

# ADMINISTRACIÓN DE SISTEMAS OPERATIVOS Y REDES DE COMPUTADORES

Contexto de aplicación y desarrollo en  
prácticas.



## ÍNDICE:

<b>1. Explicación de la práctica:</b>	<b>3</b>
<b>2. Particionado</b>	<b>4</b>
2.1. CentOS:	4
2.2. FreeBSD:	5
2.3. Windows Server	5
<b>3. Servicios:</b>	<b>6</b>
2.4. CentOS:	6
• SSH:	6
• VNC:	7
• RDP:	8
• NFS:	8
• SAMBA:	9
• X2go:	9
• FTP:	10
• CUPS:	11
• DHCP:	12
2.5. FreeBSD:	14
• SSH:	14
• VNC:	15
• RDP:	16
• NFS:	16
• SAMBA:	17
• XDMCP:	17
• FTP:	18
• CUPS:	18
• DHCP:	19
2.6. Windows Server 2016	21
• SSH:	21
• VNC:	21
• RDP:	22
• NFS:	22
• SAMBA:	23
• NX:	23
• FTP:	23
• CUPS:	23
• DHCP:	24

## 1. Explicación de la práctica:

En esta segunda práctica se pretende instalar servicios básicos sobre servidores (Windows Server, Centos/Debian, FreeBSD/Solaris).

Los sistemas operativos que voy a utilizar son:

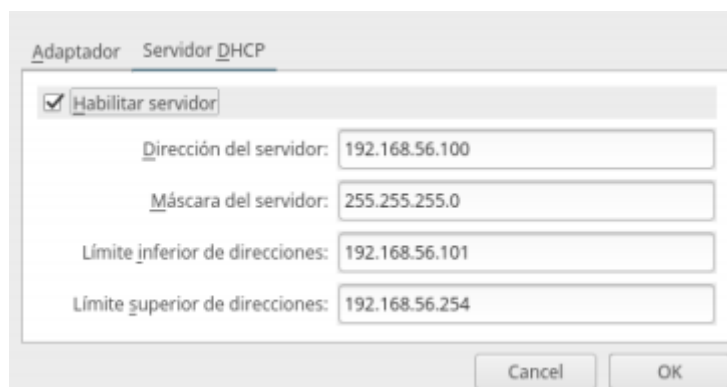
- Windows Server 2016
- CentOS
- FreeBSD

Y los servicios que voy a instalar en los diferentes sistemas operativos son los siguientes:

- SSH (Public key & Port mapping)
- VNC/RDP
- X2GO/XDMCP
- NFD/Samba
- CUPS
- FTP (jail)
- DHCP

Debemos instalar todos los servicios nombrados, crear las particiones necesarias para algún servicio, además de añadir los repositorios necesarios para instalar algunos servicios.

A la hora de utilizar el servicio DHCP de los diferentes sistemas se deberá desactivar el servicio DHCP que tiene VirtualBox en nuestra red solo-anfitrión.



Además de desactivar el firewall en los diferentes Sistemas Operativos y SELinux en los sistemas Linux.

## 2. Particionado

Para crear particiones he creado en cada uno de los sistemas operativos un nuevo disco duro de 20G en la configuración de Virtual Box.

### 2.1. CentOS:

Para crear particiones usando la herramienta fdisk, primero debemos comprobar los discos del sistema disponibles y las particiones que ya haya creadas, para ello utilizamos el parámetro -l:

```
$ sudo fdisk -l

Disco /dev/sda: 160.0 GB, 160041885696 bytes
255 cabezas, 63 sectores/pista, 19457 cilindros
Unidades = cilindros de 16065 * 512 = 8225280 bytes
Identificador de disco: 0x000c3c51

Dispositivo Inicio    Comienzo    Fin        Bloques  Id  Sistema
/dev/sda2             3233        9855       53199247+ 83  Linux
/dev/sda4             9856        19457      77128065  83  Linux
```

Vamos a crear entonces una partición de prueba que utilice el resto de espacio disponible en el disco2, ejecutando fdisk sobre el disco a utilizar:

```
sudo fdisk /dev/sda
```

Después de esto le daremos a la N para empezar a crear una nueva particion en el disco.

Seleccionamos si queremos una partición extendida o primaria, en este caso podemos crearla como primaria, pulsamos “p” y dejamos que automáticamente se configure el número de la partición (se puede especificar, del 1 al 4).

Cuando nos aparezca Último cilindro deberemos poner el tamaño que quieres que tenga la partición, en nuestro caso +4G.

```
Número de partición (1-4): 3
Primer cilindro (1-19457, valor predeterminado 1):
Último cilindro, +cilindros o +tamaño{K,M,G} (1-3232, valor predeterminado 3232):
Se está utilizando el valor predeterminado 3232
```

Necesitaré 3 particiones primary de +4G y una extended que se divide en 2 particiones de +4G.

Después de esto guardaremos los cambios apretando la W y formatearemos las particiones con:

```
mkfs.ext4 /dev/sdb1
```

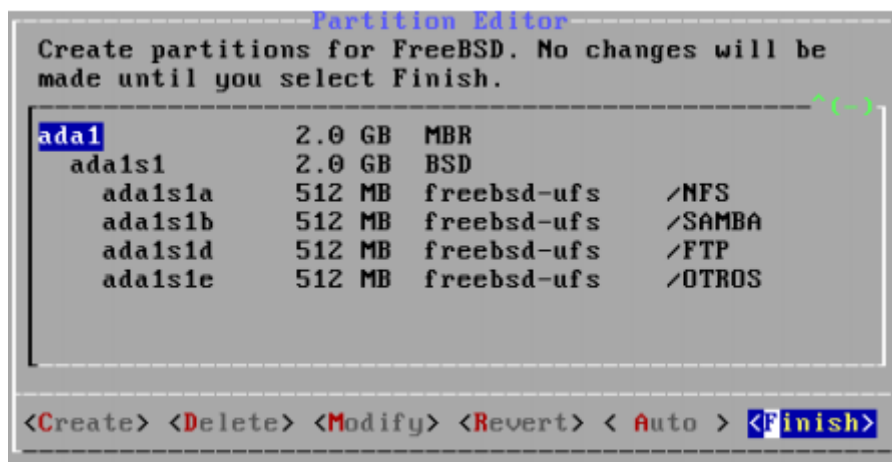
Crearemos una carpeta para montar la partición:

```
mkdir /sdb1
```

```
mount /dev/sdb /sdb
```

## 2.2. [FreeBSD:](#)

Para crear las particiones en FreeBSD se crearán mediante el programa sade:



Y crearemos los diferentes directorios en cada punto de montaje de las nuevas particiones y lo comprobaremos con el fichero /etc/fstab.

## 2.3. [Windows Server](#)

Para crear las particiones en Windows se hace con la opción que tiene Windows: Formatear o crear particiones de Windows.



### 3. Servicios:

Instalé xfce4 para mayor comodidad a la hora de instalar los servicios en CentOS y FreeBSD.

A continuación, explico como he instalado los diferentes servicios pedidos en el Hito 2:

#### 2.4. CentOS:

- **SSH:**

El servicio SSH viene instalado por defecto en CentOS.

Hay que cambiar del archivo `/etc/ssh/sshd_config`

```
Port 22
Protocol 2
AllowUsers pedro
PermitRootLogin no
```

Y si se quiere configurar para una clave pública se deberá cambiar esto:

```
PubkeyAuthentication yes
PasswordAuthentication no
ChallengeResponseAuthentication no
UsePAM yes
```

Y genero las claves en el usuario pedro con `ssh-keygen -t rsa` y hago `mv` para cambiar el nombre a `authorized_key`.

Para probarlo como cliente en Windows se deberá descargar PuTTY y PuTTYGen, cuando tengamos PuTTYGen generaremos las claves pública y privada. Y enviaré la clave pública al servidor y la meteremos en `authorized_key`.

Después de todo esto iniciaremos ssh:

```
Systemctl restart sshd
Ssh pedro@192.168.56.103
```

Luego en Putty pondremos la clave privada, ponemos la ip del servidor, el puerto 22, le damos a open, deberemos poner nuestro usuario y ya estaría.

Para por mapping utilizaremos `ssh -L 8081:www.ua.es:80 pedro@192.168.56.105`.

Entonces en servidor al poner `localhost:8081` nos enviará a `ww.ua.es`, en cliente deberemos poner en SSH/Tunnels para que funcione tras conectarse a la maquina virtual.

- VNC:

Instalaremos vnc: (<https://www.realvnc.com/en/connect/download/viewer/>)

```
pkg install tightvnc
```

Hacemos lo siguiente:

```
cp /lib/systemd/system/vncserver@.service  
/etc/systemd/system/vncserver@:1.service
```

Y cambiamos en `/etc/systemd/system/vncserver@:1.service` los `<USER>` por `pedro`:

```
# ExecStart=/sbin/runuser -l <USER> -c "/usr/bin/vncserver %i"  
# PIDFile=/home/<USER>/.vnc/%H%i.pid  
ExecStart=/sbin/runuser -l pedro -c "/usr/bin/vncserver %i"  
PIDFile=/home/pedro/.vnc/%H%i.pid
```

Nos conectamos como `pedro` e iniciamos `vncserver`:

```
su - pedro  
vncserver
```

Ponemos una contraseña e iniciamos los servicios:

```
su -  
systemctl daemon-reload  
systemctl enable vncserver@:1.service  
systemctl start vncserver@:1.service
```

Nos descargamos en Windows cliente VNC Viewer:

(<https://www.realvnc.com/en/connect/download/viewer/>)

Y en el programa ponemos `192.168.56.203:5901` y ya funcionaría.

- **RDP:**

Instalamos xrdp:

```
Yum install xrdp -enablerepo=cr
```

Lo iniciamos:

```
systemctl start xrdp
```

En Windows cliente vamos a Conexión a Escritorio Remoto ponemos la ip y usuario y ya estaría.

- **NFS:**

Utilizaremos la partición sdb1.

Instalamos nfs-server

```
Yum install nfs-server
```

Y a continuación lo siguiente:

```
Chown nfsnobody:nfsnobody /sdb1
```

```
Chmod -R 755 /sdb1
```

Modificamos el archivo /etc/exports y ponemos lo siguiente:

```
/sdb1 *(rw,sync,no_root_squash,no_all_squash)
```

E iniciamos el servicio:

```
Systemctl start nfs-server.service
```

En Windows será necesario habilitar NFS por la configuración del Sistema y después de eso en cmd poner:

```
Mount 192.168.56.103:/sdb1 Z:
```

Para desmontarlo:

```
Umount 192.168.56.103:/sdb1 Z:
```



- **SAMBA:**

Instalamos samba:

```
yum -y install samba samba-client samba-common
```

Y configuramos el archivo `/etc/samba/smb.conf`

```
[global]
workgroup = WORKGROUP
server string = samba
security = user
map to guest = bad user
```

```
[Publico]
Path = /sdb2
browseable = yes
writeable = yes
guest ok = yes
```

Después esto:

```
chmod -R 777 /sdb2/
```

Y lo iniciamos:

```
systemctl start nmb
systemctl start smb
```

En Windows como cliente abrir la aplicación de escritorio Ejecutar y poner:

```
\\192.168.56.103
```

- **X2go:**

Instalamos x2goserver:

```
yum -y install x2goserver
```

Ponemos lo siguiente y lo iniciamos:

```
x2godbadmin -created
systemctl status x2gocleansessions.service
```

Como cliente utilizaremos el programa X2GO, seleccionamos la clave privada del ssh para que no de problemas.

Ponemos la ip del servidor y nos logeamos y ya estaría.

- FTP:

He seguido los pasos de aquí (<http://creativapixel.com/blog/index.php?post/Crear-cuentas-ftp-con-acceso-a-ciertas-carpetas-en-CentOS-7>)

Primero instalamos el servidor ftp:

```
yum install vsftpd
```

Luego configuramos el servidor ftp ingresando a:

```
nano /etc/vsftpd/vsftpd.conf
```

Y modificamos los siguientes parametros:

```
listen = YES
anonymous_enable = NO
local_enable = YES
write_enable = YES
local_umask = 022
chroot_local_user = YES
chroot_list_enable = YES
chroot_list_file = /etc/vsftpd.chroot_list
```

Comentamos la línea:

```
# Listen_ipv6 = YES
```

Grabamos y salimos. Adicionalmente agregamos una línea más a nuestro archivo de configuración:

```
echo 'allow_writeable_chroot=YES' >> /etc/vsftpd/vsftpd.conf
```

Luego creamos un grupo

```
groupadd ftp
```

Creamos un shell fantasma

```
mkdir /bin/ftp
```

Editamos el shell del sistema

```
nano /etc/shells
```

y agregamos en la última línea

```
/bin/ftp
```

Grabamos y salimos. Ahora crearemos un usuario:

```
sudo useradd -g ftp -d /ruta_carpeta_destino -c "Nombre del Usuario" nombre_usuario
```

Le asignamos una clave

```
sudo passwd usuarioftp
```

Buscamos a nuestro usuario recién creado en:

```
nano /etc/passwd
```

Copiamos (CTRL+C) la línea de nuestro usuario parecida a:

```
usuario:x:1004:118:Nombre del  
Usuario:/ruta_carpeta_destino:/bin/ftp
```

Y lo pegamos con un clic en:

```
nano /etc/vsftpd/chroot_list
```

NOTA: Si chroot\_list no existe debemos crearlo

Finalmente reiniciamos y listo:

```
systemctl restart vsftpd.service
```

Para probarlo en Windows como cliente deberemos descargar FileZilla, iniciarlo, poner la ip y logearnos.

- **CUPS:**

Instalamos cups:

```
Yum install cups cups-pdf
```

Editamos la configuración desde /etc/cups/cupsd.conf y agregamos las redes que queramos que tengan acceso a la interfaz web.

Iniciamos el servicio:

```
Systemctl enable
```

Y desde un navegador web accedemos a la página de administración de impresoras CUPS y agregamos la impresora PDF.

The screenshot shows the CUPS web interface for adding a printer. It has three sections: 'Impresoras locales:' with a radio button selected for 'CUPS-PDF (Virtual PDF Printer)'; 'Impresoras en red descubiertas:' which is empty; and 'Otras impresoras en red:' with a list of radio buttons for various protocols: 'Protocolo de Impresión de Internet IPP (https)', 'Protocolo de Impresión de Internet IPP (ipp)', 'Protocolo de Impresión de Internet IPP (ipps)', 'Equipo o impresora LPD/LPR', 'AppSocket/HP JetDirect', and 'Protocolo de Impresión de Internet IPP (http)'. At the bottom of this list is a 'Siguiente' (Next) button.

**Nombre:** Virtual\_PDF\_Printer  
(Puede contener cualquier carácter imprimible excepto "/", "#", y espacio)

**Descripción:** Virtual PDF Printer  
(Descripción fácilmente leible tal como "HP LaserJet de doble cara")

**Ubicación:** VirtualBox ASORC  
(Ubicación fácilmente leible tal como "Lab 1")

**Conexión:** cups-pdf:/

**Compartición:** ☒ Compartir esta impresora

**Nombre:** Virtual\_PDF\_Printer  
**Descripción:** Virtual PDF Printer  
**Ubicación:** VirtualBox ASORC  
**Conexión:** cups-pdf:/  
**Compartición:** compartir esta impresora

**Marca:** Generic

**Modelo:** Generic CUPS-PDF Printer (en) ^  
Generic PCL Laser Printer (en)  
Generic PostScript Printer (en) v

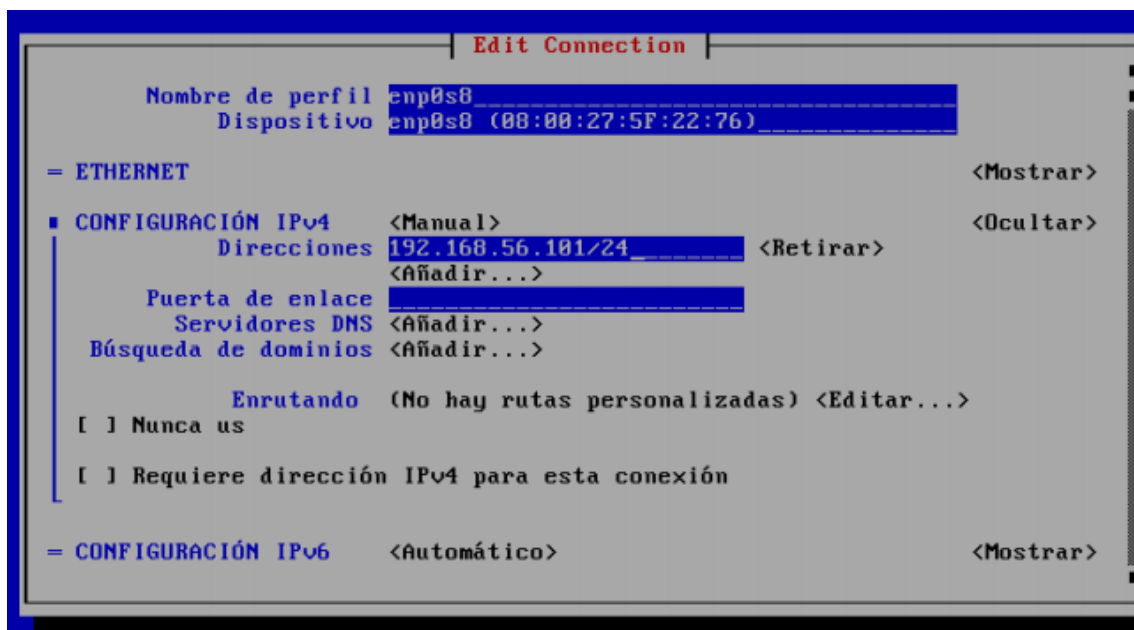
**O proporcione un archivo PPD:**  Ningún archivo seleccionado

Para ver si va desde un cliente agregamos la impresora en Windows en la opción del panel de control, Dispositivos e impresoras y ponemos:

<http://192.168.103:631/>

- DHCP:

Tendremos que poner la dirección IP estática, podemos hacerlo con la herramienta nmtui:



### Instalaremos el paquete DHCP

```
Yum install dhcp  
Y configuramos /etc/dhcp/dhcpd.conf:  
subnet 192.168.56.0 netmask 255.255.255.0 {  
  range 192.168.56.130 192.168.56.140;  
  option domainnameservers 8.8.8.8;  
  option routers 192.168.56.100;  
  option broadcastaddress 192.168.56.255;  
  defaultlease time 600;  
  maxlease time 7200;  
}
```

### E iniciamos el servicio:

```
Systemctl start dhcpd
```

Podemos ver si el servidor está en funcionamiento con el programa dhcptest.

## 2.5. FreeBSD:

- SSH:

El servicio SSH viene instalado por defecto en FreeBSD.

Hay que cambiar del archivo `/etc/ssh/sshd_config`

```
Port 22
Protocol 2
AllowUsers Pedro
PermitRootLogin no
```

Y si se quiere configurar para una clave pública se deberá cambiar esto:

```
PubkeyAuthentication yes
PasswordAuthentication no
ChallengeResponseAuthentication no
UsePAM yes
```

Y genero las claves en el usuario pedro con `ssh-keygen -t rsa` y hago `mv` para cambiar el nombre a `authorized_key`.

Para probarlo como cliente en Windows se deberá descargar PuTTY y PuTTYGen, cuando tengamos PuTTYGen generaremos las claves pública y privada. Y enviaré la clave pública al servidor y la meteremos en `authorized_key`.

Después de todo esto iniciaremos ssh:

```
Service sshd restart
Ssh Pedro@192.168.56.105
```

Luego en Putty pondremos la clave privada, ponemos la ip del servidor, el puerto 22, le damos a open, deberemos poner nuestro usuario y ya estaría.

Para por mapping utilizaremos `ssh -L 8081:www.ua.es:80 Pedro@192.168.56.105`.

Entonces en servidor al poner `localhost:8081` nos enviará a `ww.ua.es`, en cliente deberemos poner en SSH/Tunnels para que funcione tras conectarse a la maquina virtual.

- VNC:

Instalaremos vnc: (<https://www.realvnc.com/en/connect/download/viewer/>)

```
pkg install tightvnc
```

Hacemos lo siguiente:

```
cp /lib/systemd/system/vncserver@.service  
/etc/systemd/system/vncserver@:1.service
```

Y cambiamos en `/etc/systemd/system/vncserver@:1.service` los `<USER>` por pedro:

```
# ExecStart=/sbin/runuser -l <USER> -c "/usr/bin/vncserver %i"  
# PIDFile=/home/<USER>/.vnc/%H%i.pid  
ExecStart=/sbin/runuser -l Pedro -c "/usr/bin/vncserver %i"  
PIDFile=/home/Pedro/.vnc/%H%i.pid
```

Nos conectamos como pedro e iniciamos vncserver:

```
su - Pedro  
vncserver
```

Ponemos una contraseña e iniciamos los servicios:

```
su -  
systemctl enable vncserver@:1.service  
systemctl start vncserver@:1.service
```

Nos descargamos en Windows cliente VNC Viewer:

(<https://www.realvnc.com/en/connect/download/viewer/>)

Y en el programa ponemos 192.168.56.105:5901 y ya funcionaría.

- **RDP:**

Instalamos el paquete xrdp:

```
pkg install xrdp
```

Después de la instalación, nos pedirá añadir dos entradas al fichero /etc/rc.conf para poder iniciar el servicio automáticamente al inicio:

```
xrdp_enable="YES"
xrdp_sesman_enable="YES" 9
```

Ahora iniciamos el servicio:

```
service xrdp start
```

En Windows cliente vamos a Conexión a Escritorio Remoto ponemos la ip y usuario y ya estaría.

- **NFS:**

Utilizaremos la partición ada1.

Instalamos nfs-server:

```
Pkg install nfs-server
```

Y modificamos /etc/exports:

```
/ada1 192.168.56.1
```

Además de modificar /etc/rc.conf:

```
rpcbind_enable="YES"
nfs_server_enable="YES"
mountd_flags="-r"
```

E iniciamos el servicio:

```
service nfs-server.service start
```

En Windows será necesario habilitar NFS por la configuración del Sistema y después de eso en cmd poner:

```
Mount 192.168.56.105:/ada1 Z:
```

Para desmontarlo:

```
Umount 192.168.56.105:/ada1 Z:
```



- **SAMBA:**

Instalamos Samba:

```
Pkg install samba48
```

Añadimos el servicio al archivo /etc/rc.conf:

```
samba_enable="YES"
```

Modificamos el archivo /etc/smb4.conf:

```
[compartida]
comment = Compartida
path = /ada2
valid users = Pedro
public = no
writable = yes
printable = no
```

Añadimos el nuevo usuario a la base de datos que utiliza Samba con el siguiente comando:

```
pdbedit -a -u Pedro
```

Iniciamos el servicio:

```
Service samba restart
```

En Windows como cliente abrir la aplicación de escritorio Ejecutar y poner:

```
\\192.168.56.105
```

- **XDMCP:**

Instalación del cliente XMING en windows que permite abrir ventanas remotas de x11.

Instalar xdm:

```
Pkg install xdm
```

Para establecer la conexión con el escritorio remoto, iniciamos el cliente ssh "putty". Antes de conectar vía ssh, debemos habilitar la opción "Enable X11 forwarding".

Una vez establecida la sesión SSH, podemos abrir cualquier aplicación indicando que se debe ejecutar en el display manager actual.

Por ejemplo abrimos nautilus con el comando:

```
Nautilus
```

Con lo que se abrirá el administrador de archivos nautilus, pero en lugar de mostrarse en el servidor, se mostrará por medio del gestor de ventanas XMING.

- FTP:

Utilizaremos la partición ada3.

Instalamos nfs-server:

Pkg install ftp

Modificamos /etc/ftpchroot:

@ /ada3

Y modificamos /etc/rc.conf:

ftpd\_enable="YES"

Finalmente reiniciamos y listo:

```
service vsftpd.service restart
```

Para probarlo en Windows como cliente deberemos descargar FileZilla, iniciarlo, poner la ip y logearnos.

- CUPS:

Instalamos cups:

```
pkg install cups cups-pdf
```

Editamos la configuración desde /etc/cups/cupsd.conf y agregamos las redes que queramos que tengan acceso a la interfaz web.

Ademas de poner:

```
cupscctl --remote-admin
```

Iniciamos el servicio:

```
Service cupsd restart
```

Y desde un navegador web accedemos a la página de administración de impresoras CUPS y agregamos la impresora PDF.



**Nombre:** Virtual\_PDF\_Printer  
(Puede contener cualquier carácter imprimible excepto "/", "#", y espacio)

**Descripción:** Virtual PDF Printer  
(Descripción fácilmente leible tal como "HP LaserJet de doble cara")

**Ubicación:** VirtualBox ASORC  
(Ubicación fácilmente leible tal como "Lab 1")

**Conexión:** cups-pdf/

**Compartición:** ☒ Compartir esta impresora

**Nombre:** Virtual\_PDF\_Printer  
**Descripción:** Virtual PDF Printer  
**Ubicación:** VirtualBox ASORC  
**Conexión:** cups-pdf/  
**Compartición:** compartir esta impresora

**Marca:** Generic

**Modelo:** Generic CUPS-PDF Printer (en) ^  
Generic PCL Laser Printer (en)  
Generic PostScript Printer (en) v

**O proporcione un archivo PPD:**  Ningún archivo seleccionado

Para ver si va desde un cliente agregamos la impresora en Windows en la opción del panel de control, Dispositivos e impresoras y ponemos:

`http://192.168.103:631/`

- **DHCP:**

Configuramos una dirección IP estática a nuestra interfaz de red solo-anfitrión desde `/etc/rc.conf`:

```
ifconfig_em1="inet 192.168.56.105 netmask 255.255.255.0"
```

Instalaremos el paquete DHCP:

```
Pkg install isc-dhcp44-server
```

Añadimos dos entradas para el servicio en `/etc/rc.conf`:

```
dhcp_enable="YES"
dhcp_ifaces="em1"
```

Y configuramos /etc/dhcp/dhcpd.conf:

```
subnet 192.168.56.0 netmask 255.255.255.0 {  
    range 192.168.56.130 192.168.56.140;  
    option domainnameservers 8.8.8.8;  
    option routers 192.168.56.100;  
    option broadcastaddress 192.168.56.255;  
    defaultleasetime 600;  
    maxleasetime 7200;  
}
```

E iniciamos el servicio:

```
Service dhcpd restart
```

Podemos ver si el servidor está en funcionamiento con el programa dhcptest.

## 2.6. Windows Server 2016

- SSH:

Para ssh instalaré en el servidor Bitvise SSH Server y pondremos la clave pública generada del cliente gracias a PuTTYGen (Estos pasos son los mismos que en CentOS y FreeBSD).

Para ello debemos ir a:

Edit advanced settings -> Windows accounts -> adds -> Authentication -> Public keys

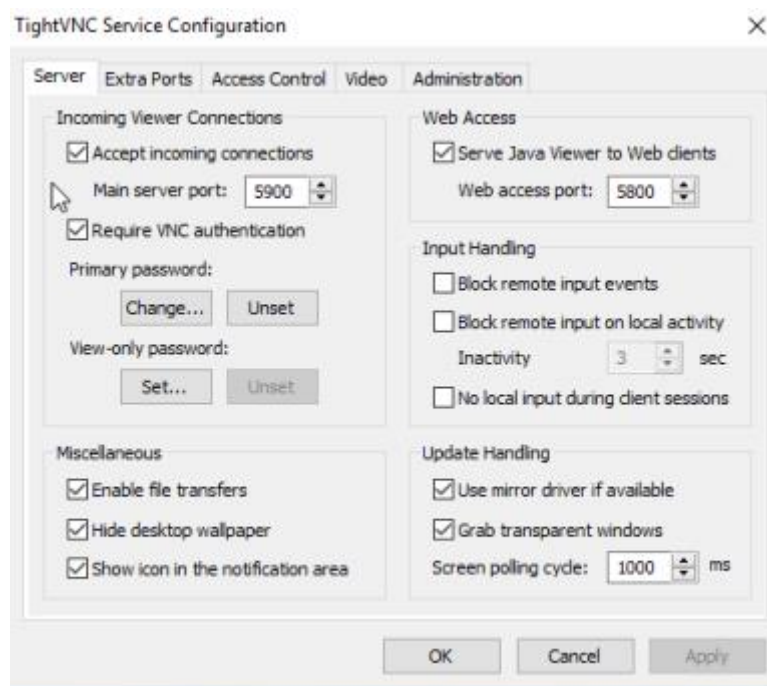
Añadimos la clave pública y le metemos el perfil de Administrador.

Después de esto le cambiamos los puertos a otro diferente a 22 e iniciamos el server ssh.

Para utilizarlo en el cliente, igual que en CentOS y FreeBSD.

- VNC:

Instalamos el programa TightVNC. Una vez instalado, accedemos a la interfaz gráfica del programa, cambiamos el puerto por 5901 y establecemos una contraseña.

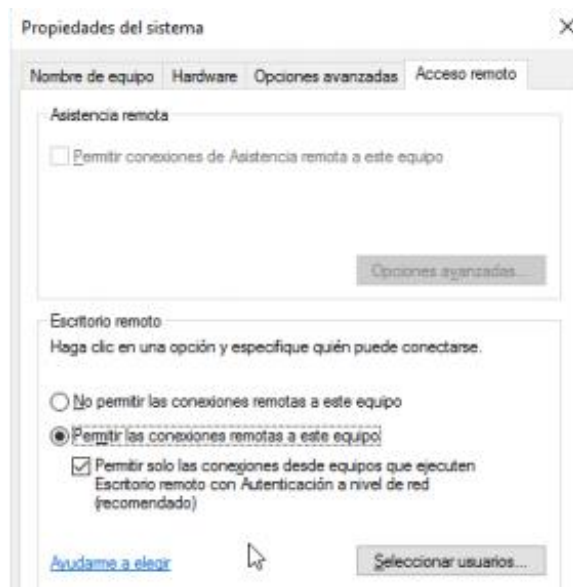


Probamos que funcione correctamente en Windows cliente con VNCViewer poniendo:

192.168.56.101:5901

- RDP:

Accedemos a Panel de Control -> Sistema y Seguridad -> Sistema -> Configuración avanzada del sistema -> Pestaña acceso remoto -> Activamos Permitir las conexiones remotas a este equipo:

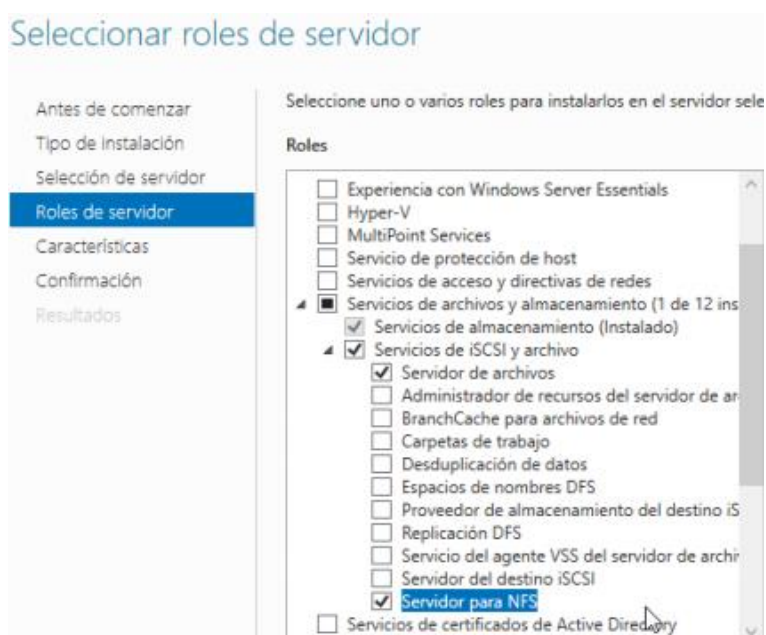


En Windows cliente vamos a Conexión a Escritorio Remoto ponemos la ip y usuario y ya estaría.

- NFS:

Vamos a Administrador del servidor y a Agregar roles y características.

Seguimos los pasos hasta llegar a la selección de roles y seleccionamos Servicios de NFS y le damos a siguiente hasta terminar e instalar.



Vamos a servicios de archivos y almacenamiento -> Recursos compartidos -> TAREAS -> Nuevo recurso compartido.

Seleccionamos “Recurso compartido NFS – Rápido”, seleccionamos la partición que queramos, y agregamos permisos de escritura y lectura.

En Windows cliente será necesario habilitar NFS por la configuración del Sistema y después de eso en cmd poner:

```
Mount 192.168.56.101:/share Z:
```

Para desmontarlo:

```
Umount 192.168.56.101:/share Z:
```

- **SAMBA:**

Se hace exactamente igual que NFS pero se debe cambiar al agregar un nuevo rol la opción de “Recurso compartido NFS – Rápido” a “Recurso compartido SMB – Rápido”

- **NX:**

Instalamos el programa NoMachine en el cliente y en el servidor

Y nos conectamos con NoMachine del cliente Windows al servidor.

- **FTP:**

Instalamos Serv-U y lo configuramos para que el usuario no pueda salir del directorio /home.

Para probarlo en Windows como cliente deberemos descargar FileZilla, iniciarlo, poner la ip y logearnos.

- **CUPS:**

Instalamos PDF Creator y desde panel de control damos click a propiedades de la impresora. Desde la pestaña Compartir activamos el uso compartido asegurándonos de que en el Centro de redes y recursos compartidos también tengamos activados todos los permisos.

Luego, desde un cliente, nos conectamos, agregando la impresora en Windows en la opción del panel de control, Dispositivos e impresoras y ponemos:

```
\\192.168.101\PDFCreator
```

- DHCP:

Accedemos a Administrador del servidor → Agregar roles y características → Instalación basada en características y roles → Seleccionar un servidor del grupo de servidores (habiendo seleccionado el nuestro propio) → Seleccionamos el checkbox de Servidor DNS y seguimos las instrucciones de instalación.

Desde el Administrador del servidor, nos vamos a Herramientas → DHCP. Hacemos click derecho a IPv4 -> Ámbito nuevo y seguimos los pasos y seleccionamos el rango adecuado para las IPs y finalizamos.

Podemos ver si el servidor está en funcionamiento con el programa dhcptest.