Main PID: 4176 (xrdp)
    Tasks: 1 (limit: 23823)
   Memory: 1.3M
    CGroup: /system.slice/xrdp.service
            └─4176 /usr/sbin/xrdp --nodaemon
```

Con la aplicación para escritorios remotos de Windows colocamos la ip del servidor y nos conectamos



VNC

Para activar el servicio de vnc hay que instalat TigerVNC en primer lugar "dnf install tigervnc-server tigervnc-server-module -y"

Después hemos de crear el servicio de vnc, para ello modificamos el siguiente archivo [/etc/systemd/system/vncserver@.service](#) y añadiremos lo siguiente:

```
root@localhost:~
GNU nano 2.9.8 /etc/systemd/system/vncserver@.service

[Unit]
Description=Remote Desktop VNC Service
After=syslog.target network.target

[Service]
Type=forking
WorkingDirectory=/home/franvnc
User=franvnc
Group=franvnc

ExecStartPre=/bin/sh -c '/usr/bin/vncserver -kill %i > /dev/null 2>&1 || :'
ExecStart=/usr/bin/vncserver -autokill %i
ExecStop=/usr/bin/vncserver -kill %i

[Install]
WantedBy=multi-user.target
```

Después recargamos el fichero con `systemctl daemon-reload`, después activamos el servicio con `systemctl start vncserver@:1.service` y por ultimo `systemctl enable vncserver@:1.service`.

Comprobamos el estado con `systemctl status vncserver@:1.service`

```
root@localhost:~
[root@localhost ~]# systemctl start vncserver@:1.service
[root@localhost ~]# systemctl enable vncserver@:1.service
Created symlink /etc/systemd/system/multi-user.target.wants/vncserver@:1.service -> /etc/systemd/system/vncserver@.service.
[root@localhost ~]# nano /etc/systemd/system/vncserver@.service
[root@localhost ~]# [root@localhost ~]# systemctl status vncserver@:1.service
● vncserver@:1.service - Remote Desktop VNC Service
   Loaded: loaded (/etc/systemd/system/vncserver@.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2020-11-10 17:03:12 CET; 4min 0s ago
     Tasks: 184 (limit: 23823)
    Memory: 358.6M
   CGroup: /system.slice/system-vncserver.slice/vncserver@:1.service
           └─3546 /usr/bin/Xvnc :1 -auth /home/franvnc/.Xauthority -desktop localhost.localdomain:1 (franvnc) -fp catala
             └─3555 sh -c (/home/franvnc/.vnc/xstartup; /usr/bin/vncserver -kill :1) >> '/home/franvnc/.vnc/localhost.loc
             └─3556 /bin/sh /home/franvnc/.vnc/xstartup
             └─3557 /usr/libexec/gnome-session-binary
```

Nos conectamos a la dirección 192.168.56.222 en el puerto 5901 en VNCViewer



Networking

DHCP

En primer lugar, tenemos que colocar la ip de la interfaz host-only en estática, para ello modificaremos el fichero `/etc/sysconfig/network-scripts/ifcfg-enp0s8`

- Modificar `BOOTPROTO=none`
- Añadir `IPADDR=192.168.56.222`
- Añadir `NETMASK=255.255.255.0`

Después reiniciamos la interfaz `enp0s8` con `"nmcli con reload"` y después con `"nmcli con up enp0s8"`

Desactivamos el servidor dhcp de VirtualBox nos descargamos `"dhcp-server"`

Modificamos el archivo `/etc/dhcp/dhcpd.conf`:

```
[root@localhost ~]# cat /etc/dhcp/dhcpd.conf
#
# DHCP Server Configuration file.
#   see /usr/share/doc/dhcp-server/dhcpd.conf.example
#   see dhcpd.conf(5) man page
#
shared-network network.net {
    subnet 192.168.56.0 netmask 255.255.255.0 {
        #option routers 192.168.56.1;
        option subnet-mask 255.255.255.0;
        option broadcast-address 192.168.56.255;
        #option domain-name-servers 192.168.2.100;
        range 192.168.56.51 192.168.56.100;
    }
}
host learn {
    option host-name "nodoA.network.net";
    hardware ethernet 00:25:d3:66:63:b3;
    fixed-address 192.168.2.101;
}
[root@localhost ~]#
```

Finalmente activamos dhcp `systemctl start dhcpd`.

Encendemos una maquina de prueba, en micaso un linuxMint y vemos que la ip esta en el rango adecuado

```
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.56.94 netmask 255.255.255.0 broadcast 192.168.56.255
    inet6 fe80::d9e8:a299:32e1:bdbe prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:dc:75:e6 txqueuelen 1000 (Ethernet)
```

Servidor OwnCloud

Para hacer un servidor owncloud, necesitamos instalarnos previamente LAMP (Linux – Apache – MySQL – PHP):

```
yum install httpd httpd-tools
yum install mariadb-server mariadb -y
yum install php
dnf install php-curl php-gd php-intl php-json php-ldap php-mbstring php-mysqlnd
php-xml php-zip php-opcache
```

Despues nos instalamos owncloud y extraemos su carpeta `/var/www`:

```
wget https://download.owncloud.org/community/owncloud-10.3.2.tar.bz2
tar -jxf owncloud-10.3.2.tar.bz2 -C /var/www/
```

Creamos la base de datos, en ella le colocamos usuario y contraseña para la base de datos

```
[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 22
Server version: 10.3.17-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE owncloud_db;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> GRANT ALL ON owncloud_db.* TO 'usuario'@'localhost' IDENTIFIED BY '1234';
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> EXIT;
Bye
[root@localhost ~]#
```

```
CREATE DATABASE owncloud_db;
GRANT ALL ON owncloud_db.* TO 'usuario'@'localhost' IDENTIFIED BY 'contraseña';
FLUSH PRIVILEGES;
EXIT;
```

Modificamos el archivo de configuración /etc/httpd/conf.d/owncloud.conf y colocamos lo siguiente:

```
Alias /owncloud "/var/www/owncloud/"
```

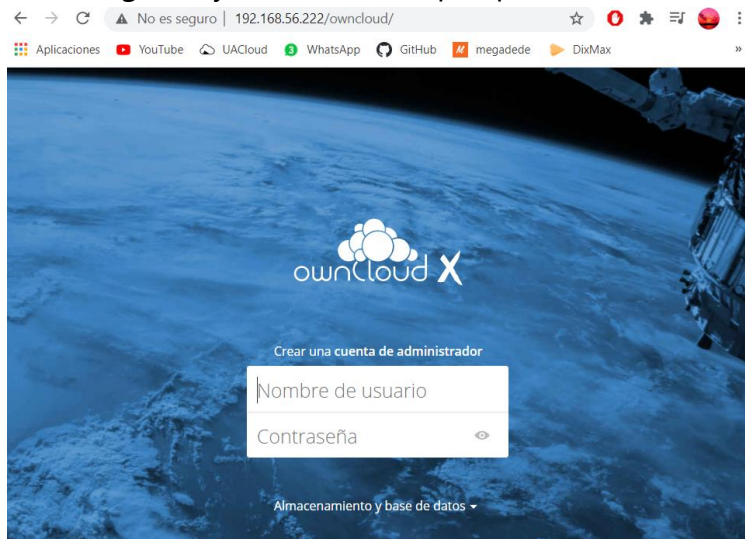
```
<Directory /var/www/owncloud/>
Options +FollowSymLinks
AllowOverride All
```

```
<IfModule mod_dav.c>
Dav off
</IfModule>
```

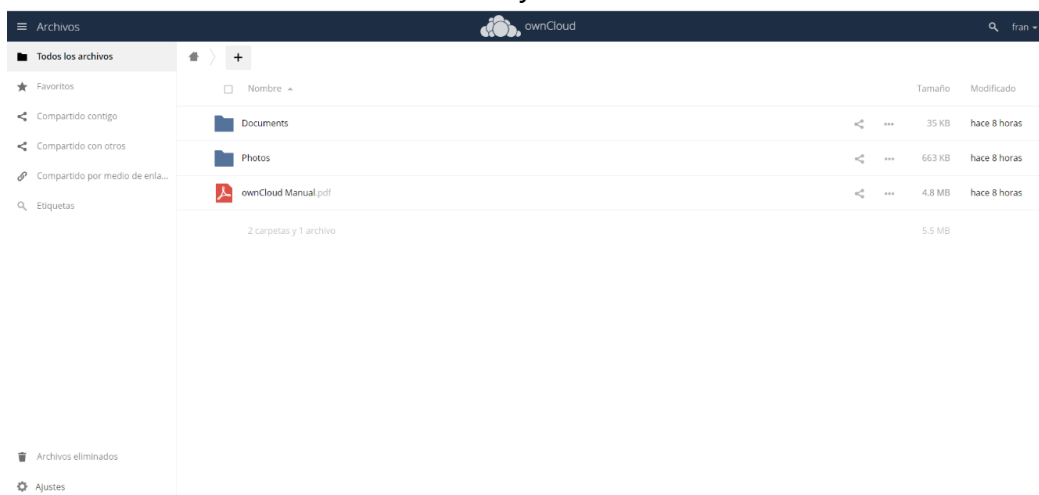
```
SetEnv HOME /var/www/owncloud
SetEnv HTTP_HOME /var/www/owncloud
```

```
</Directory>
```

Por abrimos un navegador y colocamos `http://ip-servidor/owncloud`

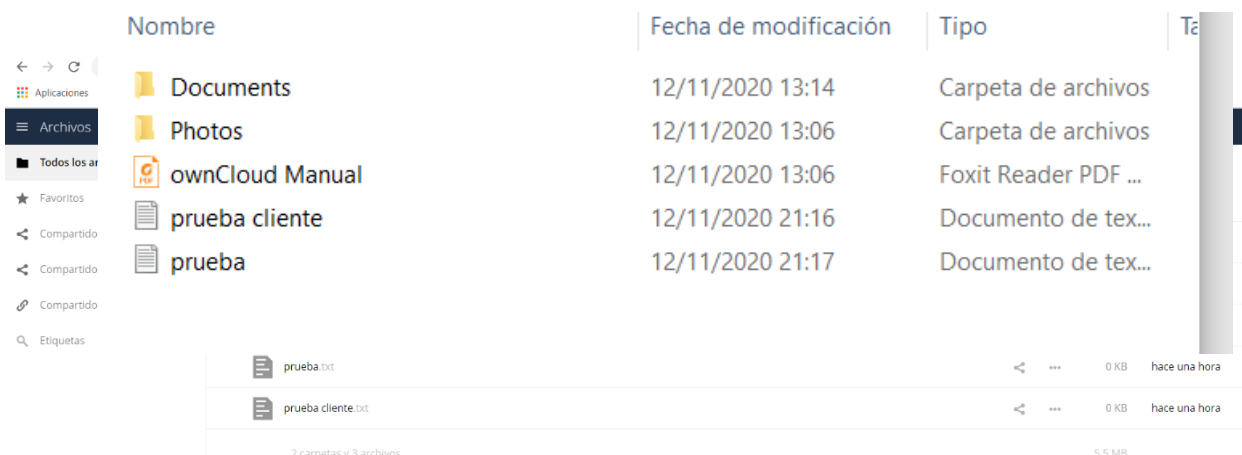


Aquí podremos crearnos un usuario administrador y en la pestaña almacenamiento y base de datos, en la opción MySQL tendremos que colocar el usuario contraseña de la base de datos y la base de datos.



Y ya tendríamos nuestro servicio owncloud.

En Windows nos descargamos el cliente y nos logueamos colocamos una carpeta en la carpeta sincronizada y en la web aparecerá



Servidor GIT

Para instalar un servidor git hemos de instalarnos previamente:

```
git, yum install git
```

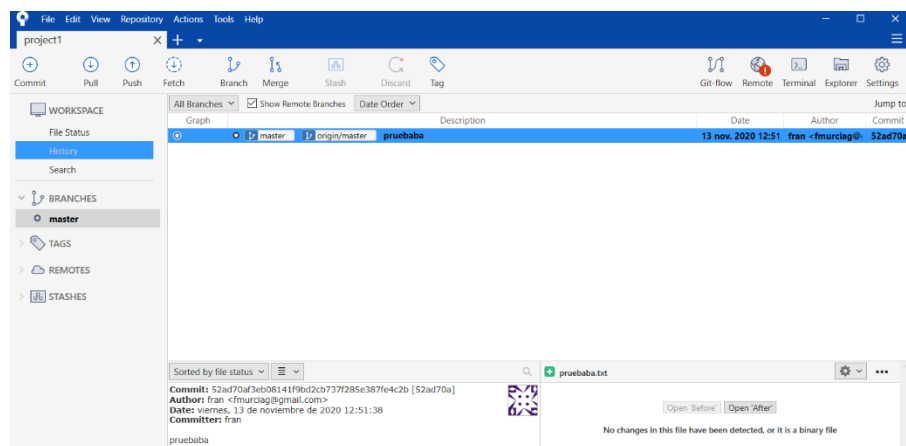
A continuación, nos creamos un repositorio en una carpeta:

```
mkdir ~/repo  
cd ~/repo  
git init --bare --shared project1
```

A continuación, habilitamos post-update hook de la siguiente manera:

```
cd ~/repo/project1/hooks/  
cp post-update.sample post-update
```

Una vez hecho todo esto, nos descargamos un cliente, en mi caso SourceTree, colocamos la dirección del repositorio en el instalador y ya podríamos trabajar con el:



File Server

NFS

En primer lugar, crearemos una carpeta para compartirla `mkdir /data/nfs_shared`, y nos instalamos el servicio `yum -y install nfs-utils`.

Modificamos el fichero `/etc/exports` y añadimos lo siguiente:

```
/directorio_a_compartir 192.168.56.1/24(rw,no_root_squash)
```

Después activamos el servicio y comprobamos su estado


```
[root@localhost ~]# systemctl start nfs-server
a[root@localhost ~]# systemctl status nfs-server
● nfs-server.service - NFS server and services
   Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; vendor preset: enabled)
   Active: active (exited) since Wed 2020-11-11 10:51:11 COT; 1min 28s ago
     Process: 3825 ExecStart=/bin/sh -c if systemctl -q is-running nfs-server.service; then systemctl start nfs-server.service; fi
     Process: 3810 ExecStart=/usr/sbin/rpc.nfsd (code=exited, status=0/SUCCESS)
     Process: 3808 ExecStartPre=/usr/sbin/exportfs -rv (code=exited, status=0/SUCCESS)
     Main PID: 3825 (code=exited, status=0/SUCCESS)
```

SAMBA

Para realizar samba primero tenemos que descargarnos el servicio de samba:

```
dnf -y install samba samba-client samba-common
```

Una vez instalado, creamos un directorio que es el que será compartido:

```
mkdir -p /data/samba_sared
```

Modificamos el archivo de configuración de samba:

```
nano /etc/samba/smb.conf
```

- workgroup = workgroup
- [samba_sared]
 - comment=carpeta compartida
 - path=/data/samba_sared
 - browsable =yes
 - writable = yes
 - guest ok = yes
 - read only = no

Nos creamos una contraseña de samba con el comando `smbpasswd -a <usuario>` y encendemos el servicio `systemctl start nmbd smb`

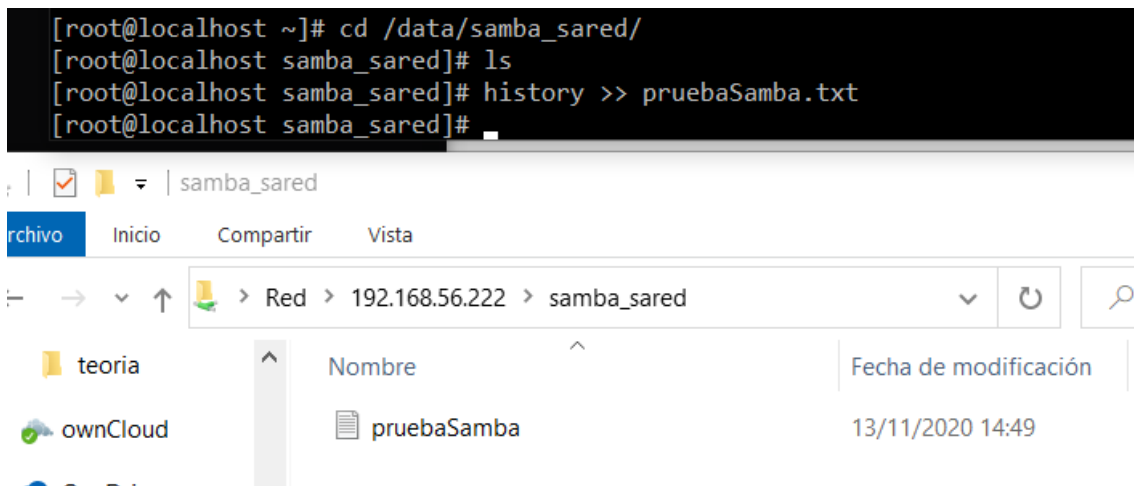
Damos permiso al firewall para que habilite samba:

```
firewall-cmd --add-service=samba --zone=public --permanent
```

En el host abrimos la ventana de ejecución `win+R` e introducimos:

```
\\<ipServidor>
```

Nos abrirá la carpeta compartida y ya podríamos compartir archivos:



FreeNAS

Nos descargamos la iso de freeNass y creamaos una maquina con esa iso, instalamos la maquina y nso metemos en a url que nos indica la maquina FreeNAS

En ella nos creamos una pool con estas caracteriaticas :

The screenshot shows the FreeNAS web interface. At the top, it indicates the system is 'HEALTHY' with 10.17 GiB (78%) used and 2.91 GiB free. Below this is a table showing the storage configuration:

Name	Type	Used	Available	Compression	Compression Ratio	Readonly	Dedup	Comments
cestosStorage	dataset	10.17 GiB	2.91 GiB	lz4	49.11x	false	off	
volumen	zvol	10.16 GiB	13.07 GiB	Inherits (lz4)	23.86x	false	off	

En la página, habilitamos el servicio ISCSI, y creamos un disco con dicha característica, luego nos vamos a la maquina CentOS y nos instalamos ISCSI dnf install iscsi-utils

Después ejecutamos los siguientes comandos:

```
[root@localhost ~]# iscsiadm -m iface -I enp0s8 -o new
New interface enp0s8 added
[root@localhost ~]# iscsiadm -m discovery -t st -p 192.168.56.62
192.168.56.62:3260,-1 iqn.2005-10.org.freenas.ctltarget
[root@localhost ~]# chkconfig iscsid on
Nota: Reenviando petición a 'systemctl enable iscsid.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/iscsid.service → /usr/lib/systemd/system/iscsid.service.
[root@localhost ~]# iscsiadm -m node -T iqn.2005-10.org.freenas.ctltarget -p 192.168.56.62 -l -I enp0s8
iscsiadm: No records found
[root@localhost ~]# iscsiadm -m node -T iqn.2005-10.org.freenas.ctltarget -p 192.168.56.62 -l -I enp0s8
Logging in to [iface: enp0s8, target: iqn.2005-10.org.freenas.ctltarget, portal: 192.168.56.62,3260]
Login to [iface: enp0s8, target: iqn.2005-10.org.freenas.ctltarget, portal: 192.168.56.62,3260] successful.
[root@localhost ~]# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sda	8:0	0	20G	0	disk	
├─sda1	8:1	0	1G	0	part	/boot
├─sda2	8:2	0	19G	0	part	
├─┌cl-root	253:0	0	17G	0	lvm	/
└─┌cl-swap	253:1	0	2G	0	lvm	[SWAP]
sdb	8:16	0	8G	0	disk	
└─sdb1	8:17	0	8G	0	part	/data
src	8:32	0	10G	0	disk	
sr0	11:0	1	7,7G	0	rom	/run/media/fran/CentOS-8-2-2004-x86_64-dvd

```
[root@localhost ~]#
```

Una vez hecho, montamos el disco nuevo con el proceso explicado en el apartado de [particiones](#)

Como resultado tenemos el disco sdc instalado y montado:

```
Disco /dev/sdc: 10 GiB, 10737434624 bytes, 20971552 sectores
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 16384 bytes
Tamaño de E/S (mínimo/óptimo): 16384 bytes / 1048576 bytes
[root@localhost ~]#
```

Impresión remota

CUPS

Para instalar cups en centos necesitamos instalar una impresora, en nuestro caso la cups-pdf, para ello ejecutaremos los siguientes comandos:

```
yum update
yum -y install cups gcc gcc-c++ cups-devel tar wget
wget https://www.cups-pdf.de/src/cups-pdf_3.0.1.tar.gz
tar -xvf cups-pdf_3.0.1.tar.gz
cd cups-pdf-3.0.1/src/
gcc -O9 -s cups-pdf.c -o cups-pdf -lcups
chmod 700 cups-pdf
cp -p cups-pdf /usr/lib/cups/backend/
cd ../extra
cp cups-pdf.conf /etc/cups/
cp CUPS-PDF_opt.ppd /usr/share/cups/model/
firewall-cmd --zone=public --add-port=631/tcp --permanent
firewall-cmd --reload
sed -i "s/Allow \@LOCAL/Allow all/g" /etc/cups/cupsd.conf
systemctl restart cups
systemctl enable cups
cupsctl --remote-admin
```

después modificamos el fichero /etc/cups/cupsd.conf y añadimos:

```
Port 631
Listen localhost:631
Listen 192.168.56.222:631
```

Después entramos en <https://192.168.56.222:631/> y en administración, añadir impresoras

Añadir impresora

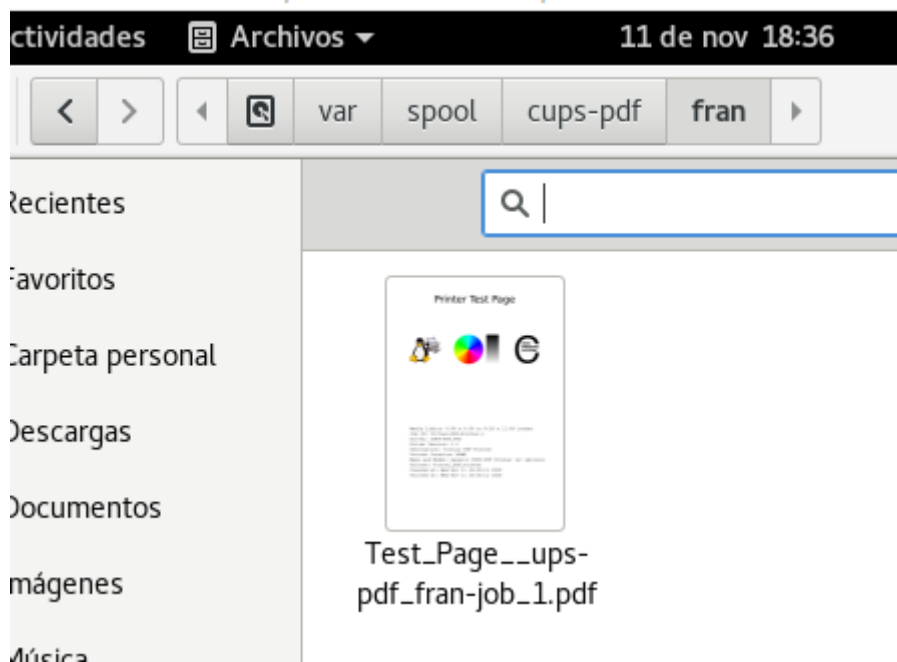
Impresoras locales: ☒ CUPS-PDF (Virtual PDF Printer)

Impresoras en red descubiertas:

Otras impresoras en red:

- ☐ Protocolo de Impresión de Internet IPP (ipp)
- ☐ AppSocket/HP JetDirect
- ☐ Equipo o impresora LPD/LPR
- ☐ Protocolo de Impresión de Internet IPP (http)
- ☐ Backend Error Handler
- ☐ Protocolo de Impresión de Internet IPP (https)
- ☐ Protocolo de Impresión de Internet IPP (ipp)

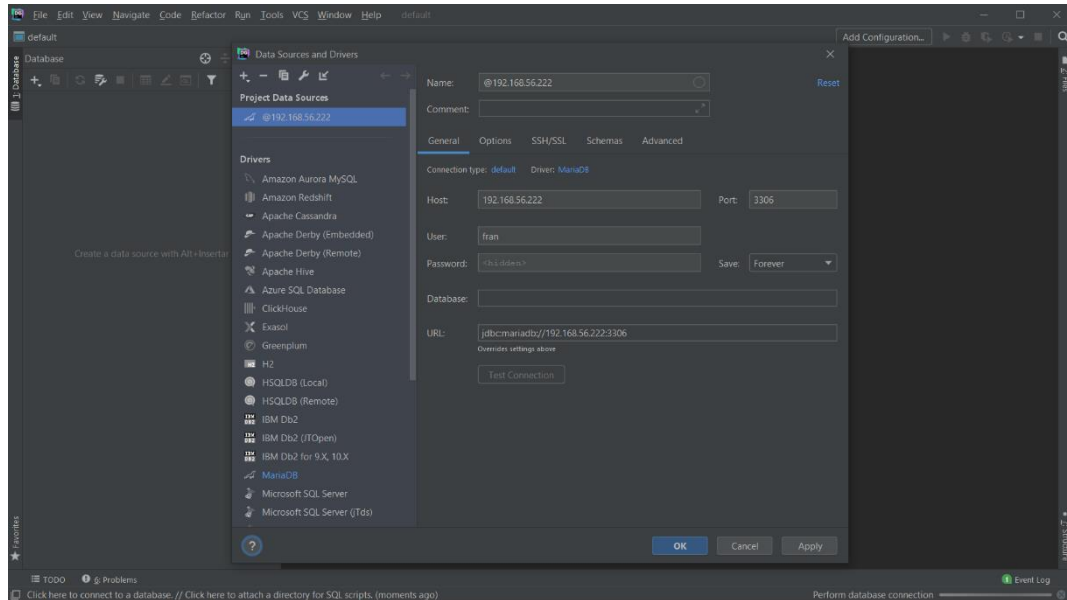
Añadimos la CUPS_PDF y cuando imprimimos una pagina de prueba nos aparecerá en el directorio /var/spool/cups-pdf



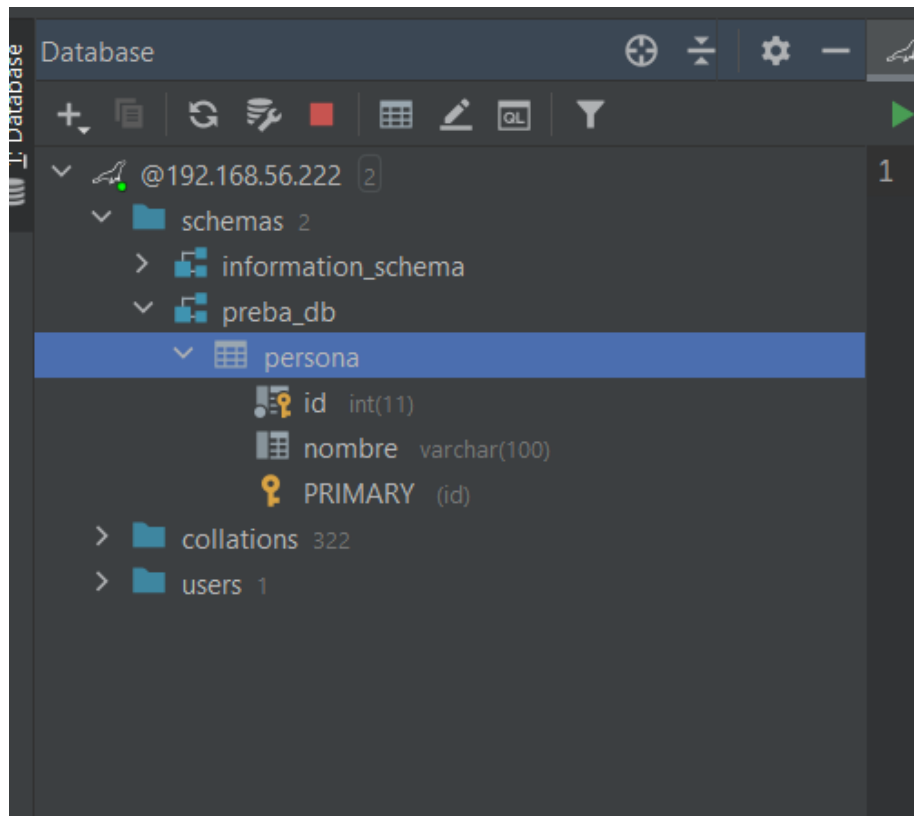
Base de datos y web servicios

Base de datos

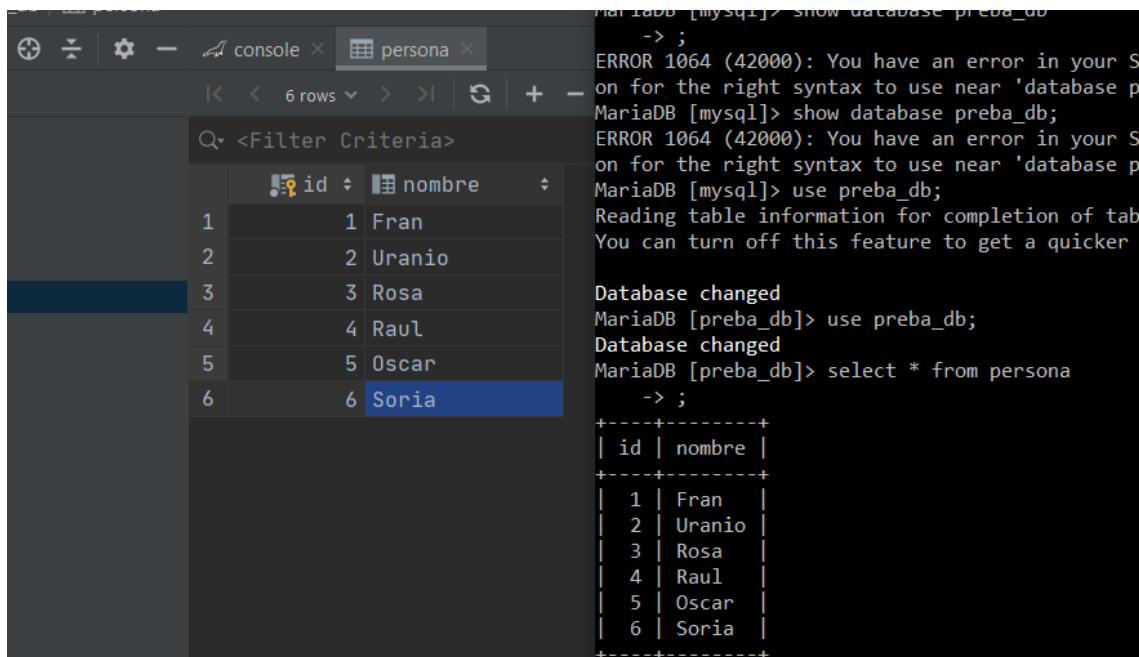
Utilizando el proceso de instalación utilizado en OwnCloud, creamos una base de datos, luego nos descargamos el cliente de bases de datos DataGrep y colocamos la base de datos:



Nos aparecerán nuestras bases de datos a la izquierda:



Con la interfaz de DataGrep nos creamos unos usuarios:



The screenshot shows the DataGrep application interface. On the left, a table with columns 'id' and 'nombre' is displayed, containing six rows of user data. On the right, a terminal window shows the execution of MySQL commands to create a database, use it, and query the 'persona' table.

id	nombre
1	Fran
2	Uranio
3	Rosa
4	Raul
5	Oscar
6	Soria

```
MariaDB [mysql]> show database preba_db;
-> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
MariaDB [mysql]> show database preba_db;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
MariaDB [mysql]> use preba_db;
Reading table information for completion of table
You can turn off this feature to get a quicker
Database changed
MariaDB [preba_db]> use preba_db;
Database changed
MariaDB [preba_db]> select * from persona
-> ;
+----+-----+
| id | nombre |
+----+-----+
| 1  | Fran   |
| 2  | Uranio |
| 3  | Rosa   |
| 4  | Raul   |
| 5  | Oscar  |
| 6  | Soria  |
+----+-----+
```

Modificamos el fichero `/etc/httpd/conf/httpd.conf` y modificamos

`ServerName asorc:80`

En centos nos creamos una queri en php para que nos muestre la base de datos:

`/var/www/html/index.php`

```
<?php
$servername = "localhost";
$username = "fran";
$password = "fran";
$dbname = "preba_db";

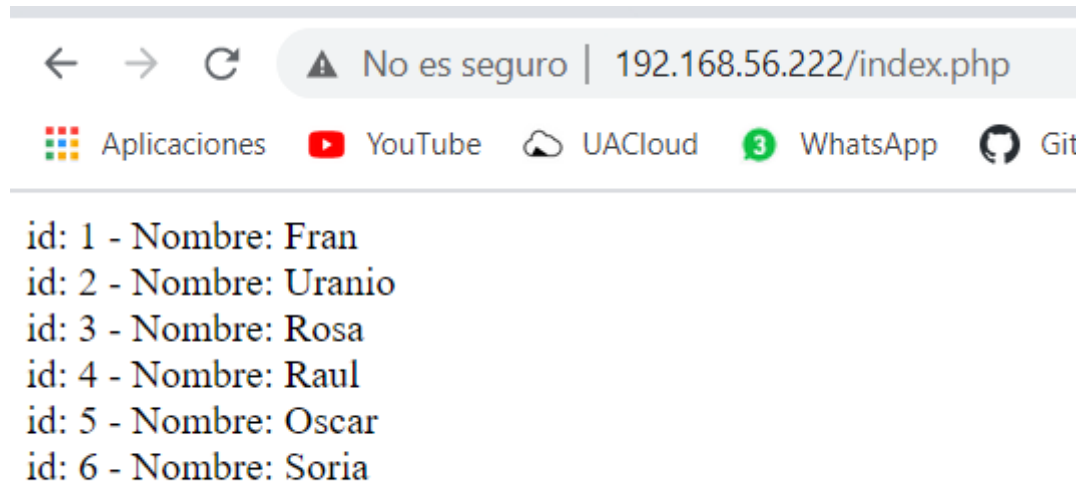
//Crear conexion
$conn = new mysqli($servername, $username, $password, $dbname);
//Comprobar conexion
if($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT * FROM persona";
$result = $conn->query($sql);

if($result->num_rows > 0) {
    //salida de cada fila
    while($row = $result ->fetch_assoc()) {
        echo "id: " . $row["id"]. " - Nombre: " . $row["nombre"]. "<br>";
    }
}
```

```
}  
} else {  
    echo "0 results";  
}  
$conn->close();  
?>
```

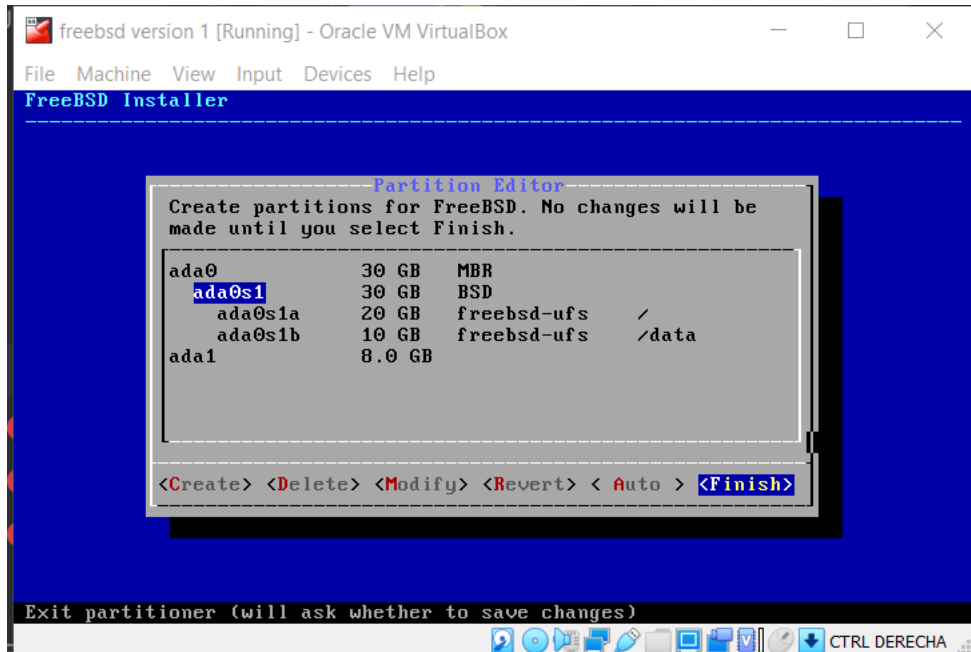
Si introducimos la url 192.168.56.222/index.php nos aparecerá la base de datos



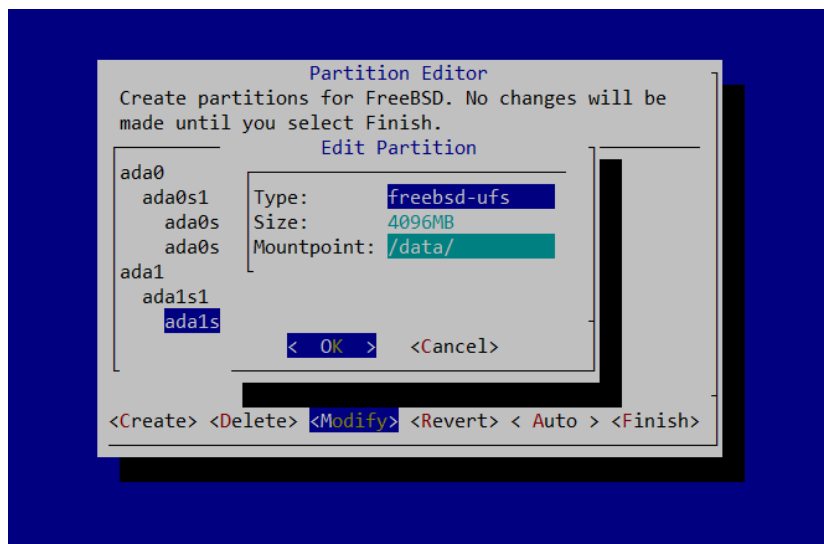
FreeBSD

Particionado

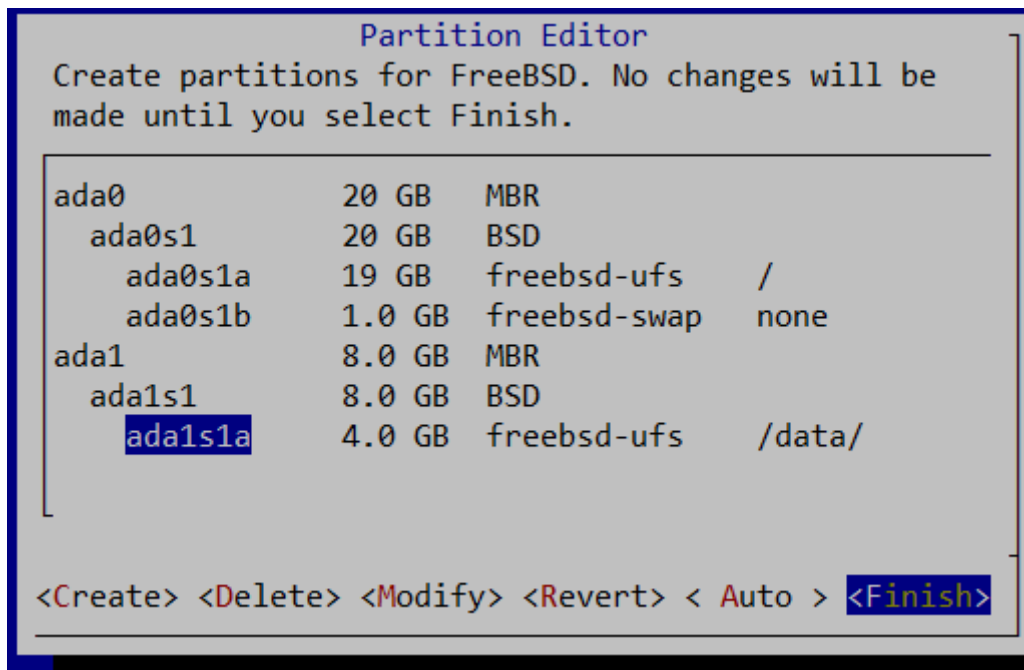
Ejecutamos el comando "sade", y nos aparecerá una ventana, en ella podemos crear particiones en los discos que queramos



Seccionamos el disco ada1 y lo particionamos:



Como resultado tenemos las siguientes particiones:



Comprobamos la partición con cat /etc/fstab

```
root@freebsd:~ # cat /etc/fstab
# Device      Mountpoint      FStype  Options  Dump    Pass#
/dev/ada0s1a  /                ufs     rw       1        1
/dev/ada0s1b  none            swap    sw       0        0
proc /proc        procfs         rw      0        0
/dev/ada1s1a  /data/          ufs     rw       2        2
root@freebsd:~ #
```

Luego tenemos que montarlo en la carpeta data:

```
mkdir /data/
mount -a
```

después ejecutamos df en la carpeta data y nos aparecerá la partición montada

```
root@freebsd:/data # df
Filesystem      1K-blocks    Used   Avail Capacity  Mounted on
/dev/ada0s1a    19278748 10498876 7237576    59%      /
devfs           1          1        0    100%      /dev
procfs          4          4        0    100%      /proc
/dev/ada1s1a    4053308   32836 3696208     1%      /data
root@freebsd:/data #
```

Administración remota

SSH/SFTP/SCP

SSH viene instalado en BSD, solo hay que entrar en el archivo `/etc/rc.conf` y colocar `sshd_enable="YES"`, para configurarlo hay que acceder al fichero `/etc/ssh/sshd_config`, en el aparece toda la configuración de SSH, en el añadimos lo siguiente:

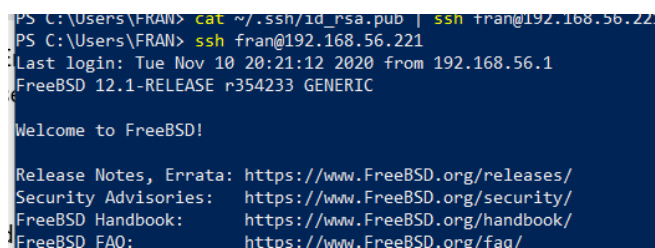
- Añadir: `AllowUsers <usuarios permitidos>`
- Modificar `PubkeyAuthentication` yes
- Modificar: `UsePAM` yes
- Modificar: `PermitRootLogin` no

Después se conecta a BSD colocando en la cmd del host `ssh <usuario>@<ip servidor>`.

Para la clave publica en este caso solo habría que enviarla puesto que ya la creamos para CentOS, la enviaremos esta vez sin el `mkdir` puesto que ya viene creada la carpeta en BSD:

```
cat ~/.ssh/id_rsa.pub | ssh <usuario>@<ip servidor> "cat > ~/.ssh/authorized_keys"
```

y ya nos podríamos conectar:



```
PS C:\Users\FRAN> cat ~/.ssh/id_rsa.pub | ssh fran@192.168.56.221
PS C:\Users\FRAN> ssh fran@192.168.56.221
Last login: Tue Nov 10 20:21:12 2020 from 192.168.56.1
FreeBSD 12.1-RELEASE r354233 GENERIC

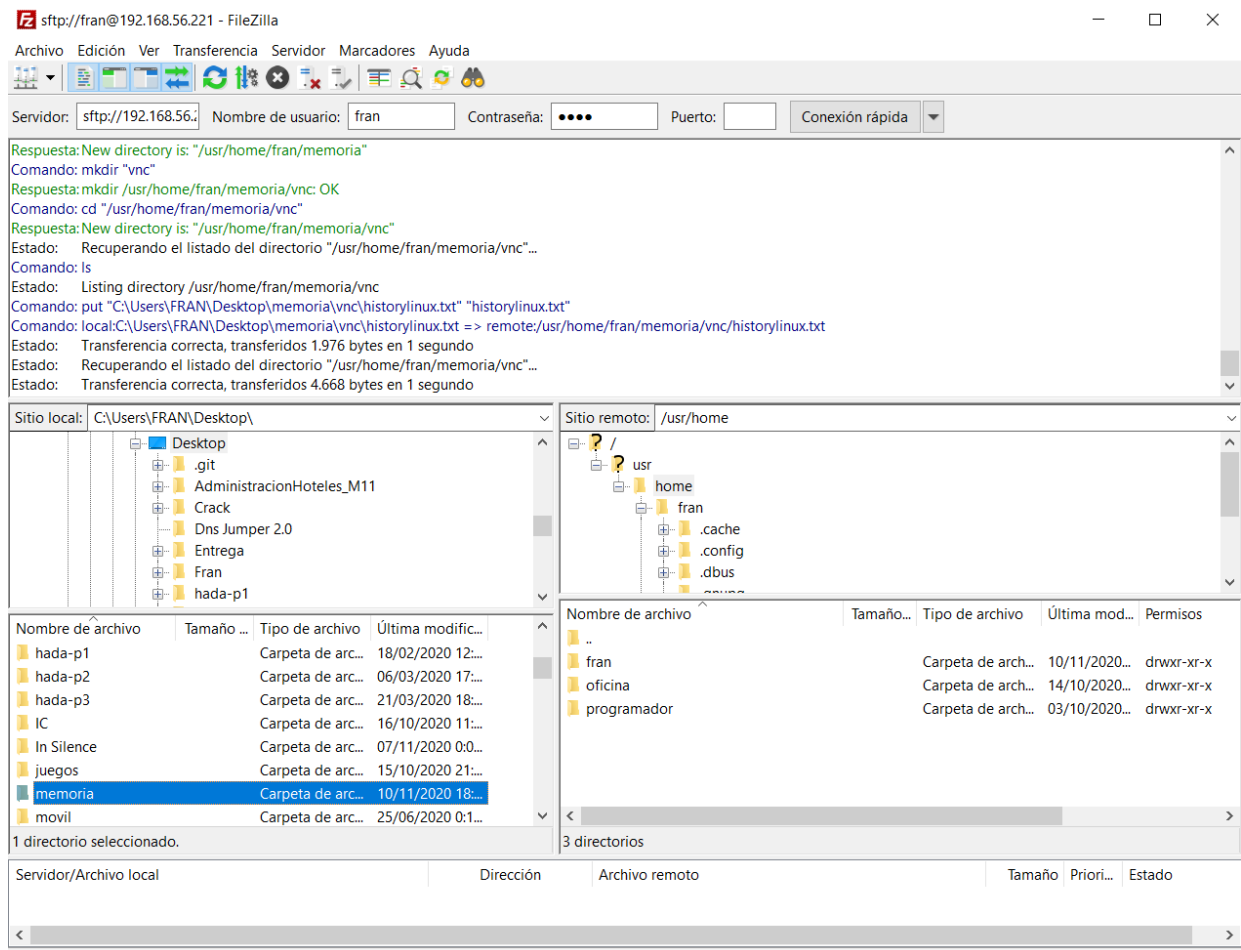
Welcome to FreeBSD!

Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories:  https://www.FreeBSD.org/security/
FreeBSD Handbook:    https://www.FreeBSD.org/handbook/
FreeBSD FAQ:         https://www.FreeBSD.org/faq/
```

Para realizar la conexión SFTP, se ha utilizado el programa de gestión de archivos `filezilla`.

Nos conectamos a nuestro servidor con la ip usuario y contraseña

Y para pasar archivos del host al servidor o viceversa solo hay que pinchar y arrastrar, por ejemplo, he pasado un archivo prueba al servidor



Para comprobar el servicio SCP he enviado un documento de prueba

`scp -p <puerto> <archivo> <usuario>@<ip>:<ruta destino>`

```
C:\Users\FRAN\Desktop>scp -p 22 scpPrueba.txt fran@192.168.56.221:/home/fran
22: No such file or directory
scpPrueba.txt 100% 0 0.0KB/s 00:00
C:\Users\FRAN\Desktop>
```

Networking

DHCP

Para usar dhcp en FreeBSD, hemos de instalar el servicio: `pkg install isc-dhcp44-server`.

Modificamos el archivo `/usr/local/etc/dhcpd.conf` y añadimos lo siguiente:

```

shared-network network.net {
    subnet 192.168.56.0 netmask 255.255.255.0 {
        #option routers 192.168.56.1;
        option subnet-mask 255.255.255.0;
        option broadcast-address 192.168.56.255;
        #option domain-name-servers 192.168.2.100;
        range 192.168.56.101 192.168.56.150;
    }
}
host learn {
    option host-name "nodoA.network.net";
    hardware ethernet 00:25:d3:66:63:b3;
    fixed-address 192.168.2.101;
}

```

Modificamos el archivo `-c /etc/rc.conf` y añadimos `dhcpd_enable="YES"` para habilitar dhcp y `dhcpd_ifaces="em1"` para asignar la interfaz host-only

Activamos el servicio `/usr/local/etc/rc.d/isc-dhcpd start`

```

root@freebsd:~ # /usr/local/etc/rc.d/isc-dhcpd start
Starting dhcpd.
Internet Systems Consortium DHCP Server 4.4.2
Copyright 2004-2020 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
Config file: /usr/local/etc/dhcpd.conf
Database file: /var/db/dhcpd/dhcpd.leases
PID file: /var/run/dhcpd/dhcpd.pid
Wrote 0 class decls to leases file.
Wrote 0 deleted host decls to leases file.
Wrote 0 new dynamic host decls to leases file.
Wrote 0 leases to leases file.
Listening on BPF/em1/08:00:27:74:cc:42/network.net
Sending on BPF/em1/08:00:27:74:cc:42/network.net
Sending on Socket/fallback/fallback-net
root@freebsd:~ #

```

Comprobamos su estado:

```

Sending on Socket/fallback/fallback-net
root@freebsd:~ # service isc-dhcpd status
dhcpd is running as pid 1340.
root@freebsd:~ #

```

He arrancado una maquina debian para comprobar el servicio y vemos que funciona

```

: enp0s8: <BROADCAST,MULTICAST>
group default qlen 1000
    link/ether 08:00:27:74:d0
    inet 192.168.56.102/24 brd
        valid_lft 545sec prefe

```

Git server

Para instalar un servidor git hemos de instalarnos previamente el gestor git:

```
pkg install git
```

A continuación, nos creamos un repositorio en una carpeta:

```
mkdir ~/repo
```

```
cd ~/repo
```

```
git init --bare --shared project1
```

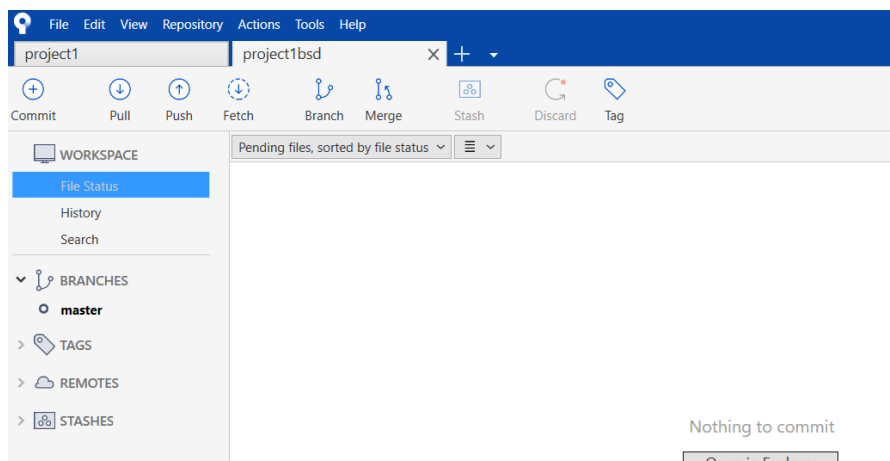
```
$ mkdir ~/repo
mkdir: /home/fran/repo: File exists
$ cd ~/repo
$ ls
$ git init --bare --shared project1
Initialized empty shared Git repository in /usr/home/fran/repo/project1/
$ cd ~/repo/project1/hooks/
$ cp post-update.sample post-update
$
```

A continuación, habilitamos post-update hook de la siguiente manera:

```
cd ~/repo/project1/hooks/
```

```
cp post-update.sample post-update
```

Una vez hecho todo esto, nos descargamos un cliente, en mi caso SourceTree, colocamos la dirección del repositorio en el instalador y ya podríamos trabajar con el:



OwnCloud server

Para implementar el servicio owncloud, primero tenemos que crear una pila FAMP, si seguimos los pasos de [esta web](#) instalaremos php, mariadb, apache y sus respectivas configuraciones

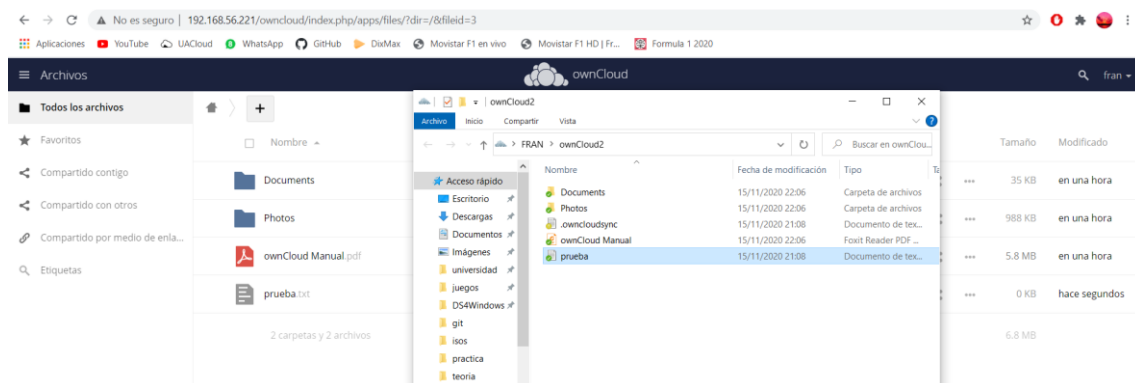
```
root@localhost [(none)]> CREATE DATABASE owncloud_db;
Query OK, 1 row affected (0.000 sec)

root@localhost [(none)]> GRANT ALL ON owncloud_db.* TO 'usuario'@'localhost' IDENTIFIED BY '1234';
Query OK, 0 rows affected (0.001 sec)

root@localhost [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

root@localhost [(none)]> EXIT;
Bye
root@freebsd:~ #
```

Después para configurar owncloud, seguiremos los pasos de [esta web](#) donde instalaremos owncloud lo vincularemos a la base de datos y configuraremos php y apache para activar el servicio



Servicios de impresión

Cups

Configuramos el fichero /etc/cups/cupsd.conf y añadimos:

Port 631

Listen localhost:631

Listen 192.168.56.221:631

Nos instalamos Cups PDF con pkg install cups-pdf

Configuramos el fichero /etc/cups/cups-pdf.conf y añadimos:

Out\${HOME}

Depues entramos en <https://192.168.56.22:631/> y en administración, añadir impresoras y añadimos una impresora genérica de tipo cups-pdf

Virtual_PDF_Printer

Virtual_PDF_Printer (inactiva, aceptando trabajos, no compartida)

Mantenimiento

Administración

Descripción: Virtual PDF Printer

Ubicación: HOME

Controlador: Generic CUPS-PDF Printer (no options) (color)

Conexión: cups-pdf/

Opciones predeterminadas: rótulos={job_sheets_default} papel=desconocido caras=one-sided

Trabajos

Buscar en Virtual_PDF_Printer:

Jobs listed in print order; held jobs appear first.

Bases de datos y web services

Base de datos

Para implementar la base de datos nos instalaremos PostgreSQL:

pkg install -y postgresql12-server

Entramos al gestor con: -u postgres psql

Creamo una base de datos :

```
fran=# \d
          List of relations
 Schema | Name   | Type  | Owner
-----+-----+-----+-----
 public | persona | table | fran
(1 row)

SO fran=# \q
```

Modificamos el archivo: /var/db/postgres/data12/pg_hba.conf:

```
host    all             all             ::1/128         trust
# Allow replication connections from localhost, by a user with the
# replication privilege.
local   replication      all                                     trust
host    replication      all             127.0.0.1/32    trust
host    replication      all             ::1/128         trust
host    all              fran            192.168.56.1/24 trust
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify
```

Después, modificamos /var/db/postgres/data12/postgresql.conf

```
#-----  
# CONNECTIONS AND AUTHENTICATION  
#-----  
  
# - Connection Settings -  
  
listen_addresses = '*'          # what IP address  
                                # comma-separated  
                                # list of addresses  
                                # (change  
port = 5432                     # (change  
max_connections = 100           # (change  
#superuser_reserved_connections = 3 # (change
```

Modificamos /var/db/postgres/data12/postgresql.conf:

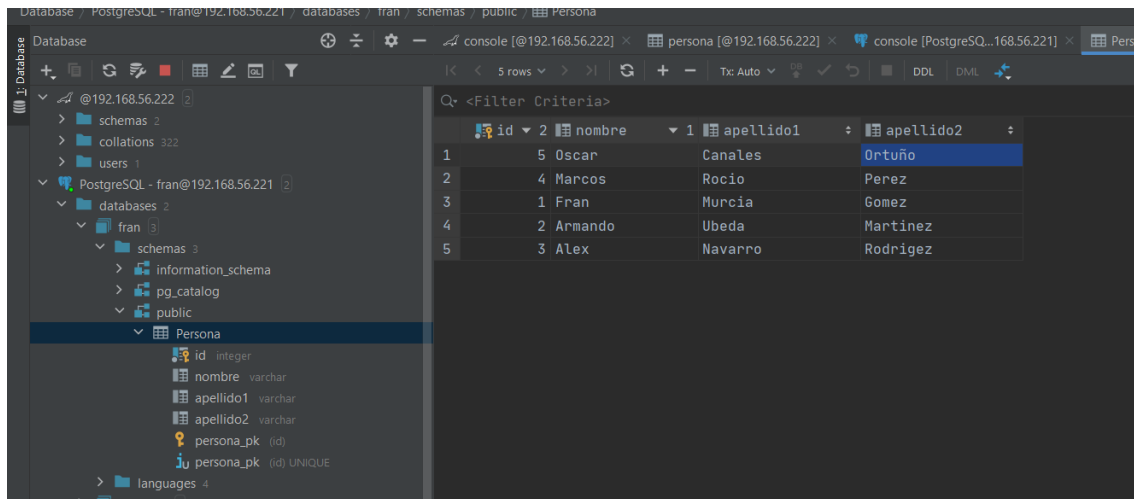
```
<IfModule dir_module>  
    DirectoryIndex index.html index.php  
</IfModule>  
  
#  
# The following lines prevent .htaccess and .h  
# viewed by Web clients.  
#  
<Files ".ht*">  
    Require all denied  
</Files>  
  
<FilesMatch "\.php$">  
    SetHandler application/x-httpd-php  
</FilesMatch>  
<FilesMatch "\.phps$">  
    SetHandler application/x-httpd-php-source  
</FilesMatch>
```

Y finalmente instalamos php:

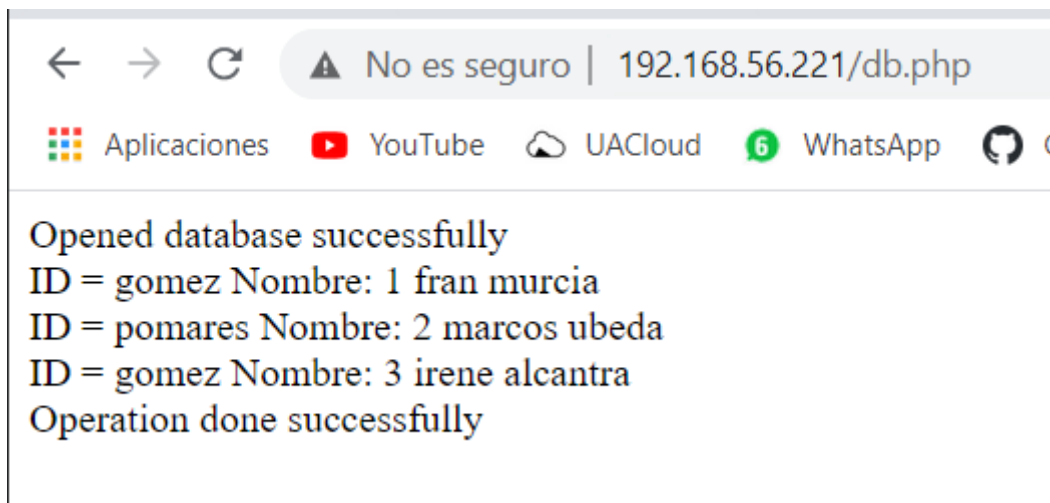
pkg install php74 mod_php74 php74-curl php74-session php74-pgsql

Y creamos una query de php en /usr/local/www/apache24/data

Creamos una tabla en la base de datos con el gestor datagreap



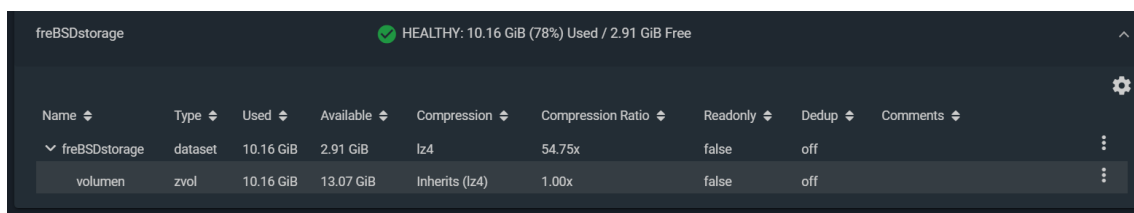
Y finalmente si ponemos en el navegador lo siguiente aparece la bd



File server

FreeNAS

Nos creamos un pool en FreeNAS con las siguientes características:



En la maquina BSD iniciamos el servicio ISCSI y ejecutamos el siguiente comando:

```
iscsictl -A -p<ipFreeNAS> -t <target>
```

```
iscsiectl ~# [~v] [~w timeout]
root@asorc:~ # iscsiectl -A -p 192.168.56.102 -t iqn.2005-10.org.freenas.ctl:target
root@asorc:~ # iscsiectl
Target name                Target portal      State
iqn.2005-10.org.freenas.ctl:target  192.168.56.102    Connected: da0
root@asorc:~ #
```

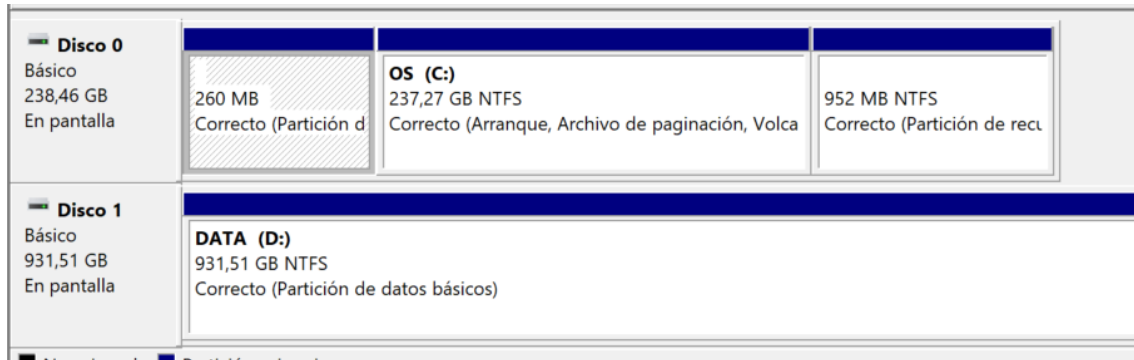
Nos instalamos gpart para montar el disco

```
pkg install gpart
gpart create -s gpt /dev/da0
gpart add -t freebsd-ufs -l 1m /dev/da0
newfs -U /dev/da0p1
mkdir /iscsi_share
mount -t ufs -o rw /dev/da0p1 /iscsi_share
```

Windows

Partición

Con la herramienta Administración de discos le damos clic derecho al disco nuevo y le damos a añadir volumen

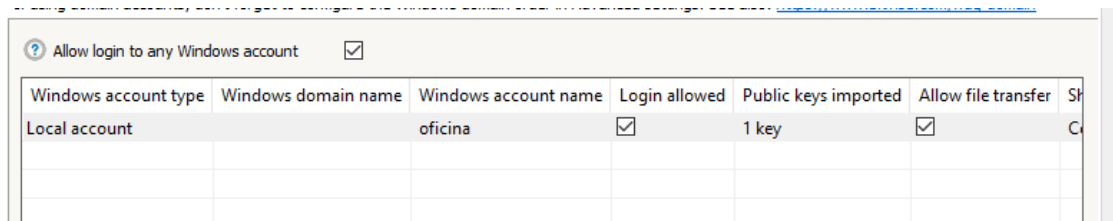


Administración remota

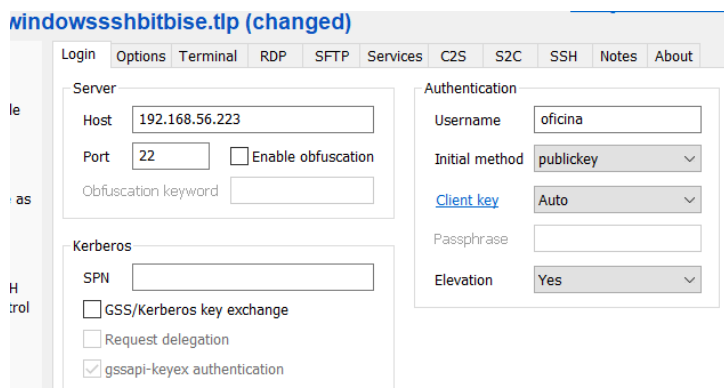
SSH/SFTP/SCP

Nos descargamos el cliente "bitvise SSH server",

En el se damos a "setings" y habilitamos el servicio y el login por clave publica

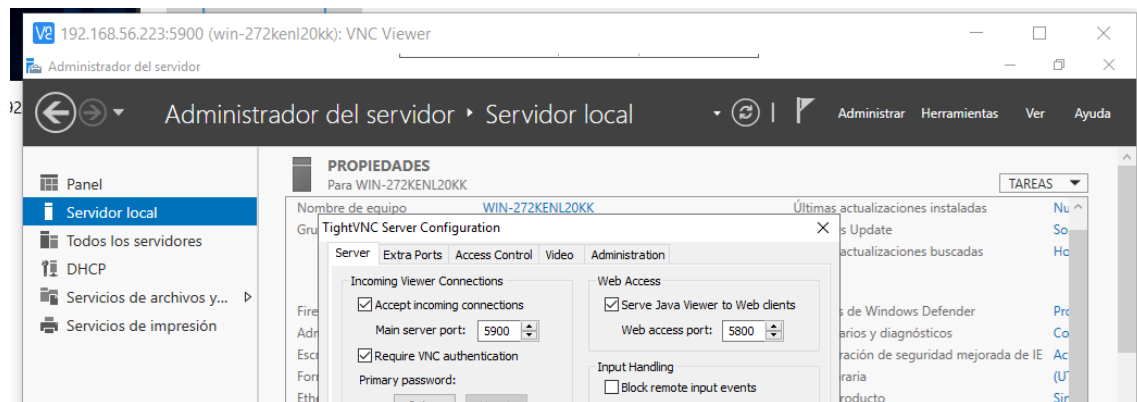


En le host en el "bitvise SSH client",en configuración introducimos la ip del servidor y habilitamos el login por clave publica

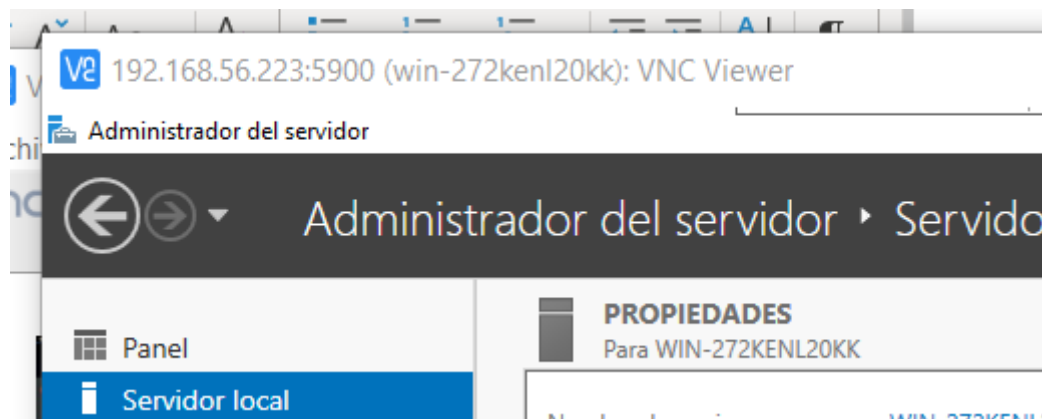


VNC

Para vnc nos descargaremos un cliente, en mi caso tightvnc, iniciamos el servicio y en el host con un cliente nos conectamos a la 192.168.56.223:5900

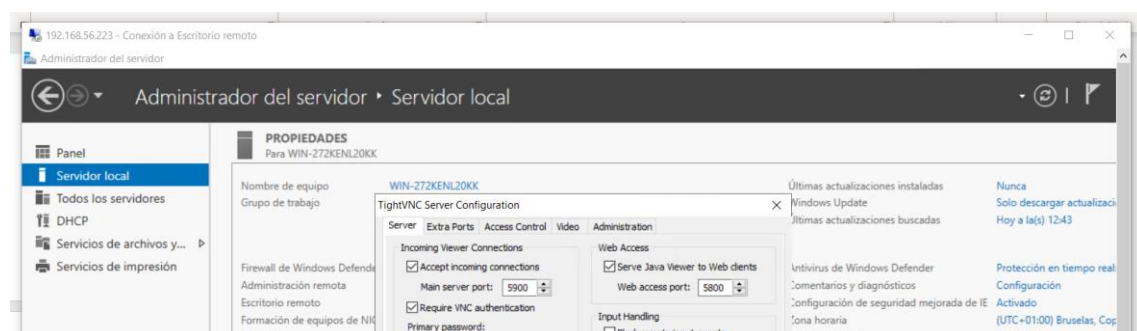


En un cliente introducimos la ipa y el puerto 59000

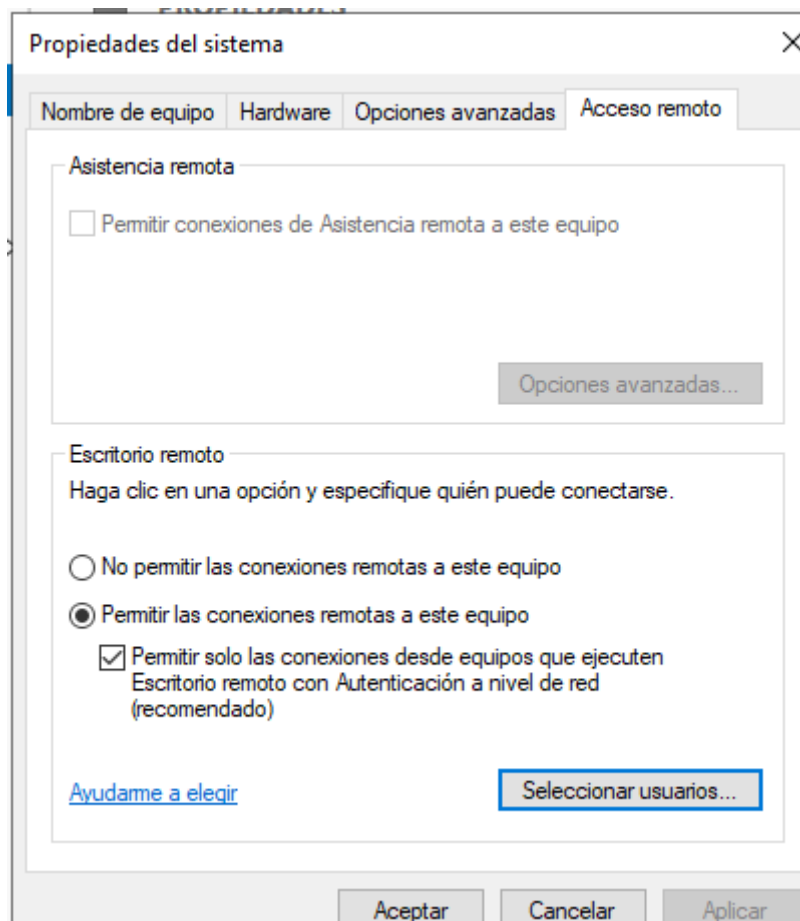


RDP

En administrador del servidor en la sección servidor local habilitamos el escritorio remoto y en escritorio remoto de Windows nos conectamos a la ip de WS



Y permitimos al acceso



Networkig

DHCP

Primero tenemos que colocar nuestra ip a estática en la ruta:

Panel de control\Redes e Internet\Centro de redes y recursos compartidos

General

Puede hacer que la configuración IP se asigne automáticamente si la red es compatible con esta funcionalidad. De lo contrario, deberá consultar con el administrador de red cuál es la configuración IP apropiada.

☐ Obtener una dirección IP automáticamente
☒ Usar la siguiente dirección IP:

Dirección IP:
 Máscara de subred:
 Puerta de enlace predeterminada:

Introducimos la ip 192.168.56.223

Para activar el servicio de dhcp nos iremos al panel del servidor y añadiremos el rol de DHCP, en e programa de DHCP que nos generará, le damos click derecho en ipv4 y en configuración nos aparecerá las opciones del rango de ip

DHCP

Archivo Acción Ver Ayuda

win-272ken20kk

IPv4

Opciones de servidor

Directivas

Filtros

Permitir

Denegar

IPv6

Agregar un ámbito

Un ámbito dinámico

Asistente para ámbito nuevo

Para agregar un ámbito nuevo, seleccione un intervalo de direcciones IP.

Intervalo de direcciones IP

Para definir el intervalo de direcciones del ámbito debe identificar un conjunto de direcciones IP consecutivas.

Opciones de configuración del servidor DHCP

Escriba el intervalo de direcciones que distribuye el ámbito.

Dirección IP inicial: 192 . 168 . 56 . 151

Dirección IP final: 192 . 168 . 56 . 200

Opciones de configuración que se propagan al cliente DHCP

Longitud: 24

Máscara de subred: 255 . 255 . 255 . 0

< Atrás Siguiente > Cancelar

Acciones

IPv4

Acciones adicionales

File server

FreeNAS

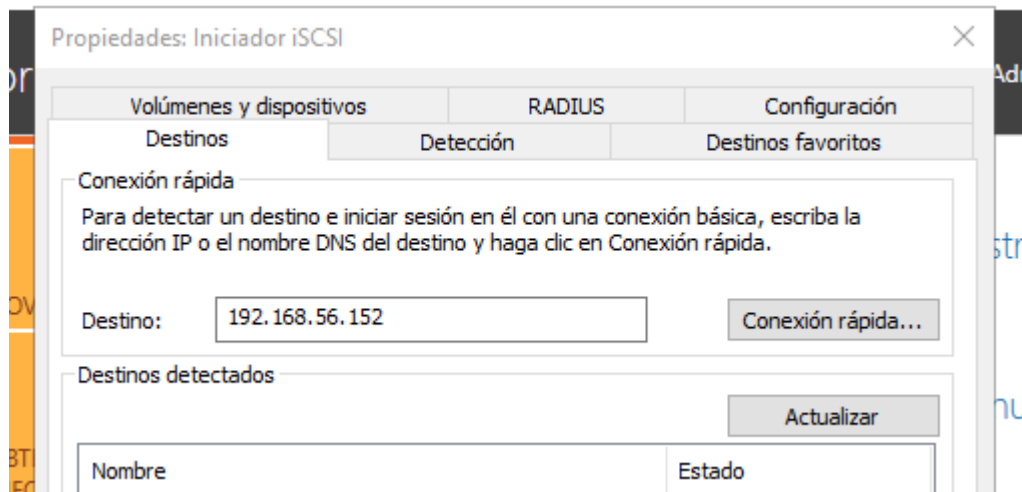
En primer lugar, instalamos una maquina FreeNAS y creamos un pool con las siguientes características:



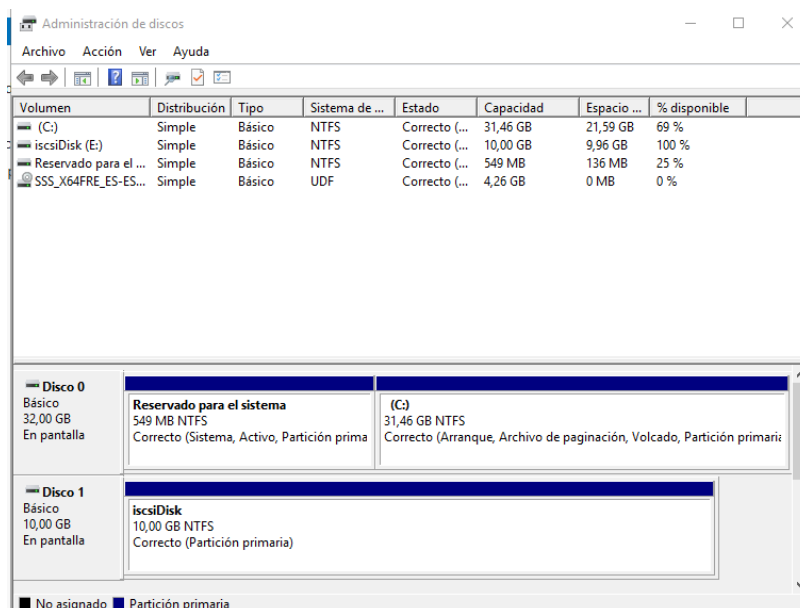
The screenshot shows the 'WindowsStorage' pool configuration in FreeNAS. The status is 'HEALTHY: 10.16 GiB (78%) Used / 2.91 GiB Free'. The table below lists the dataset and volume details.

Name	Type	Used	Available	Compression	Compression Ratio	Readonly	Dedup	Comments
WindowsStorage	dataset	10.16 GiB	2.91 GiB	lz4	77.67x	false	off	
volumen	zvol	10.16 GiB	13.07 GiB	Inherits (lz4)	1.00x	false	off	

Después en la sección del administrador del servidor/administrar/agregar roles y características y en servicios de almacenamiento seleccionamos iSCSI, después, introducimos la ip del FreeNAS

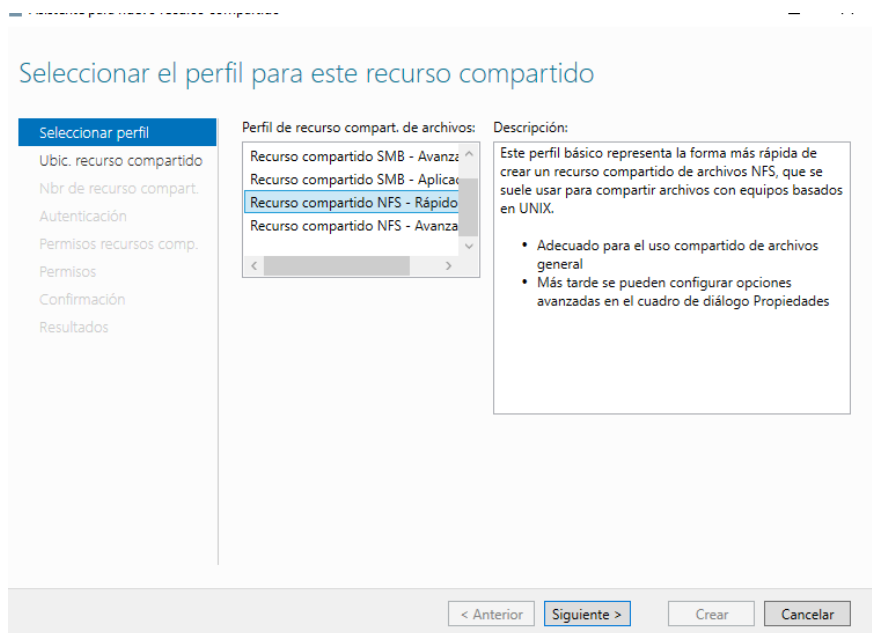


Después añadimos el disco como se hizo en la partición:



NFS/SAMBA

Para hacer nfs y samba nos instalamos en el administrador del servidor administrar/agregar rolles y características servicios de almacenamiento, allí en tareas le damos a añadir recurso compartido y seleccionamos samba o NFS, después indicamos el nombre de la carpeta compartida y el lugar, en nuestro caso en la partición



Después Repetir lo mismo para NFS o samba