

Sistema de archivos en red

Acosta Arias Josué Isaac

AA01132424

Vega Yanes Miguel Alejandro

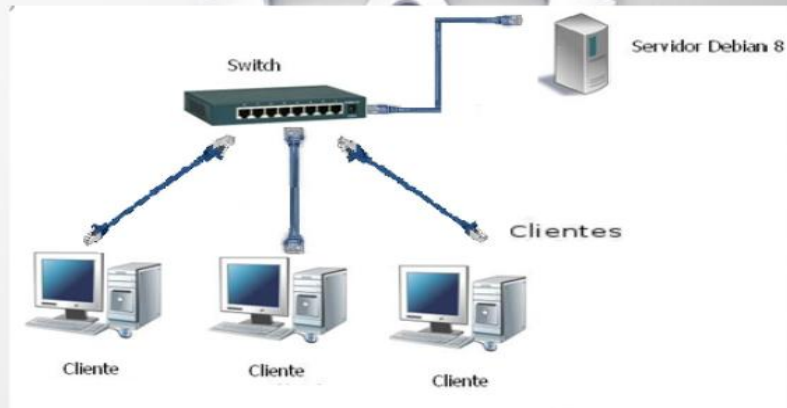
VY01132301

Villafranco Nahúm de Jesús

V01121164

Bonilla Henríquez Alexi waldir

BH01132333



¿Que es samba?

Es una aplicación servidor de implementación de código abierto del protocolo Server Message Block (SMB), que permite la interconexión de redes Microsoft Windows, Linux, UNIX y otros Sistemas Operativos juntos, permitiendo el acceso a archivos basados en Windows y compartir impresoras.

Lo que Samba puede hacer:



- ✓ Sirve árboles de directorios e impresoras a clientes Linux, UNIX y Windows ✓ Asiste en la navegación de la red (con o sin NetBIOS).
- ✓ Autentifica las conexiones a dominios Windows.
- ✓ Proporciona resolución de nombres de Windows Internet Name Service (WINS).
- ✓ Actúa como un Controlador de Dominio Primario (Primary Domain Controller, PDC) estilo Windows.
- ✓ Actúa como un Backup Domain Controller (BDC) para un PDC basado en Samba.
- ✓ Actúa como un miembro servidor de dominio de Active Directory.
- ✓ Une un Windows NT/2000/2003 PDC.

Lo que Samba no puede hacer:



- ✓ Actúa como un Backup Domain Controller (BDC) para un Windows PDC (y viceversa)
- ✓ Actúa como un controlador de dominio de Active Directory

COMPARTIR ARCHIVOS EN RED CON SAMBA

Paso 1 ✓ Como usuario root instalamos el servidor samba usando `apt-get install samba`.

```
root@debian:/home/nahum# apt-get install samba
```

Paso 2

✓ Procedemos a crear los usuarios, asignando nombre y número de identificación por usuarios de la siguiente manera.
`#adduser -system -no-create-home -uid 603 lacrimosa`

Este comando se repite de acuerdo al número de usuarios que queremos crear, con la variante que el número (uid incrementaría de acuerdo al numero de usuarios que dese crear).

```
root@debian:/home/nahum# adduser -system -no-create-home -uid 603 lacrimosa
```

Paso 3

✓ Añadimos los usuarios a samba de la siguiente manera.

```
# smbpasswd -a lacrimosa
```

Con este comando le asignaremos el password a los usuarios creados, nos pedirá que ingresemos un password y luego la confirmación del mismo, esto lo haríamos hasta terminar con todos los usuarios a crear.

```
root@debian:/home/nahum# smbpasswd -a lacrimosa
New SMB password:
Retype new SMB password:
Added user lacrimosa.
root@debian:/home/nahum# █
```


Paso 4

- ✓ Procedemos a la configuración del archivo de samba.

nano /etc/samba/smb.conf

(Procedemos a poner el nombre del grupo al que pertenece la PC para eso buscamos la línea Workgroup en la sección Global).

```
##### Global Settings #####
[global]
## Browsing/Identification ##
# Change this to the workgroup/NT-domain name your Samba server will part of
workgroup = FileSystem
```

(Habilitamos la autenticación del usuario des comentando la línea security de la sección Authentication).

```
##### Authentication #####
# Server role. Defines in which mode Samba will operate. Possible
# values are "standalone server", "member server", "classic primary
# domain controller", "classic backup domain controller", "active
# directory domain controller".
security = user
# Most people will want "standalone sever" or "member server".
# Running as "active directory domain controller" will require first
# running "samba-tool domain provision" to wipe databases and create a
# new domain.
```

(Procedemos a configurar el archivo con el recurso compartido).

Nos ubicamos en las últimas líneas para poder agregar nuestra configuración

Esta es la configuración que ingresaremos en nuestro archivo de samba

[Publik]----->Nombre del recurso (como lo deban bs clientes)
path=/home/nahum/publik-smb----->Ruta del directorio a compartir
read only=no----->Permiso de lectura en general
security=user----->
null password=yes----->
case sensitive=no----->
comment=Carpeta Compartida----->Comentarios
guest ok=yes----->
browseable=yes----->Visibilidad del recurso en la red
umask=000----->Permisos de escritura o lectura para archivos

```
[Publik]
path = /home/nahum/publik-smb
read only = no
security = user
null password = yes
case sensitive = no
comment = Carpeta Compartida
guest ok = yes
browseable = yes
umask = 000
```

Después de haber configurado los recursos compartidos Guardamos el archivo y lo cerramos (ctrl+o, enter y ctrl+x).

Paso 5

- ✓ Como no tenía creado el directorio que se desea compartir, se procede a crearlo y darle permisos.

```
#mkdir /home/nahum/Publik-smb
```

```
#chmod 777 /home/nahum/Publik-smb/
```

```
root@debian:/home/nahum# mkdir /home/nahum/Publik-smb
root@debian:/home/nahum# chmod 777 /home/nahum/Publik-smb/
root@debian:/home/nahum# █
```

Luego de presionar ENTER nos mostrará las configuraciones que tiene dicho archivo, recuerde que si todo esta hecho bien, dará esta información al momento de ir dando ENTER.

```
(Publik)
path = /home/nahum/Publik-smb
read only = no
security = user
null password = yes
case sensitive = no
comment = Carpeta Compartida
guest ok = yes
browseable = yes
useask = 000
```

Paso 6

- ✓ Reiniciamos el servicio de samba de la siguiente manera.

```
# /etc/init.d/samba restart
```

```
root@debian:/home/nahum# /etc/init.d/samba restart
[ ok ] Restarting nmbd (via systemctl): nmbd.service.
[ ok ] Restarting smbd (via systemctl): smbd.service.
[ ok ] Restarting samba-ad-dc (via systemctl): samba-ad-dc.service.
root@debian:/home/nahum# █
```

Hemos terminado con la configuración de samba y ahora procedemos a conectar los clientes

Paso 7

- ✓ Instalar smbclient en la pc del cliente.

```
#aptitude install smbclient
```

Paso 8.

- ✓ En los equipos cliente que quieran tener acceso a este directorio compartido debe tener el mismo usuario y contraseña que hemos creado o conocerlas ya que al tratar de acceder le pedirá esos datos.



Paso 9.

- ✓ Colocar la contraseña y el usuario y aceptar, e inmediatamente podrá entrar al directorio que compartió.



COMPARTIR ARCHIVOS EN RED CON FTP

Configuraciones en la PC del Servidor:

Paso 1

- ✓ Instalar FTP como usuario **root** digitamos en la terminal:
apt-get install gftp

```
root@debian:/home/nahum# apt-get install gftp
```

Paso 2

- ✓ Configurar el documento del fichero **proftpd.conf**, digitando lo siguiente:

```
nano /etc/proftpd/proftpd.conf
```


Paso 3

En el archivo proftpd realizamos los siguientes cambios.

Des comentamos DefaultRoot ~

```
# Use this to jail all users in their homes
DefaultRoot ~

# Users require a valid shell listed in /etc/shells to login.
# Use this directive to release that constrain.
# RequireValidShell off
```

Paso 4

Reiniciamos el proftpd, digitando lo siguiente:

```
/etc/init.d/proftpd restart
```

Paso 5

✓ En la PC del cliente instalamos gftp.

```
apt-get install gftp
```

Paso 6

Abrimos un navegador o una carpeta y podremos establecer conexión

1- Poniendo la siguiente dirección en el navegador

```
ftp://nahum:villafranco@192.168.60.101
```

2- Abriendo una ventana y conectar con el navegador y podremos ver la carpeta de archivos del Servidor.



Conectarse al servidor

Detalles del servidor

Servidor: 192.168.60.101 Puerto: 21 - +

Tipo: FTP (con login)

Carpeta: /

Detalles de usuario

Nombre de usuario: nahum

Contraseña: ●●●●●●●●

Recordar esta contraseña

Ayuda Cancelar Conectar

COMPARTIR ARCHIVOS EN RED CON EL PROTOCOLO NFS

Para compartir archivos en red seguiremos los siguientes pasos.

Paso 1.

- ✓ Instalamos el servidor NFS con

```
#apt-get install nfs-kernel-server
```

```
root@debian:/home/nahum# apt-get install nfs-kernel-server
```

Paso 2.

- ✓ Creación del directorio a compartir y añadir a la lista de exportación.

Creamos el directorio en la ruta que deseemos y le otorgamos los permisos adecuados.

```
#mkdir /home/nahum/Publik  
#chmod 777 /home/nahaum/Publik
```

```
nahum@debian:~$ su  
Contraseña:  
root@debian:/home/nahum# mkdir Publik  
root@debian:/home/nahum# chmod 777 Publik  
root@debian:/home/nahum# █
```


Paso 3.

- ✓ Configuración del archivo export para agregar permisos.

Editamos el archivo `/etc/exports` y agregamos el directorio a la lista de exportación y agregamos los permisos que consideremos más adecuados.

#nano /etc/export

```
GNU nano 2.2.6          Archivo: /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_sub$
#
# 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)
#
/home/nahum/Publik 192.168.68.0/24(rw,sync,no_root_squash)
```

Entre los permisos que podemos otorgar a un directorio se encuentran:

- rw/ro= Exporta el directorio en modo lectura/escritura o sólo lectura.
- root_squash= Mapea los requerimientos del UID/GID 0 al usuario anónimo (por defecto usuario nobody con UID/GID 65534); es la opción por defecto.
- no_root_squash= No mapea root al usuario anónimo.
- all_squash= Mapea todos los usuarios al usuario anónimo.
- subtree_check/no_subtree_check= Si se exporta un subdirectorio (no un filesystem completo) el servidor comprueba que el fichero solicitado esté en el árbol de directorios exportado.
- sync modo síncrono= Requiere que todas las escrituras se completen antes de continuar; es opción por defecto.
- async modo asíncrono= No requiere que todas las escrituras se completen; más rápido, pero puede provocar pérdida de datos en una caída.
- secure= Los requerimientos deben provenir de un puerto por debajo de 1024
- insecure= Los requerimientos pueden provenir de cualquier puerto.

Paso 4.

- ✓ Reiniciamos el servidor usando.
`# /etc/init.d/nfs-kernel-server restart`

```
root@debian:/home/nahum# /etc/init.d/nfs-kernel-server restart
[ ok ] Restarting nfs-kernel-server (via systemctl): nfs-kernel-server.service.
root@debian:/home/nahum# █
```

EN LA PC CLIENTE SEGUIREMOS LOS SIGUIENTES PASOS

Paso 5

- ✓ Instalacion del cliente nfs usamos el siguiente comando.

```
#aptitude install nfs-common && apt-get install portmap
```

```
root@debian:/home/jorge# apt-get install nfs-common && apt-get install portmap
```

Paso 6.

- ✓ Creamos el directorio donde montaremos la carpeta compartida y le damos los permisos para poder editarla.

```
nahum@debian:~$ su
Contraseña:
root@debian:/home/nahum# mkdir Publik
root@debian:/home/nahum# chmod 777 Publik
root@debian:/home/nahum# █
```

Paso 7.

- ✓ Montamos el directorio compartido usando el siguiente comando

```
#mount -t nfs 192.168.60.101:/home/nahum/Publik /home/jorge/Escritorio/Share
```

```
root@debian:/home/jorge# mount -t nfs 192.168.60.101:/home/nahum/Publik /home/jorge/Escritorio/Share
```

Podemos verificar que directorio se está compartiendo del servidor ejecutamos el comando
`showmount -e ipservidor`

Paso 8.

- ✓ Montar automáticamente el directorio compartido.

Para montar automáticamente un directorio compartido con nfs editamos el archivo `/etc/fstab` agregando los datos correspondientes al recurso compartido.

```
root@debian:~# nano /etc/fstab
/etc/fstab: static file system information.
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
#-file system- <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda7 during installation
UUID=afa4d5fe-b208-463c-b535-c79604d624c8 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda8 during installation
UUID=4d274e55-ff2e-4643-a519-9799f78b4fbc none swap sw 0 0
192.168.60.101:/home/nahum/Publik /home/jorge/Escritorio/Share nfs rw 0 0
```

Así al reiniciar el cliente se montara automáticamente el directorio.

COMPARTIR ARCHIVOS EN RED CON SSH

Para compartir archivos en red con SSH haremos las siguientes configuraciones en la PC servidor.

Paso 1

- ✓ Instalar ssh server usando el comando.

```
# aptitude install ssh
```

Paso 2

Configuraciones del archivo sshd_config.

Configuramos el archivo sshd_config que se encuentra en /etc/ssh haciendo las siguientes modificaciones:

```
#nano /etc/ssh/sshd_config
```

1. PermitRootlogin NO "permite autenticarse como root"

```
# Authentication:
LoginGraceTime 120
PermitRootLogin no
StrictModes yes
```


Paso 3

Al finalizar las configuraciones reiniciamos el servidor ssh.

```
#/etc/init.d/ssh restart
```



CONFIGURACIONES NECESARIAS PARA LA PC DEL CLIENTE

Paso 4

Instalación para la PC cliente usamos el comando:

```
#apt-get install sshfs
```

Paso 5

Nos conectamos vía túnel por medio de terminal al servidor o también podemos hacerlo por interfaz, para hacerlo por vía túnel debemos escribir el siguiente comando en la terminal del cliente.

```
ssh -p 22 nahum@192.168.60.101
```

No pedirá contraseña de root del servidor y

podremos estar dentro de la terminal, como si fuera la nuestra.

Modo interfaz, para poder ingresar al servidor de forma grafica

