

1 Creación e configuración do esqueleto da estrutura de cartafoles con Samba4

- Neste apartado imos crear a estrutura de carpetas nas que se almacenarán a información persoal e o perfil de cada usuario, así como os documentos compartidos entre distintos usuarios.
- Seguiremos uns pasos moi similares aos da [Creación e configuración do esqueleto da estrutura de cartafoles](#) da parte III do curso, xa que a estrutura ten moitas partes en común.

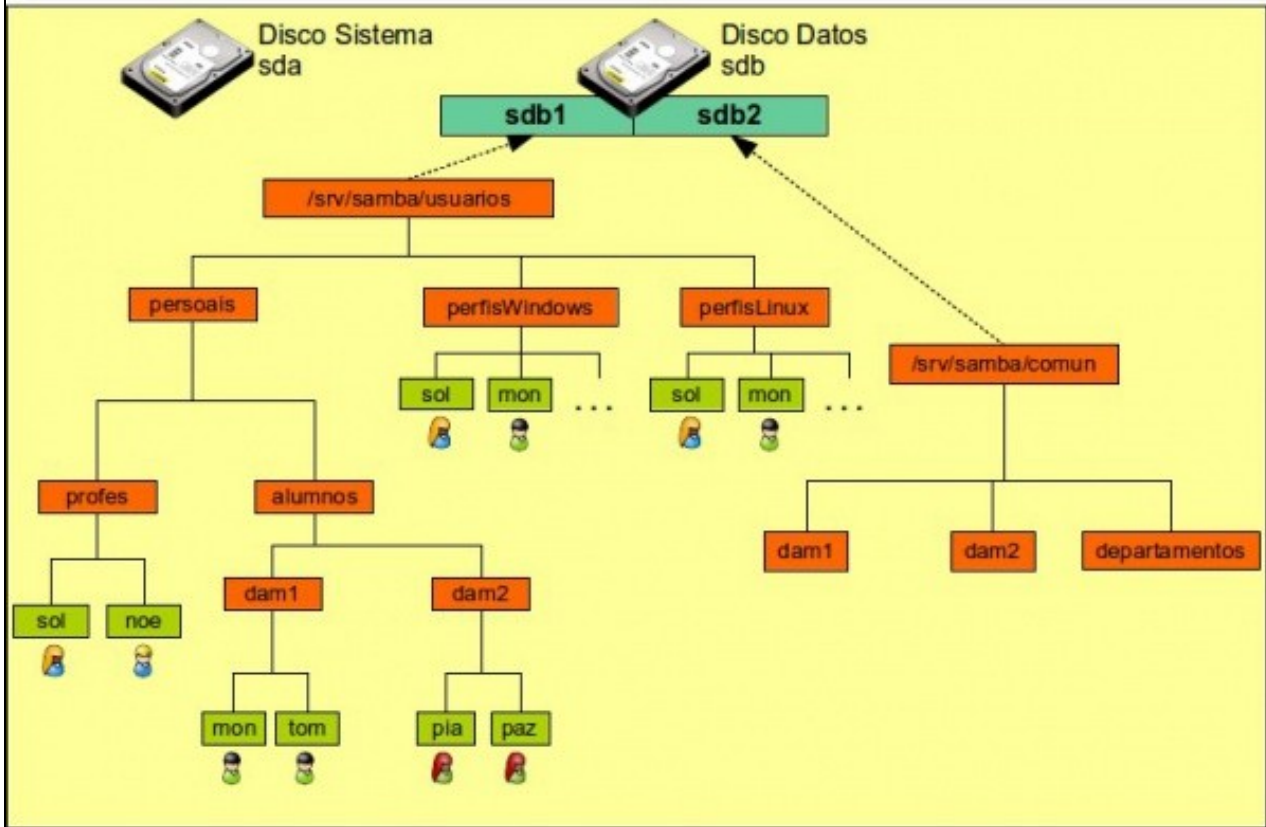
1.1 Sumario

- [1 Esquema de carpetas](#)
- [2 Engadir segundo disco a dserver00](#)
- [3 Crear particións e formato](#)
- [4 Montaxe das particións](#)
- [5 Crear esqueleto](#)

1.2 Esquema de carpetas

- A estrutura de carpetas, que se mostra a continuación, é moi similar á creada na parte III, salvo un par de detalles:
 - ◆ Os puntos de montaxe das carpetas son distintos, e colocáronse dentro de */srv/samba*. Isto non ten maior importancia, e poderían estar en calquera outra carpeta, pero simplemente facémolo así para separar un chisco máis as carpetas de datos do sistema de ficheiros do sistema Debian do servidor.
 - ◆ Por cada usuario, ademais da carpeta persoal, que ten como obxectivo almacenar os documentos persoais de ese usuario, hai unha carpeta dentro de *perfisWindows* e outra dentro de *perfisLinux*. A razón de definir esta estrutura é a seguinte:
 - ◇ Separar a carpeta cos documentos persoais do usuario da carpeta ou carpetas nas que se almacena a súa configuración persoal dos distintos sistemas operativos que temos no dominio. Iso pode facilitar a xestión dos perfís, a xestión de cotas e a migración dun almacenamento a outro se fose necesario.
 - ◇ No caso de que teñamos máis sistemas operativos instalados nos equipos clientes do dominio (por exemplo, outra distribución de Linux, como Linux Mint, Fedora ou OpenSuse), poderíamos crear outra carpeta (*perfisFedora*, por exemplo) na que se almacenaría o perfil de cada usuario nese sistema. Hai que ter en conta que se se usase a mesma carpeta de perfil para distintas distribución de Linux poderíamos ter problemas se as distribucións non son moi "compatibles" entre si nese sentido. Por poñer un exemplo, se nunha hai instalada unha versión de Firefox moi máis antiga que na outra, a carpeta de configuración de unha pode dar problemas na outra.

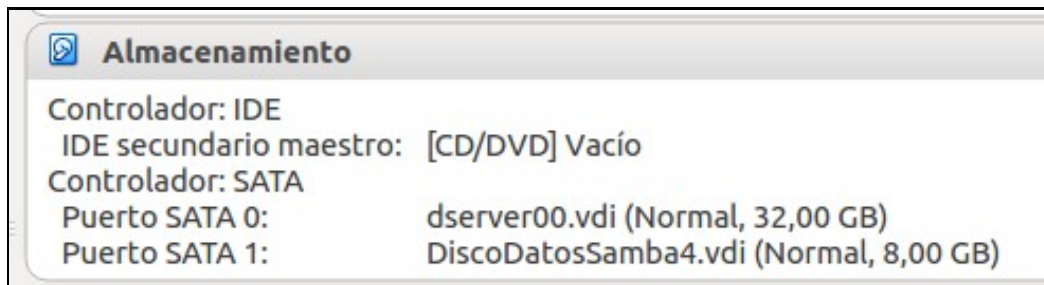
Estructura de carpetas dserver00: Usuarios



Esquema de carpetas do dominio

1.3 Engadir segundo disco a dserver00

- Da mesma forma que fixemos na Parte III do curso, engadimos un novo disco a *dserver00*. Debe ser un disco novo, distinto ao disco que engadimos na parte III, xa que imos almacenar información diferente. Téñase en conta que non se debe borrar o ficheiro do disco virtual que se engadiu na parte III, xa que aínda que agora non o usemos máis adiante restauraremos instantáneas anteriores e si o precisaremos:



Segundo disco engadido a *dserver00*

1.4 Crear particións e formato

- Seguimos os mesmos pasos vistos no punto de **crear particións e formato** da parte III, repartindo o espazo do disco en dúas particións.
- Vemos a continuación o que nos debe devolver o comando **blkid** tras o particionado e formatado:

```

root@dserver00:~# blkid
/dev/sda5: UUID="87914631-9ef4-4f28-a200-2b02133af184" TYPE="swap"
/dev/sda1: UUID="8e286597-4bc3-4954-acf3-9cd024474e00" TYPE="ext4"
/dev/sda6: UUID="4f7c3d8b-ea7f-4cee-8d6c-06f3b38afa9b" TYPE="ext4"
/dev/sdb1: UUID="6defba49-6bf4-493b-b521-d379d7cbc2ab" TYPE="ext4" LABEL="Usuarios"
/dev/sdb2: UUID="d178ccfa-e0d9-479d-9bc7-e88de467374f" TYPE="ext4" LABEL="Comun"
root@dserver00:~# █

```

Partición do disco `/dev/sdb`

1.5 Montaxe das particións

- Neste caso, en lugar de utilizar os permisos POSIX propios de Linux, imos utilizar ACLs para establecer os permisos na estrutura de carpetas do dominio. Máis adiante abordaremos esta configuración de permisos baseada en ACLs, pero de momento imos a instalar o soporte para ACLs no servidor Debian e actualalas para as dúas unidades do disco `sdb`.

- Instalamos o paquete `acl` en `dserver00`.

```
apt-get install acl
```

- Engadimos no ficheiro `/etc/fstab` as dúas liñas que se mostran a continuación, que permitirán montar as dúas particións creadas en `/srv/samba/usuarios` e `/srv/samba/comun`, engadir o parámetro `acl` para activar nelas o soporte de permisos por ACLs:

```

/dev/sdb1    /srv/samba/usuarios    ext4    user_xattr,acl 0 0
/dev/sdb2    /srv/samba/comun       ext4    user_xattr,acl 0 0

```

- Creamos as carpetas nas que imos montar as dúas particións e montámolas con `mount -a`:

```

root@dserver00:~# mkdir -p /srv/samba/usuarios
root@dserver00:~# mkdir -p /srv/samba/comun
root@dserver00:~# mount -a

```

- Comprobamos que as particións están montadas nas carpetas co comando `mount`:

```

root@dserver00:~# mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,relatime,size=10240k,nr_inodes=62125,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,noexec,relatime,size=50892k,mode=755)
/dev/disk/by-uuid/54aca566-b079-4404-8419-433c7bf57b3c on / type ext4 (rw,relatime,errors=remount-ro,user_xattr,barrier=1,data=ordered)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
tmpfs on /run/shm type tmpfs (rw,nosuid,nodev,noexec,relatime,size=248000k)
rpc_pipefs on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
/dev/sdb1 on /srv/samba/usuarios type ext4 (rw,relatime,user_xattr,barrier=1,data=ordered)
/dev/sdb2 on /srv/samba/comun type ext4 (rw,relatime,user_xattr,barrier=1,data=ordered)
root@dserver00:~# █

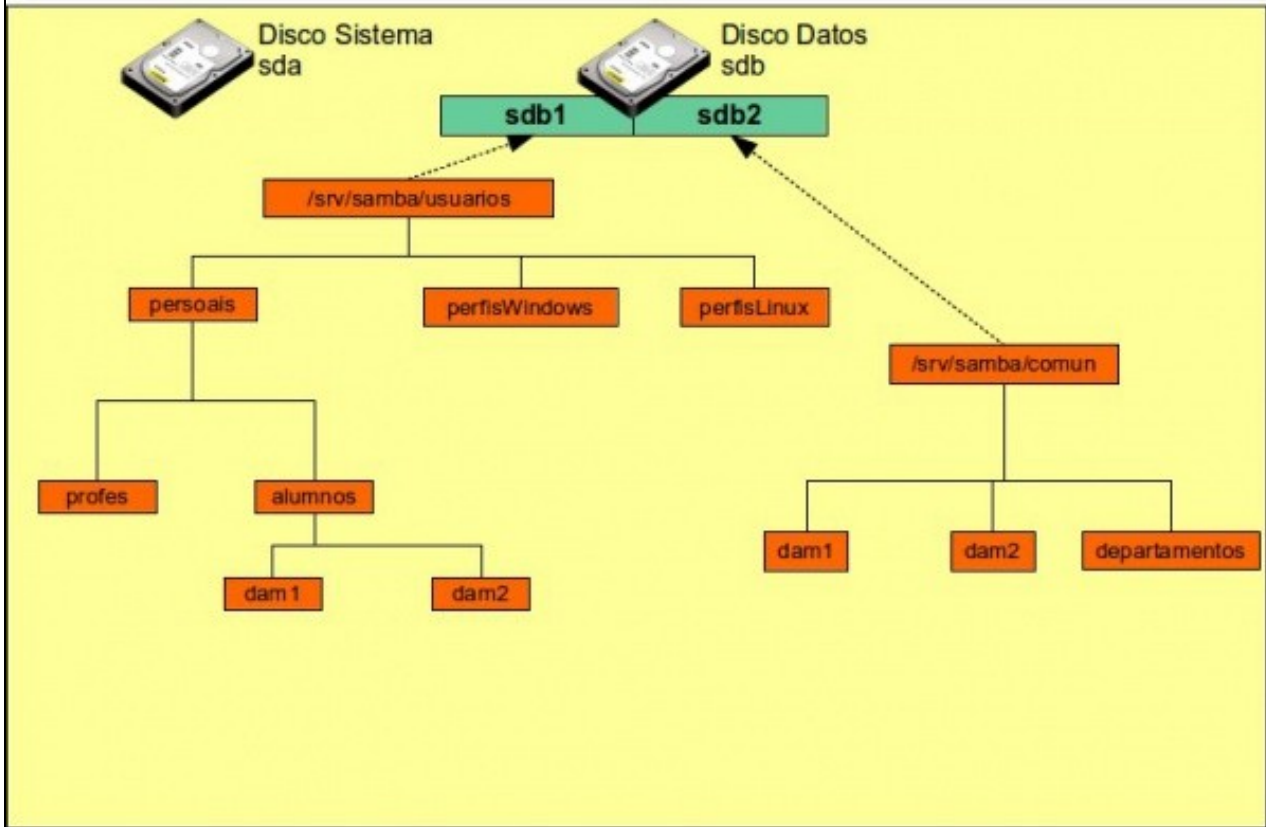
```

Execución do comando `mount` onde podemos ver as particións montadas

1.6 Crear esqueleto

- Da mesma forma que fixemos na [Parte III do curso](#), agora imos crear o esqueleto de carpetas base tanto para os documentos persoais dos usuarios do dominio como para os documentos compartidos entre eles.

Estrutura de carpetas dserver00: Esqueleto



Esquema de carpetas do dominio

- Seguindo a mesma filosofía, imos realizar toda a configuración con scripts, pois así sempre podemos recorrer a eles e caso de necesidade e ademais deixamos rexistrado en ficheiros a traza do que realizamos.
- Neste caso imos crear un directorio en *dserver00* no que ter organizados todos os scripts (podemos facelo dentro do directorio */root* ou en calquera outro sitio):

```
mkdir scripts_samba4
cd scripts_samba4
```

• FICHEIRO DE CURSOS: f00_cursos.txt

```
dam1
dam2
```

• SCRIPT DE VARIABLES GLOBAIS: 00_variables.sh

```
#!/bin/bash

# Define variable globais que van usar os demais scripts

#Variables
DIR_USUARIOS=/srv/samba/usuarios
DIR_COMUN=/srv/samba/comun

# Exportar variables
# Nos scripts que se van usar a continuación non faría falla que se exportasen as variables.
# Pero quedan exportadas por se a posteriori calquera dos scripts que vai importar
# o contido deste ficheiro precisase chamar a outros scripts que precisasen usar estas variables
export DIR_USUARIOS
export DIR_COMUN
```

• SCRIPT: 01_crear_esqueleto.sh

```
#!/bin/bash

#Chamar ao script de variables, temos varias opcións:

. ./00_variables.sh # Tamén podería ser: source ./00_variables.sh

#Crear esqueleto profes
#Por se executamos o script varias veces, comprobamos se xa existe o directorio
test -d $DIR_USUARIOS/persoais/profes || mkdir -p $DIR_USUARIOS/persoais/profes

#Crear esqueleto para os perfis de Windows
test -d $DIR_USUARIOS/perfisWindows || mkdir -p $DIR_USUARIOS/perfisWindows

#Crear esqueleto para os perfis de Linux
test -d $DIR_USUARIOS/perfisLinux || mkdir -p $DIR_USUARIOS/perfisLinux

#Crear esqueleto alumnos e comun
#Lemos o ficheiro cursos e procesamos cada curso
for CURSO in $(cat f00_cursos.txt)
do
    test -d $DIR_USUARIOS/persoais/alumnos/$CURSO || mkdir -p $DIR_USUARIOS/persoais/alumnos/$CURSO
    test -d $DIR_COMUN/$CURSO || mkdir -p $DIR_COMUN/$CURSO
done

#Crear en comun a carpeta para os departamentos
test -d $DIR_COMUN/departamentos || mkdir -p $DIR_COMUN/departamentos
```

- Executamos o script para crear o esqueleto de carpetas:

```
sh 01_crear_esqueleto.sh
```

- Comprobamos a súa execución, visualizando co comando **tree** o contido de */srv/samba/usuarios* e */srv/samba/comun*:

```
tree /srv/samba/usuarios/
/srv/samba/usuarios/
??? lost+found
??? perfisLinux
??? perfisWindows
??? persoais
    ??? alumnos
    ?   ??? dam1
    ?   ??? dam2
    ??? profes

8 directories, 0 files
```

```
tree /srv/samba/comun/
/srv/samba/comun/
??? dam1
??? dam2
??? departamentos
??? lost+found

4 directories, 0 files
```