

Sistemas de Arquivos

Montaxe do Sistema de Arquivos

En Linux **todo o sistema de arquivos colga dun único directorio raíz**. Se queremos que os datos almacenados nun dispositivo externo (cd-rom, pendrive ..) sexan accesibles, hai que integrar os seus sistemas de arquivos no directorio raíz.

Esta operación chámase montaxe dos dispositivos de disco. Unha vez que un dispositivo está montado podemos acceder o seu contido como se dun directorio calquera se tratase.

mount

Sintaxe:

```
mount [-t sistArquivos] dispositivo directorioDeMontaxe
```

Exemplos:

```
mount /dev/sdb1 /mnt
mount /dev/sr0 /media/cdrom
```

Hoxe en día non soe ser necesario especificar o sistema de arquivos

Sen parámetros amosa os sistemas montados, coas súas opcións.

```
/dev/sdb1 on / type ext4 (rw,errors=remount-ro)
```

Cando rematamos de traballar con unha unidade de disco extraíble temos que desmontala. Podémolo facer de xeito manual, ou o apagar o sistema faixe de xeito automático.

umount

Sintaxe:

```
umount dispositivo o umount punto_de_montaje
```

Exemplos:

```
umount /media/cdrom0 ou umount /dev/sr0
```

Dende o entorno gráfico, sobre todo se empregamos Gnome, a montaxe de dispositivos externos faise de xeito automático.

Problema: Para montar unidades manualmente temos que **empregar o comando como root**.

Engadindo unha nova partición ó noso sistema

Xestión de particións

Dispoñemos de varios comandos para xestionar particións dende Linux

Consola	Entorno Gráfico
<ul style="list-style-type: none">• fdisk• cfdisk	<ul style="list-style-type: none">• Gparted• Discos (Gnome)

Formatear

mkfs → `mkfs -t tipo dispositivo`

Para poder formatear con formato vfat é necesario instalar **dosfstools**

```
mkfs -t ext4      /dev/sdb1
mkfs -t vfat      /dev/sdb5
```

Realmente mkfs é un front-end para outros comandos que nos permiten formatear para sistemas de arquivos concretos como mkfs.ext3, mkfs.vfat,...

Una vez formateadas, temos que montalas nunha carpeta

```
mount -t ext4 /dev/sdb1 /mnt/particion1
```

Agora xa podemos acceder o seu contido. **Para que se monten automaticamente o arrancar o sistema temos que incluílas en /etc/fstab**

mkswap → `mkswap dispositivo`

Formatea una partición como swap. Teríamos que engadila en **/etc/fstab**

```
mkswap /dev/sda7
```

fdisk -l

Amosa as unidades de disco e particións do noso sistema.

Para traballar co sistema de arquivos ntfs é necesario ter os seguintes paquetes:

- **ntfs-3g** : Permite montar en modo L/E unha partición ntfs
`mount -t ntfs-3g /dev/sda1 /mnt/win`
- **ntfsprogs**: Utilidades para xestionar particións ntfs: `ntfsfix`, `mkntfs`.

Distintas nomenclaturas para as nosas particións

Tradicionalmente empregamos para identificar os dispositivos de disco os nomes otorgados polo kernel: /dev/sda, /dev/sdb.

Problema: Os nomes de dispositivo son asignados segundo a orde no que se detecten. Non podemos asegurar que /dev/sdb sexa un pendrive concreto.

Solución: Empregar código únicos (UUID) ou etiquetas para referirnos aos dispositivos. Un **UUID** (IDentificador Único Universal) é un código que identifica dun xeito único unha partición.

blkid lista todos os identificadores dun sistema de arquivos:

```
root@debian:/home/usuario# blkid
/dev/sdb1: LABEL="MUSICA" UUID="4e4c0a05-f4d3-4a88-8a9a-755b843a3b4b"
TYPE="ext4" PARTUUID="3f8cdefd-01"
/dev/sdb2: LABEL="PELM-CM-^MCULAS" UUID="12D179E972F78AEA" TYPE="ntfs"
PTTYPE="dos" PARTUUID="3f8cdefd-02"
/dev/sdb3: LABEL_FATBOOT="DATOS" LABEL="DATOS" UUID="1BBD-A7DB" TYPE="vfat"
PARTUUID="3f8cdefd-03"
```

Tamén podemos listar os dispositivos destes xeitos:

- Por uuid

```
root@debian:/home/usuario# ls -l /dev/disk/by-uuid/
lrwxrwxrwx 1 root root 10 Abr 10 08:35 4e4c0a05-f4d3-4a88-8a9a-
755b843a3b4b -> ../../sdb1
```

- Podemos cambiar o UUID dunha partición con

```
tune2fs /dev/sdb1 -U `uuid`
```

- Por etiquetas

```
ls -l /dev/disk/by-label
```

- Para cambiar a etiqueta dunha partición:

Ext4	e2label /dev/sda5 nuevaEtiqueta
FAT	dosfslabel /dev/sda5 nuevaEtiqueta
NTFS	ntfslabel /dev/sda5 nuevaEtiqueta
Swap	mkswap -L nuevaEtiqueta /dev/sda5

Para cambiar a etiqueta da partición de swap debe empregar `swapoff/swapon`

Automatizando a montaxe de dispositivos

Linux necesita montar todos os sistemas de arquivos cada vez que arranca o sistema. Sería moi pesado ter que facelo manualmente.

Existe o arquivo **/etc/fstab** onde podemos enumerar los sistemas de arquivos que queremos que se monten de xeito automático cando se inicie o sistema.

<dispositivo>	<directorio de montaxe>	<sistema de arquivos>	<opcións>	<dump>	<pass>
/dev/sda5	/	ext4	errors=remount-ro	0	1
/dev/sda6	/home	ext4	defaults	0	2
/dev/sda7	none	swap	sw	0	0
/dev/sr0	/media/cdrom0	udf,iso9660	user,noauto	0	0

Opciones

rw	Lectura, escritura
defaults	Especifica que un dispositivo é de L/E, asíncrono, de bloques. Que os usuarios ordinarios non poden facer montaxes nel, e que se poden executar programas nel.
ro	Especifica que é de só lectura
noauto	Que non é montado de xeito automático. Emprégase tanto en cd-rom como en unidades de disco remotas, porque non se sabe si haberá algo nelas cando se arranque el sistema.
user	Permite que un usuario corrente monte un sistema de arquivos (en principio só o root pode facelo). Logo só o pode desmontar el. (aparte do root)
users	Permite que calquera usuario poda montar e desmontar el dispositivo, independentemente de quen o montase.
group	Permite a un usuario que non sexa root, montar o sistema de arquivos se pertence ao grupo do dispositivo.
umask=342	Permite montar un dispositivo con unos permisos específicos
loop	Permite montar un arquivo que contén unha imaxe (.iso) como sei fose un sistema de arquivos.

En **/etc/fstab** podemos listar unha partición de tres xeitos diferentes

- Empregando os códigos UUID

```
# / was on /dev/sda5 during installation
UUID=633e69e4-836b-437b-b4dc-d947ff56d32a / ext4 errors=remount-ro 0 1
```

- Empregando os nomes de dispositivo

```
/dev/sda5 / ext4 errors=remount-ro 0 1
```

- Empregando as etiquetas

```
LABEL=debian / ext4 errors=remount-ro 0 1
```

O campo **dump** determina se é necesario facer copias de seguridade del sistema de arquivos. Un 1 significa facer copia.

El campo **pass** emprégase para determinar en que orde se comprobarán os sistemas de arquivos cando o sistema arranca.

- 1 → Partición de arranque
- 0 → fsck non ten que comprobar o sistema de arquivos

Montar as particións mediante fstab soluciona os problemas:

- Non temos que montar as particións manualmente, **se montan automaticamente durante el arranque**.
- **Permitir aos usuarios montar diversas particións ou dispositivos**. Para elo temos que empregar a opción **user**. (Ex: Cdrom)

Nese caso os usuarios empregarán a seguinte sintaxis

```
mount <dispositivo> o bien mount <punto montaje>
```

Exemplo:

Para montar o cd-rom poderíamos facer unha de estas dúas cousas:

```
mount /dev/sr0 o bien mount /mnt/cdrom
```

Comandos para controlar o espazo en disco

df

Amosa estatísticas dos sistemas de arquivos que están montados: espazo total, espazo usado, el espacio libre, etc.

`df /dev/hda` → Amosa información só dese sistema de arquivos

`df -h` → Amosa o tamaño e espazo libre en Gb, Mb, ..

du

Amosa o espazo de disco empregado por unha carpeta e todos os seus subdirectorios.

`du -h` → Amosa os tamaños empregando múltiplos do byte.

`du -ah` → Tamén amosa información dos arquivos, nos só de los directorios.

Otros comandos relacionados con la gestión de disco

fsck

Comproba e repara sistemas de arquivos

```
fsck /dev/hda1
```

tune2fs

Cando arrancamos, o sistema automaticamente monta os sistemas de arquivos listados en `/etc/fstab`. Despois dun número de montaxes `x` faise un chequeo do sistema de arquivos con `fsck`. Este comando establece o número de montaxes a partir dos cales o sistema é comprobado con `fsck`.

`tune2fs -l /dev/sda5` → Amosa cantas veces foi montado, y cantas faltan para a súa seguinte revisión automática. Por defecto é 27.

`tune2fs -c 30 /dev/sda5` → Chequearía o sistema cada 30 reinicios.

Montando sistemas de archivos remotos (NFS)

NFS (Network File System) permite compartir carpetas entre ordenadores Linux de xeito que temos a impresión de estar traballando no noso disco duro local.

Para elo necesitamos dúas máquinas:

- **Servidor:** será onde compartamos a carpeta (192.168.63.100)
- **Cliente:** Será dende onde accedamos ás carpetas compartidas (192.168.63.4)

Instalación del Software

- **Cliente**
É preciso ter instalado o paquete **nfs-common**
Pode que xa o teñas instalado.

- **Servidor**
Necesitamos instalar **nfs-kernel-server**.
O instalalo inicia o servidor nfs (os demos nfsd, mountd) e crea un arquivo **/etc/exports** baleiro.

Unha vez instalado, podemos comprobar como o **servicio nfs** está escoitando no porto **2049**. Para elo empregamos **ss (socket statistics)**, este programa amosa as conexións de rede abertas no noso sistema e en que portos están escoitando os diversos servizos. (substitúe a netstat).

```
root@NFS-01:~# ss -4ta | grep nfs
State      Recv-Q      Send-Q      Local Address:Port      Peer Address:Port
LISTEN     0            64          0.0.0.0:nfs              0.0.0.0:*
```

Por defecto ss non amosa os números de porto dos servizos máis habituais. Para elo

```
root@NFS-01:~# ss -4nta | grep 2049
LISTEN     0            64          0.0.0.0:2049              0.0.0.0:*
```

O noso equipo está escoitando peticións no porto 2049 polo que o servidor nfs está funcionando.

Configuración do Servidor

En **/etc/exports** configúranse as carpetas que queremos compartir, con que permisos e con que equipos.

Vemos la sintaxe para compartir de NFS v3 que é máis sinxela.

```
/srv/nfs/compartida1 192.168.63.4(ro)
/srv/nfs/compartida2 192.168.0.0/255.255.0.0(rw)

directorio           máquinaA(opción,opción) máquinaB(opción,opción)
```

Temos que crear as correspondentes carpetas compartidas

Cada vez que modificamos o arquivo /etc/exports é necesario reiniciar o servidor nfs para que actualice los cambios. Para elo:

```
systemctl restart nfs-kernel-server
```

Olo: O comando anterior non debe devolver nada. Se o executalo obtemos algo como

```
root@NFS-01:~# systemctl restart nfs-kernel-server
Job for nfs-server.service canceled.
```

É que temos un erro, seguramente en /etc/exports. Para obter máis información:

```
root@NFS-01:~# systemctl status nfs-kernel-server
● nfs-server.service - NFS server and services
   Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled; vendor preset: enabled)
   Active: failed (Result: exit-code) since Wed 2020-04-15 01:44:08 CEST; 9s ago
     View: https://openstack.org
   Docs: man:exports(5)
   Link: https://www.fedoraproject.org/
   Unit: nfs-kernel-server.service
   CGroup: /systemd/system/nfs-kernel-server.service
           └─ nfs-kernel-server.service

Abr 15 01:44:08 NFS-01 systemd[1]: Starting NFS server and services...
Abr 15 01:44:08 NFS-01 exportfs[542]: exportfs: /etc/exports [2]: since nfs-utils version 1.0.x
Abr 15 01:44:08 NFS-01 exportfs[542]: exportfs: Failed to stat /srv/entregar: No such file or directory
```

Tíñamos mal a ruta da carpeta.

Configuración do cliente

Comprobamos que temos instalado o paquete **nfs-common**, se non é así instalámolo

```
dpkg -l | grep nfs-common
```

Podemos comprobar que accedemos o arquivo export do servidor con:

```
showmount -e 192.168.63.100
```

Creamos a carpeta onde queremos montar a carpeta compartida

```
mkdir /mnt/remota
```

Montamos a carpeta compartida no sistema de arquivos local

```
mount -t nfs 192.168.63.100:/srv/nfs/compartida /mnt/remota
```

Comprobamos que está montada co comando mount

Se a montamos como ro, non poderemos modificar o seu contido.

Modos de compartir una carpeta

No apartado anterior compartimos a carpeta en modo só lectura, pero temos moitas opcións para compartir carpetas.

Compartimos outra carpeta chamada compartida2 en modo lectura/escritura. Creámola e engadimos a seguinte liña no /etc/exports

```
/srv/nfs/compartida2 192.168.63.4 (rw)
```

Problema: Se a montamos dende o ordenador cliente, tampouco nos deixa escribir nela, pero a mensaxe de erro é diferente

- Compartida1 → Sistema de arquivos de Só Lectura
- Compartida2 → Permiso denegado

Trátase dun problema de permisos. Pero se facemos comprobamos os permisos

```
ls -ld /mnt/compartida2  
rwxrwxr-x root root /mnt/compartida2
```

Problema: Aínda que na máquina cliente estamos como root, non podemos modificar a carpeta

Motivo: Por dúas razóns:

- Por razóns de seguridade: O root do cliente non se considera como o root do servidor cando intentamos acceder a través de NFS
- Polos permisos de compartida2.

Non podemos escribir porque aplícanse os permisos do grupo outros.

Solución: No servidor cambiamos os permisos de compartida2

```
chmod 777 /compartida2
```

Os cambios de permisos no Servidor NFS aplícanse inmediatamente no cliente, así que xa poderemos modificar o seu contido dende o cliente.

Usuarios nobody nogroup

Problema:

- Como root da máquina cliente creamos un arquivo en /mnt/compartida2.
- Facemos un listado largo de ese arquivo.
- Creámolo como root, pero asignáronsele un usuario e grupo anónimos (nobody, nogroup)

Motivo:

- Como xa vimos no apartado anterior, cando accedemos a unha carpeta montada por NFS, por defecto non fai corresponder ó root do Cliente co root do Servidor

Solución:

- Cambiamos o comportamento por defecto modificando as opcións de compartición en /etc/exports

```
/compartida ip (rw,no_root_squash)
```

Aplicamos o cambio (reiniciando o servidor nfs) e creamos algo como root dende o cliente. O creado terá asignado como usuario e grupo ao root.

Montar unha carpeta compartida durante o arranque

Podemos automatizar a montaxe dunha carpeta remota mediante /etc/fstab

```
192.168.63.100:/compartida/mnt/remoto nfs defaults 0 0
```

Problema: Se iniciamos o cliente e o servidor non está accesible, producirse un erro no arranque do cliente e teremos que esperar un timeout para finalizar o arranque.

```
/dev/sda5: clean, 35687/1099440 files, 389224/4394240 blocks
[ 2.701285] [drm:vmw_host_log [vmwgfx]] *ERROR* Failed to send host log messa
ge.
[ 2.701295] [drm:vmw_host_log [vmwgfx]] *ERROR* Failed to send host log messa
ge.
[ *** ] A start job is running for /mnt/apuntes (45s / 1min 31s)
```

Solución: O parámetro **nofail**. Este parámetro indica que se falla a montaxe do dispositivo non agarde por un timeout.

```
192.168.63.100:/compartida/mnt/remoto nfs defaults,nofail 0 0
```

NFS v4

Pensado para mellorar a seguridade de NFS. Hai que xente que NFS significa “No File Secure”, debido aos buracos de seguridade que presenta el servizo.

Características:

- Autenticación dos usuarios mediante **kerberos** e non dos equipos clientes.
- Emprego dun único porto TCP 2049, o que permite configurar un cortafuegos.
- Permite ACL's

Para probalo, podemos simplificar a súa configuración e indicar que non empregue kerberos.

- En /etc/default/nfs-common:
NEED_GSSD=no
- En /etc/default/nfs-kernel-server:
NEED_SVCGSSD=no

Para compartir una carpeta engadimos en /etc/exports

```
# Compartida con un equipo concreto
/compartida1 192.168.63.4(rw,sync,fsid=00,crossmnt,no_subtree_check)
# Compartida con todos os equipos
/compartida1 *(rw,sync,fsid=00,crossmnt,no_subtree_check)
```

Cuotas de Disco

As cuotas **permitennos asignar o espazo máximo de dispositivo de disco que pode empregar un determinado usuario ou grupo**. Para instalalas:

```
apt-get install quota
```

Habilitando as cuotas

Unha vez instalado, temos que indicar en que sistemas de arquivos queremos empregar as cuotas. Para elo modificamos /etc/fstab.

- usrquota -> Habilita cuotas para usuarios
- grpquota -> Habilita cuotas para grupos.

```
/dev/sdal / ext4 defaults,errors=remount-ro,usrquota,grpquota 0 1
```

Para que os cambios teñan efecto temos que montar outra vez o sistema de arquivos. **Para facelo sen reiniciar**

```
mount -o remount /
```

Comproba as opcións de montaxe da partición con **mount**, antes e despois de executar o comando anterior. Se non reiniciamos temos que executar dous comandos comezar a empregaras cuotas:

- `quotacheck -cugm /`

Crea no directorio raíz os arquivos **aquota.user** y **aquota.group** onde se almacenarán as cuotas dos usuarios e grupos.

- `quotaon /` e `quotaoff /`

Para habilitar ou deshabilitar as cuotas nese sistema de arquivos.

Comprobando e establecendo cuotas

Podemos comprobar o espazo empregado polos usuarios con `repquota -a`

```
Block grace time: 7days; Inode grace time: 7days
      Block limits
User      used  soft  hard  grace
-----
root      --  926248    0    0      7days
usuario   --   324    0    0      7days
xian      --   246    0    0      7days
      File limits
      used  soft  hard  grace
-----
root      39795    0    0
usuario   61     0    0
xian      42     0    0
```

Observamos espazo empregado, arquivos creados, e que non ten ningún límite.

Para asignar cuotas empregamos:

- `edquota -u pepe` → Para usuarios
- `edquota -g grupo` → Para grupos

Executa un editor para os valores de cuota dese usuario. Eses valores especifícanse cos seguintes campos:

- **Blocks:** Cantidade de bloques de 1 KB empregados polo usuario.
- **Inodes:** Número de arquivos pertencentes ao usuario. Un **inodo** (Index Node) contén todos os atributos do arquivo, así como os bloques que ocupa en disco.
- **Soft:** Limite de gracia. Limite de bloques de 1 KB que o usuario pode empregar e que **pode superar ata que sexa excedido o período de gracia** (de modo predeterminado son 7 días).
- **Hard:** Limite absoluto. Limite de espacio en disco que non pode ser superado.

A **cuota de gracia** establece os límites de bloques ou **inodos** que un usuario ten nunha partición. Cando o usuario excede o límite establecido pola cuota de gracia, o sistema advirte ao usuario, pero permítelle continuar escribindo ata que transcorra o tempo establecido polo período de gracia.

Por defecto temos un **periodo de gracia** de 7 días que se pode modificar con **edquota -t**.

Samba

SMB (Server Message Block) é un protocolo que nos permite conectarnos a carpetas compartidas en Windows. Tamén coñecido como **CIS (Comom Internet Filesystem)**

Que podemos facer con Samba?

- Compartir carpetas Linux con máquinas Windows.
- Compartir carpetas Windows con máquinas Linux.
- Compartir impresoras.
- Autenticar clientes logeándose contra un dominio Windows.
- Configurar un Servidor de Directory Activo tanto para clientes Windows como para clientes GNU/Linux.

Ex1: Acceder a carpetas compartidas en Linux dende Windows

Temos dúas máquinas conectadas en rede.

- Comprobamos conectividade

Linux → 192.168.153.3	Windows → 192.168.153.5.
<pre>rojas@hispafuentes:~\$ ping 192.168.153.5 PING 192.168.153.5 (192.168.153.5): 56 data bytes 64 bytes from 192.168.153.5: icmp_seq=0 ttl=128 time=1.0 ms 64 bytes from 192.168.153.5: icmp_seq=1 ttl=128 time=0.8 ms 64 bytes from 192.168.153.5: icmp_seq=2 ttl=128 time=1.2 ms 64 bytes from 192.168.153.5: icmp_seq=3 ttl=128 time=0.8 ms --- 192.168.153.5 ping statistics --- 4 packets transmitted, 4 packets received, 0% packet loss round-trip min/avg/max = 0.8/0.9/1.2 ms rojas@hispafuentes:~\$</pre>	<pre>C:\Documents and Settings\rojas>ping 192.168.153.3 Haciendo ping a 192.168.153.3 con 32 bytes de datos: Respuesta desde 192.168.153.3: bytes=32 tiempo=23ms TTL=64 Respuesta desde 192.168.153.3: bytes=32 tiempo<1m TTL=64 Respuesta desde 192.168.153.3: bytes=32 tiempo<1m TTL=64 Respuesta desde 192.168.153.3: bytes=32 tiempo<1m TTL=64 Estadísticas de ping para 192.168.153.3: Paquetes: enviados = 4, recibidos = 4, perdidos = 0 (0% perdidos), Tiempos aproximados de ida y vuelta en milisegundos: Mínimo = 0ms, Máximo = 23ms, Media = 5ms</pre>
root, user1	administrador, xoaan (limitado)

Instalación de Samba

```
apt-get install samba
```

Examinamos portos abertos

O servidor Samba escoita peticións nos portos

```
root@NFS-01:~# ss -4ta
State      Recv-Q    Send-Q    Local Address:Port    Peer Address:Port
LISTEN     0          50        0.0.0.0:microsoft-ds   0.0.0.0:*
LISTEN     0          50        0.0.0.0:netbios-ssn       0.0.0.0:*

root@NFS-01:~# ss -4tan
State      Recv-Q    Send-Q    Local Address:Port    Peer Address:Port
LISTEN     0          50        0.0.0.0:445             0.0.0.0:*
LISTEN     0          50        0.0.0.0:139             0.0.0.0:*
```

Configuración de Samba

O arquivo de configuración é `/etc/samba/smb.conf`.

- Facemos unha copia de seguridade

```
cp /etc/samba/smb.conf /etc/samba/smb.conf.bak
```

- Borramos o orixinal creamos un novo baleiro.

```
#Opcións Globais
[global]
    workgroup = red
    encrypt passwords = yes

#Defino carpetas a compartir
[test]
    path = /srv/samba/DIR1
    read only = no
[test1]
    path = /srv/samba/DIR2
    read only = yes
```

Explicación (case sensitive):

- **[global]**: Información que afecta a todas as configuracións
- **workgroup**: grupo de traballo da rede de Windows
- **encrypt passwords**: Necesario
- **[test]** y **[test1]**: Nomes que terán as carpetas compartidas se accedemos dende a máquina con Windows.
- **path**: Carpeta da nosa máquina GNU/Linux a compartir
- **read only**:
 - 'yes' -> Compartimos en modo lectura.
 - 'no' -> Compartimos con permiso L/E.

Podemos comprobar que non temos ningún erro de sintaxe no arquivo de configuración con `testparm`.

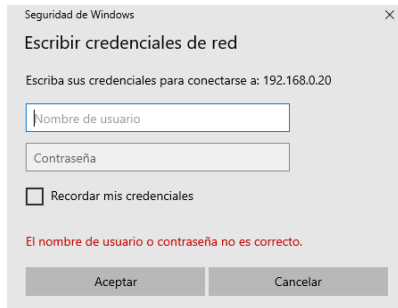
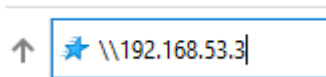
Cada vez que modifiquemos o arquivo de configuración temos que reiniciar samba (**smbd**: Samba Daemon)

```
systemctl restart smbd
```

Para comprobar que todo está correcto

```
root@NFS-01:/etc/samba# systemctl status smbd
● smbd.service - Samba SMB Daemon
   Loaded: loaded (/lib/systemd/system/smbd.service; enabled; vendor preset:
   enabled)
   Active: active (running) since Sat 2020-04-18 23:37:36 CEST; 1min 58s ago
```

Accedendo dende Windows



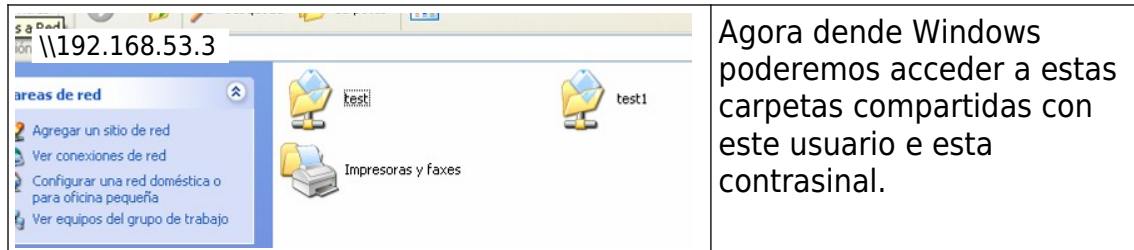
Na barra de direccións do explorador de arquivos do noso equipo Windows. Escribimos a dirección de rede do noso servidor Debian.

Problema:
Pídenos un usuario e unha contrasinal para poder acceder.

Configurando usuarios

Temos que indicar que usuarios poden conectarse as nosas carpetas compartidas en Linux dende outros sistemas. Estes usuarios teñen que existir no sistema GNU/Linux, senón dende Windows deixaría conectarte pero nunca poderías crear arquivos nin carpetas na máquina Linux.

```
smbpasswd -a user1
New SMB password: abc123.
Retype new SMB password: abc123.
```



Se o usuario de Windows se chamase user1 con contrasinal abc123. entraría ás carpetas sen necesidade de autenticación.

- Listar usuarios
`pdbedit -L`
- Cambiar contrasinal samba
`smbpasswd user1`
- Eliminar usuarios
`smbpasswd -x user1`
- **Problema:** En ambas carpetas só podemos ler pero non escribir. Iso non é correcto, xa que test non a temos compartida como só lectura.
- **Solución:** No servidor Debian temos que modificar os permisos do grupo "outros" para que así podan escribir os usuarios non autenticados.

```
chmod 777 /srv/samba/DIR1
```


Compartir carpeta de xeito anónimo (pública)

Queremos compartir unha carpeta en modo público, é dicir, que permita acceder sen necesidade de contrasinal.

- **Creamos carpeta**

Lle asignamos como propietario **nobody** e como grupo **nogroup**.

```
chown nobody.nogroup /srv/samba/publica/
```

- **Configuración Samba**


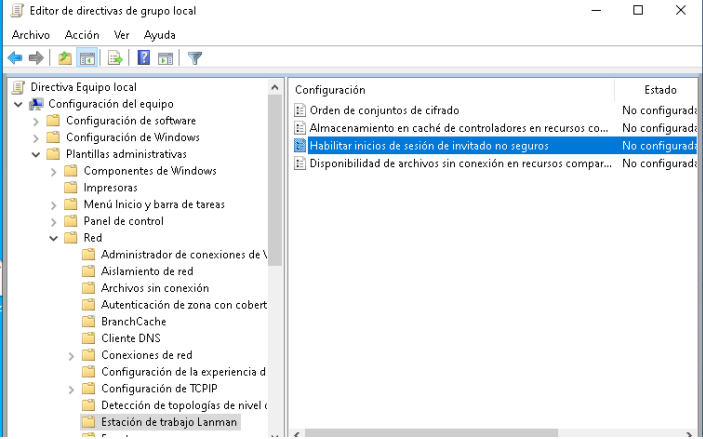
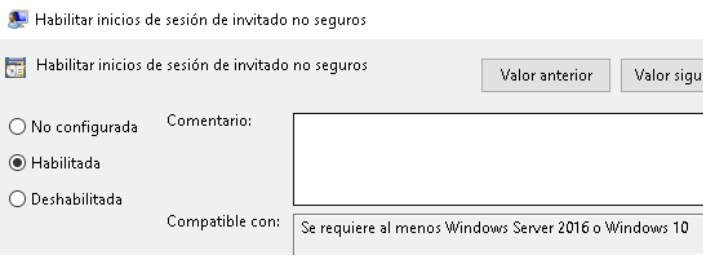
```
#Opciones Globais
[global]
    workgroup = red
    encrypt passwords = yes
    log file = /var/log/samba/%m
    log level = 1
    security = user
    guest account = nobody
    map to guest = Bad User
#Defino carpetas a compartir
[pública]
    path = /srv/samba/publica
    read only = yes
    guest ok = yes
```

- **guest ok:** Permitimos que un usuario invitado entre no servidor con estes parámetros

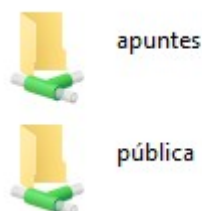
- **Reiniciamos o servidor samba**

```
systemctl restart smbd
```

- **Problema:** Agora Windows non deixa conectarse a carpetas compartidas consideradas inseguras. [link](#)
 - **Solución:** No equipo cliente cambiar unha directiva.

 gpedit.msc Documento de la consola común de ...	Dende unha conta de administrador, executamos gpedit.msc
	Configuración de Equipo -Plantillas Administrativas -Red -Estación de Trabajo Lanman -Habilitar inicios de sesión invitados no seguros
	Habilitamos
Reiniciamos o equipo Windows para asegurarnos que a directiva foi aplicada	

Ó conectarnos deberían aparecer as carpetas compartidas



Restrinxir o acceso ás carpetas compartidas

Poderíamos facelo no servidor samba, mediante os permisos locais nas carpetas compartidas. Tanto empregando os permisos UGO como as ACL's.

Por exemplo

```
ls -l /srv/samba/privada rojas G-Profesores rwxr-x---
```

Se comparto esa carpeta como lectura/escritura, accedendo a través de samba

- Rojas tería todos os permisos
- Os do grupo G-Profesores poderían ver o que hai dentro
- O resto nada.

Pero tamén podemos limitar o acceso a unha carpeta empregando **smb.conf**.

```
[privada]
    path = /srv/samba/privada
    read only = yes
    valid users = lua
```

El usuario lua, ten que existir no sistema e ser usuario de samba.

Chequear problemas

Smbstatus

Fai un test para ver as conexións producidas a nosa máquina Linux mediante samba

```
root@NFS-01:/etc/samba# smbstatus

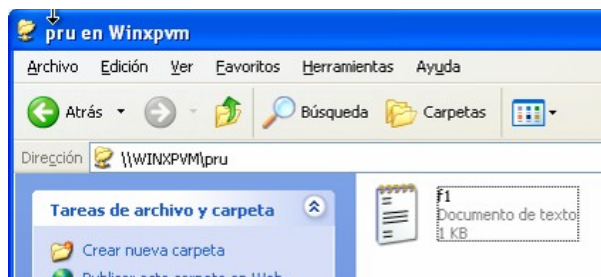
Samba version 4.9.5-Debian
PID      Username  Group    Machine              Protocol Version Encryption Signing
-----
1281     lua      lua      192.168.0.22         SMB3_11 -    partial(AES-128-CMAC)
```

Ex 2: Acceder a carpetas compartidas en Windows dende Linux

Antes configurado samba para que funcione como servidor, para poder montar carpetas compartidas dende Windows temos que instalar

```
apt-get install cifs-utils
```

Compartimos unha carpeta na nosa máquina Windows chamada **pru**. Nesa carpeta creamos un arquivo de texto. Esta carpeta a creamos co usuario rojas e nos aseguramos de que teña permisos para acceder a través da rede.



Intenta conectarnos con el usuario rojas desde Linux.

```
apt-get install cifs-utils
mount -t cifs -o username=rojas //winxpvms/pru /home/rojas/DIR3
Password:
#Introducimos o contrasinal do usuario de Windows
```

Tamén podemos conectarnos a unha carpeta compartida no dominio

```
mount -t cifs -o username=rojas //172.20.2.2/Software /mnt
```

Tamén podemos empregar **nautilus** para acceder ás carpetas Windows. Para elo na barra de direccións smb://172.20.3.60

