

# Tema 4: Servicios básicos de servidor a cliente

## Administración de Sistemas e Redes

Tomás Fernández Peña

[tf.pena@usc.es](mailto:tf.pena@usc.es)

[Página www](#)

[Página inicial](#)



[Página 1 de 97](#)

[Regresar](#)

[Pantalla completa](#)

[Cerrar](#)

[Abandonar](#)

# Índice

- Introducción
- Acceso remoto y transferencia de ficheros
- Servidores de ficheros (NFS)
- Servicios de directorio
- Servicio de directorio: LDAP
- Compartición Linux-Windows: Samba

*Página www*

*Página inicial*



*Página 2 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

Página www

Página inicial



Página 3 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

## 1. Introducción

Dos tipos de servicios:

### 1. Servicios de Internet:

- Servicios de ejecución remota: telnet, ssh
- Servicios de transferencia de ficheros: ftp, sftp
- Servicio de DNS
- Servicio de Proxy
- Servicio de correo electrónico: SMTP, POP, ...
- Servicio Web

### 2. Servicios de intranet

- Sistemas de archivos de red (NFS)
- Servicio de información de red (NIS)
- Servicio de directorio (LDAP)
- Compartición Windows/Linux (Samba)

Los servicios de DNS, Web, Proxy y e-mail se tratan en la asignatura [Administración Avanzada de Sistemas e Redes](#)

[Página www](#)[Página inicial](#)[Página 4 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## 2. Acceso remoto y transferencia de ficheros

Permiten acceder a un sistema remoto y transferir ficheros de/hacia este sistema

- Aplicaciones clásicas
  1. **telnet** (*TELEtype NETwork*) permite conectarnos a otros ordenadores de la red como terminal remoto
  2. **ftp** (*File Transfer Protocol*) permite intercambiar ficheros entre distintos ordenadores de la red
- Problema: la información se transfiere en claro
- El uso de **telnet** y **ftp** se desaconseja
- Reemplazarlos por **ssh**, **scp**, **sftp**
  1. **ssh** (*Secure Shell*) permite conectarnos a otro sistema encriptando toda la información
  2. **scp**, **sftp** permiten la transferencia de ficheros de forma encriptada
    - ★ **scp** similar a **cp** y **sftp** similar a **ftp**

## Servicio de telnet y ftp

Los servicios TCP (telnet, ftp, talk, finger, etc.) son normalmente lanzados por el superdemonio de red `inetd` (o `xinetd`)

- El fichero de configuración es el `/etc/inetd.conf`
- Ejemplo de línea

```
telnet    stream  tcp    nowait  telnetd  /usr/sbin/in.telnetd
```

- ★ cuando `inetd` reciba una petición al puerto `telnet` abre un socket tipo `stream` y ejecuta `fork()` y `exec()` del programa `/usr/sbin/in.telnetd`, bajo la identidad del usuario `telnetd`
- ★ `nowait` indica que el servidor puede continuar procesando conexiones en el socket
- Versión mejorada de `inetd`: `xinetd`
- Para mayor control usar *TCP Wrapper* (programa `tcpd`)
  - ★ Permite conceder/denegar acceso a determinados hosts/redes mediante los fichero `/etc/hosts.allow` y `/etc/hosts.deny`

[Página www](#)[Página inicial](#)[Página 6 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Servicio de telnet

### Instalación de un servidor telnet

- Descargar el paquete `telnetd`
  - ★ El paquete actualiza el `/etc/inetd.conf`
  - ★ Por defecto usa TCP wrappers
  - ★ El servidor escucha el puerto 23

### Desinstalar el servicio telnet

- ★ Desinstalar el paquete `telnetd`, o
- ★ Comentar la línea correspondiente en `/etc/inetd.conf`

Página www

Página inicial



Página 7 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

## Servicio de FTP

Transfiere ficheros a/desde un host remoto

- Permite usuarios registrados o anónimos (*anonymous*)
- Utiliza dos puertos: 21 (conexión de control) y 20 (conexión de datos)
- Dos modos de funcionamiento:
  1. Activo (modo por defecto en el comando ftp)
    - ★ El servidor inicia la conexión de datos desde su puerto 20 a un puerto  $> 1023$  del cliente
    - ★ Problema con los firewalls en el cliente
  2. Pasivo (modo recomendable, por defecto en navegadores)
    - ★ El cliente inicia las conexiones de control y datos
    - ★ No se utiliza el puerto 20
    - ★ No tiene problema con los firewall

[Página www](#)[Página inicial](#)[Página 8 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Instalación de un servidor ftp básico

### 1. Instalar el paquete ftpd

- El paquete actualiza el `/etc/inetd.conf`
- Por defecto usa TCP wrappers
- Podemos denegar el acceso ftp a ciertos usuarios incluyendo los en el fichero `/etc/ftpusers`

[Página www](#)[Página inicial](#)[◀◀](#)[▶▶](#)[◀](#)[▶](#)[Página 9 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Servicio de FTP avanzado

### Servidores avanzados de FTP

- Proporcionan numerosas facilidades, tanto para ftp normal como anónimo
- Existen **numerosos servidores** comerciales u open source: **Wu-FTPd, Pure-FTPd, ProFTPd, wzdftpd, vsftpd**
- Estos servidores proporcionan normalmente:
  - ★ Operación a través de inetd o *standalone*
  - ★ Servidores FTP virtuales (varios servidores de FTP anónimos en el mismo host)
  - ★ Usuarios FTP virtuales (cuentas ftp diferentes de las cuentas del sistema)
  - ★ Facilidades para registro y monitorización de accesos
  - ★ Facilidades para controlar y limitar accesos
  - ★ Comunicación encriptada

[Página www](#)[Página inicial](#)[Página 10 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## SSH

SSH: Shell seguro

- Permite comunicarnos de forma segura con un servidor remoto
  - ★ Permite abrir sesiones o transferir ficheros (`scp` o `sftp`)
  - ★ Reemplazo de `rlogin`, `telnet` o `ftp`
  - ★ Todos los datos viajan encriptados
  - ★ Dos versiones SSH-1 y SSH-2:
    - \* Recomendable SSH-2
    - \* Versión open-source **OpenSSH**
- Paquetes Debian:
  - ★ Cliente: `openssh-client`
  - ★ Servidor: `openssh-server`

## Modos de autenticación mediante SSH

SSH soporta 4 modos de autenticación:

1. Si el nombre del host remoto desde el cual un usuario se conecta al servidor esta listado en `~/.rhosts`, `~/.shosts`, `/etc/hosts.equiv` o `/etc/shosts.equiv` el usuario remoto puede entrar sin contraseña
  - Método absolutamente desaconsejado
2. Igual que el anterior pero la clave pública del host remoto debe aparecer en `/etc/ssh_known_hosts` o `~/.ssh/known_hosts`
  - No demasiado seguro (si el host remoto se ve comprometido, el servidor local queda comprometido)
3. La clave pública del usuario remoto debe estar en `~/.ssh/authorized_keys`
  - El usuario remoto debe tener acceso a su clave privada
  - Método más seguro, pero un poco incomodo
4. Acceso mediante contraseña (modo por defecto)
  - Menos seguro que el anterior

## Opciones para autenticación

Fichero de configuración del servidor ssh: `/etc/ssh/sshd_config`

Opción	M	Dfto.	Significado
RhostsRSAAuthentication	2	no	Si <b>yes</b> permite autenticación por host (SSH-1)
HostbasedAuthentication	2	no	Si <b>yes</b> permite autenticación por host (SSH-2)
IgnoreRhosts	2	yes	No usa los ficheros <code>~/.rhosts</code> y <code>~/.shosts</code>
IgnoreUserKnownHosts	2	no	Ignora el fichero <code>~/.ssh/known_hosts</code>
RSAAuthentication	3	yes	Autenticación de clave pública de usuario (SSH-1)
PubkeyAuthentication	3	yes	Autenticación de clave pública de usuario (SSH-2)
PasswordAuthentication	4	yes	Autenticación mediante contraseña
UsePAM	2,3,4	no	Usa PAM para autenticación

[Página www](#)
[Página inicial](#)


Página 13 de 97

[Regresar](#)
[Pantalla completa](#)
[Cerrar](#)
[Abandonar](#)

## Otras opciones de configuración del servidor

Otras opciones en `/etc/ssh/sshd_config`

Opción	Dfto.	Significado
Port	22	Puerto (puede ser interesante cambiarlo a >1024)
Protocol	2,1	Protocolo aceptado (más seguro sólo 2)
ListenAddress	Todas	Dirección local por la que escucha
PermitRootLogin	yes	Permite acceder al root
X11Forwarding	no	Permite forwarding X11

Para más opciones `man sshd_config`

## Opciones para el cliente

Fichero de configuración del cliente ssh: `/etc/ssh/ssh_config`  
o `~/.ssh/config`

- En este fichero se especifican opciones para los comandos `ssh`, `scp` o `sftp`
- Algunas de estas opciones se pueden especificar en el momento de ejecutar el comando, p.e.

```
$ ssh -p port servidor # Indica otro puerto
```

Algunas opciones:

Opción	Dfto.	Significado
Hosts		Host para los que se aplican las opciones (* implica todos)
Port	22	Puerto por defecto
Protocol	2,1	Protocolo usado por defecto
Cipher[s]		Mecanismos de cifrado usados
ForwardX11	no	Reenvío X11

Para más opciones `man ssh_config` y `man ssh`

## Otros comandos

- **ssh-keygen** generación y gestión de claves públicas/privadas para SSH
  - ★ Permite claves RSA o DSA (DSA sólo SSH-2, por defecto RSA-2)
  - ★ Ficheros (para SSH-2)
    - \* Clave privada: `~/.ssh/id_rsa` o `~/.ssh/id_dsa` (`~/.ssh/identity` para SSH-1)
    - \* Clave pública: `~/.ssh/id_rsa.pub` o `~/.ssh/id_dsa.pub` (`~/.ssh/identity.pub` para SSH-1)
  - ★ La clave privada debe tener una *passphrase* de longitud arbitraria
    - \* Puede cambiarse con la opción `-p`

[Página www](#)[Página inicial](#)[Página 16 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- `ssh-agent` Agente de autenticación
  - ★ Mantiene en memoria la clave privada
  - ★ Evita tener que escribir la *passphrase* cada vez que usemos `ssh`
  - ★ Habitualmente, si entramos en X11 se activa automáticamente
    - \* Opción `use-ssh-agent` de `/etc/X11/Xsession.options` (ver `man xsession.options`)
  - ★ Para activarlo en consola usar (como usuario)  
`eval $(ssh-agent)`
  - ★ Define las variables `SSH_AUTH_SOCK` y `SSH_AGENT_PID`

[Página www](#)[Página inicial](#)[Página 17 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- `ssh-add` Añade las claves privadas al agente

- ★ Uso:

```
ssh-add [opciones] [-t life] [fichero]
```

- ★ Por defecto añade los ficheros `~/.ssh/id_rsa`, `~/.ssh/id_dsa` y `~/.ssh/identity`, pidiendo las correspondientes *passphrases*

- ★ Pueden añadirse múltiples claves

- \* En una conexión se prueban las diferentes claves hasta que coincide

- ★ Algunas opciones:

- \* `-l` Muestra las identidades añadidas
- \* `-t life` Especifica un tiempo de vida de la identidad

### 3. Sistemas de archivos de red (NFS)

NFS (*Network File System*) permite compartir sistemas de ficheros en la red

- Introducido por Sun Microsystems en 1985, y soportado por todos los Unixes
- Versiones principales: NFSv2, NFSv3 y NFSv4 (la más reciente, RFC [3530](#), incluido en kernel 2.6, última revisión 4.1)
- NFSv2 y 3: protocolo sin estado, el servidor no mantiene información de los clientes
- NFSv4 incorpora estado: servidor y clientes mantienen información sobre ficheros abiertos y locks
  - ★ Incorpora un mecanismo complejo de recuperación de caídas
- Comunicación mediante TCP (v3 o v4) o UDP (v2 o v3)

[Página www](#)[Página inicial](#)[Página 19 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

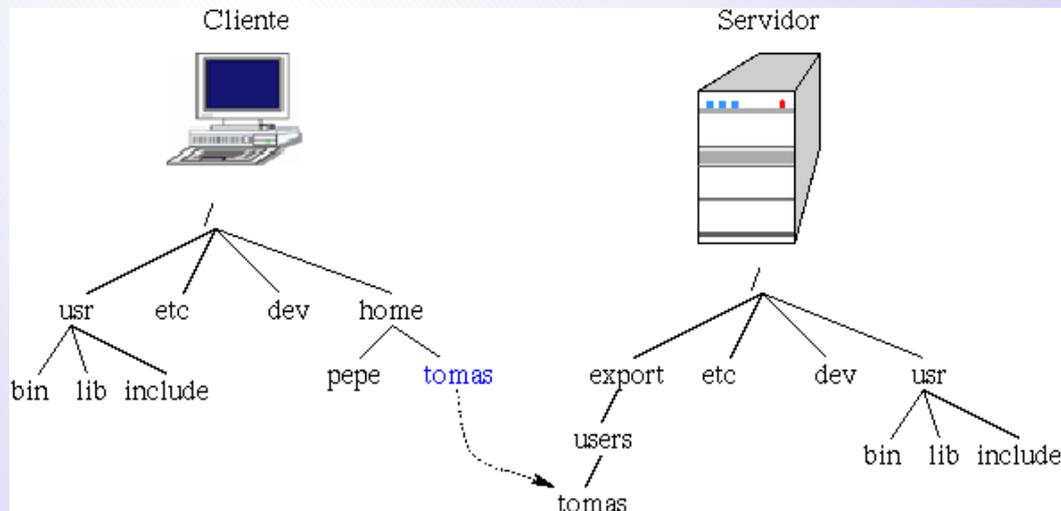
- Dos tipos de servidores en Linux:
  - ★ servidor en espacio de usuario: más lento y con problemas
  - ★ servidor en modo kernel: más rápido, menos características (versión por defecto)

Para más información:

- Capítulo 18 (*The Network File System*) del libro “Unix and Linux System Administration Handbook” (4a ed.), Evi Nemeth et.al.
- [Linux NFS-HOWTO](#)
- [nfs.sourceforge.net](http://nfs.sourceforge.net)

## Características principales

### Ejemplo de funcionamiento



[Página www](#)[Página inicial](#)[Página 21 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Procesos implicados

NFS se basa en RPC (*Remote Procedure Call*)

- el servicio *portmap* (también llamado *rpcbind*) debe estar disponible y activo
  - ★ convierte números de programas RPC en números de puertos
  - ★ utiliza el puerto 111
  - ★ necesario para aplicaciones que usen RPC
  - ★ el comando `rpcinfo` nos muestra información RPC
  - ★ en Debian, paquete `portmap`

## Otros demonios necesarios:

- **rpc.nfsd**: implementa la parte de usuario del servidor NFS (atender y resolver las peticiones de acceso del cliente a archivos situados en el directorio remoto)
- **rpc.mountd**: proceso que recibe la petición de montaje desde un cliente NFS y chequea para mirar si coincide con un sistema de ficheros actualmente exportado, y si el cliente tiene permisos suficientes para montar dicho directorio
- **rpc.rquotad**: proporciona información de cuotas a usuarios remotos
- **rpc.statd**: implementa el protocolo NSM (*Network Status Monitor*); proporciona un servicio de notificación de reinicio, cuando NFS cae; lo usa el servicio de bloqueo de ficheros *lockd*
- **rpc.lockd**: servicio de bloqueo de ficheros (*NFS lock manager*, NLM); no necesario en kernels modernos ( $\geq 2.4$ ) en los que el boqueo es realizado por el kernel

[Página www](#)[Página inicial](#)[Página 23 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

NFSv4 no usa portmap, ni los demonios `rpc.mountd` y `rpc.statd`

- Usa autenticación basada en Kerberos mediante los siguientes servicios:
  - ★ `rpcsec_gss` (cliente `rpc.gssd`, servidor `rpc.svcgssd`): autenticación de la conexión cliente-servidor
  - ★ `rpc.idmapd`: mapeo entre UIDs (o GIDs) y nombres de usuario (o nombres de grupos)

## Instalación de NFS en Debian

Veremos como instalar un servidor y con cliente NFS v3 y v4 en Debian

- Los paquetes a instalar para las dos versiones son los mismos: `nfs-kernel-server` y `nfs-common` (este último suele estar instalado por defecto)
  - ★ `nfs-kernel-server` proporciona `rpc.nfsd`, `rpc.mountd`, y para NFSv4 `rpc.svcgssd`
  - ★ `nfs-common` proporciona `rpc.lockd`, `rpc.statd`, y para NFSv4 `rpc.gssd` y `rpc.idmapd`
- El fichero básico de configuración es el mismo en las dos versiones: `/etc/exports`

## Servidor NFSv3

### 1. Configurar los directorios a exportar: fichero `/etc/exports`

- Ejemplo de fichero `/etc/exports`

```
/projects          (ro) proj*.usc.es(rw,no_subtree_check)
```

```
/home             193.144.84.0/24(rw,no_subtree_check,root_squash)
```

```
/pub              (ro,all_squash)
```

- exporta `/projects` de sólo lectura para todo el mundo y lectura/escritura para los sistemas `proj*.usc.es`

- Algunas opciones de la exportación:
  - ★ `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
  - ★ `squash_uids/squash_gids` especifica una lista de UIDs o GIDs que se deberían trasladar al usuario anónimo  
`squash_uids=0-15,20,25-50`
  - ★ `anonuid/anongid` fija el UID/GID del usuario *anónimo* (por defecto 65534)
  - ★ `subtree_check/no_subtree_check` con `subtree_check`, si se exporta un subdirectorio (no un filesystem completo) el servidor comprueba que el fichero solicitado por el cliente esté en el subdirectorio exportado; con `no_subtree_check` (opción por defecto) se deshabilita ese chequeo

[Página www](#)[Página inicial](#)[Página 27 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- ★ **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
- Para más opciones `man exports`
  - Cada vez que se modifica este fichero se debe ejecutar el comando `exportfs` para actualizar el servidor
    - # `exportfs -ra`
    - ★ ver `man exportfs` para opciones del comando

[Página www](#)[Página inicial](#)[Página 28 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

2. Iniciar el demonio:

```
# service nfs-kernel-server start
```

3. Comprobar los directorios exportados con `showmount`

```
# showmount --exports localhost
```

- `showmount` muestra información de un servidor NFS: directorios que exporta, directorios montados por algún cliente y clientes que montan los directorios

4. Podemos ver las estadísticas del servidor NFS con `nfsstat`

[Página www](#)[Página inicial](#)[◀](#)[▶](#)[◀](#)[▶](#)[Página 29 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Servidor NFSv4

### 1. Configuración de directorios y fichero export

- Los exports de NFSv4 deben residir en un pseudodirectorio, donde los directorios reales a exportar se montan con la opción `--bind`, por ejemplo para exportar `/home`

```
# mkdir /export
# mkdir /export/home
# mount --bind /home /export/home
```

- La opción `bind` permite remontar un directorio en otro sitio

- ★ Para que este montado permanezca, añadir al `fstab` la siguiente línea:

```
/home    /export/home    none    bind    0
0
```

Página *www*

Página inicial

Página *30* de *97*

Regresar

Pantalla completa

Cerrar

Abandonar

- Fichero `/etc/exports` en NFSv4

```
/export          193.144.84.0/24(rw,fsid=0,crossmnt,no_subtree_check)
/export/home     193.144.84.0/24(rw,no_subtree_check)
```

- Nuevas opciones de la exportación:

- ★ `fsid=0` designa este path como la raíz de los directorios exportados por NFSv4
- ★ `crossmnt` permite que los directorios debajo del raíz se muestren adecuadamente (alternativamente, se puede poner la opción `nohide` en cada uno de esos directorios)

2. Iniciar el demonio y comprobar que funciona igual que en el caso de NFSv3.

## Cliente NFS

El cliente NFS en Linux está integrado en el nivel del Sistema de Ficheros Virtual (VFS) del kernel

- no necesita un demonio particular de gestión (en otros UNIX, demonio *biod*)

Instalación:

1. Instalar (si no está ya instalado) el paquete `nfs-common`
2. Montar los directorios remotos con `mount -t nfs` (para v3) o `mount -t nfs4` (para v4), o añadir una entrada en `fstab` (ver **Tema 3: Montado de los sistemas de ficheros**). NOTA: en versiones actuales de Linux la opción `-t nfs` intenta montar con NFSv4 y si falla, pasa a NFSv3.

- Ejemplo de uso con `mount` (IP servidor NFS 193.144.84.1):

```
# mount -t nfs4 193.144.84.1:/home /mnt/home
```

- Ejemplo de entrada en `fstab`

```
193.144.84.1:/home      /home      nfs4      rw,auto
```

[Página www](#)[Página inicial](#)[Página 32 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- Automount se usa frecuentemente con NFS (ver la parte de Autofs en [Tema 3: Montado de los sistemas de ficheros: Autofs](#))

Opciones particulares de montado con NFS:

- **rsize= $n$ /wsize= $n$**  especifican el tamaño del datagrama utilizado por los clientes NFS cuando realizan peticiones de lectura/escritura (pueden ajustarse para optimizar)
- **hard** el programa accediendo al sistema de ficheros remoto se colgará cuando el servidor falle; cuando el servidor esté disponible, el programa continuará como si nada (opción más recomendable)
- **soft** cuando una petición no tiene respuesta del servidor en un tiempo fijado por **timeo= $t$**  el cliente devuelve un código de error al proceso que realizó la petición (puede dar problemas)

Para ver más opciones, ver `nfs(5)`

[Página www](#)[Página inicial](#)[Página 33 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Consideraciones de seguridad en NFS

NFS no fue diseñado pensando en la seguridad:

- Los datos se transmiten en claro
- Usa el UID/GID del usuario en el cliente para gestionar los permisos en el servidor:
  - ★ El usuario con UID  $n$  en el cliente obtiene permisos de acceso a los recursos del usuario con UID  $n$  en el servidor (aunque sean usuarios distintos)
  - ★ Un usuario con acceso a root en un cliente podría acceder a los ficheros de cualquier usuario en el servidor (no a los de root, si se usa la opción `root_squash`)

Página www

Página inicial

◀ ▶

◀ ▶

Página 34 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

## Precauciones básicas:

1. Usar NFS sólo en Intranets seguras, donde los usuarios no tengan acceso de administrador en sus sistemas
2. Evitar el acceso a NFS desde fuera de la Intranet
  - Bloquear los puertos TCP/UDP 111 (*portmap*) y 2049 (*nfs*)
3. Usar **NFSv4 con Kerberos**, que incluye autenticación de hosts y cifrado
4. Usar versiones seguras de NFS (**Secure NFS**) u otros sistemas de ficheros (p.e. **Self-certifying File System**, SFS)

Ver **NFS howto** y [http://www.cert.org/tech\\_tips/unix\\_configuration\\_guidelines.html#A](http://www.cert.org/tech_tips/unix_configuration_guidelines.html#A)

[Página www](#)[Página inicial](#)[Página 35 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## 4. Servicios de directorio

Necesidad de mantener una configuración única a través de múltiples sistemas

- Compartición de los ficheros de configuración
- Ficheros a compartir:
  - ★ `/etc/passwd`, `/etc/shadow`, `/etc/group`, etc.
- Mecanismos de compartición:
  - ★ Copia de ficheros de un servidor central al resto de los equipos mediante `rdist` o `rsync`
  - ★ Utilización de un servidor de dominio, que centralice esa información
    - \* NIS: Network Information Service
    - \* LDAP: Lightweight Directory Access Protocol

[Página www](#)[Página inicial](#)[Página 36 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Concepto de dominio

**Dominio** conjunto de equipos interconectados que comparten información administrativa (usuarios, grupos, contraseñas, etc.) centralizada

- Necesidad de uno (o varios) servidores que almacenen físicamente dicha información y que la comunique al resto cuando sea necesario
- Normalmente se usa un esquema cliente/servidor
  - ★ p.e. un usuario se conecta en un sistema cliente y este valida las credenciales del usuario en el servidor
- En Windows 2000, la implementación del concepto de dominio se realiza mediante el denominado Directorio Activo (*Active directory*)
  - ★ Basado en LDAP y DNS
- En UNIX, el servicio clásico de gestión de dominios es NIS
  - ★ NIS se considera bastante obsoleto

- Existen implementaciones libre del protocolo LDAP para Unix (openLDAP)
  - ★ más potente y escalable que NIS para la implementación de dominios

[Índice](#)[Introducción](#)[Acceso remoto y...](#)[Sistemas de...](#)[Servicios de directorio](#)[Servicio de...](#)[Compartición...](#)[Página www](#)[Página inicial](#)[Página 37 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

[Página www](#)[Página inicial](#)[Página 38 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Servicio de Información de Red NIS

Desarrollado por Sun Microsystem en los años 80

- Nombre original *Yellow Pages* (modificado por razones legales)
- Muy popular como sistema de administración de dominios en UNIX
- A principios de los años 90, versión NIS+ para Solaris 2.x
  - ★ muy diferente de NIS
  - ★ incorpora soporte para encriptación y autenticación de datos
  - ★ complejo y poco soportado
- Para más información:
  - ★ [The Linux NIS\(YP\)/NYS/NIS+ HOWTO](#)
  - ★ [Debian NIS HOWTO](#)
  - ★ [Introducción a NIS y NFS](#)

[Página www](#)[Página inicial](#)[Página 39 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Funcionamiento básico de NIS

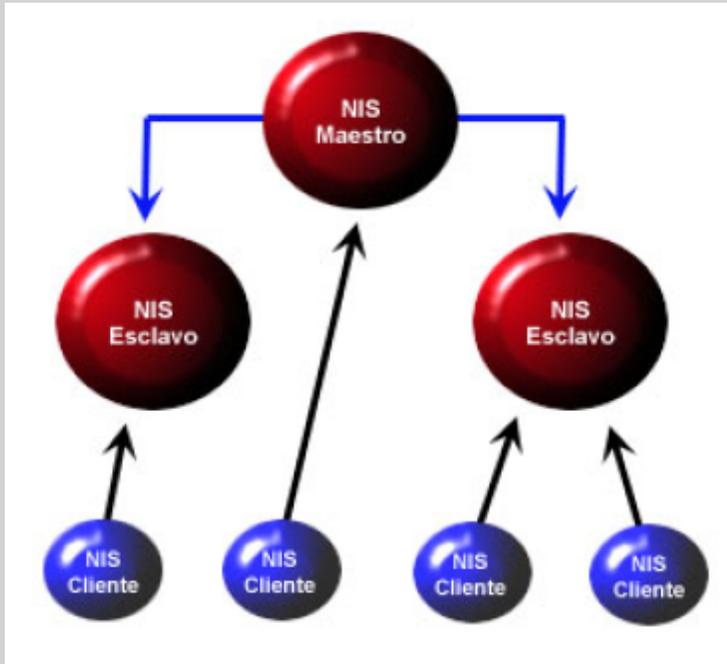
### Base de datos distribuida

- Un servidor (*master*) mantiene los ficheros de configuración de los sistemas (`/etc/passwd`, `/etc/group`, etc.)
  - ★ cada archivo de configuración se convierte en una o más tablas (*mapas* NIS) de una base de datos
  - ★ esos mapas se guardan en un formato binario llamado DBM (*DataBase Management*)
  - ★ el servidor NIS maestro debería tener ambas, las tablas ASCII y las DBM
  - ★ en una red debe haber al menos una máquina actuando como un servidor NIS maestro
- Los clientes hablan directamente con el servidor NIS para leer la información almacenada en sus bases de datos DBM

[Página www](#)[Página inicial](#)[Página 40 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- Pueden existir servidores NIS esclavos:
  - ★ tienen copias de las bases de datos NIS
  - ★ reciben estas copias del servidor NIS maestro cada vez que se realizan cambios a las bases de datos maestras
- Un servidor maestro y sus servidores esclavos y clientes constituyen un dominio NIS
  - ★ una red puede tener múltiples servidores NIS, cada uno sirviendo a un dominio NIS diferente

## Esquema de un dominio NIS



[Página www](#)[Página inicial](#)[Página 42 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Comandos básicos de NIS

NIS incluye un conjunto amplio de comandos y demonios, algunos de los cuales son:

- `ypserv` demonio de servidor
- `ypbind` demonio de cliente
- `domainname` establece el nombre del dominio
- `ypinit` configura un servidor como maestro o esclavo
- `ypxfr` descarga un mapa desde servidor maestro (en los esclavos)
- `yppush` ejecutado en el maestro, hace que los esclavos actualicen sus mapas
- `ypwhich` muestra el nombre del servidor NIS
- `ypcat` muestra las entradas de un mapa
- `yppasswd` cambia la contraseña en la base de datos de NIS

- ypchfn cambia el campo GECOS en la base de datos de NIS
- ypchsh cambia el login shell en la base de datos de NIS

[Página www](#)[Página inicial](#)[Página 43 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

[Página www](#)[Página inicial](#)[Página 44 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Instalación de NIS en Debian

El proceso de puesta en marcha de NIS depende de la distribución

- veremos como instalar un servidor maestro y un cliente en Debian

### Servidor maestro

#### 1. Instalar el paquete `nis`

(a) Indicar un nombre de dominio NIS (que no tiene que corresponder con el dominio de RED)

- El nombre de dominio puede cambiarse con `domainname`

(b) La configuración puede tardar, ya que intenta iniciarse como cliente NIS y se queda buscando un servidor

#### 2. Cambiar el fichero `/etc/default/nis`

- debemos poner `NISSERVER=master`

3. En el fichero `/etc/ypserv.securenets` añadir el número de la red local, para permitir acceso exclusivo a los sistemas de esa red

255.255.255.0

192.168.0.0

4. Ejecutar `/usr/sbin/ypserv` para iniciar el servidor
5. Editar (si es necesario) el fichero `/var/yp/Makefile`
  - permite configurar características generales así como las tablas a partir de las cuales se crean los mapas NIS
  - haciendo **make** en ese directorio se crean los mapas que se guardan en `/var/yp/dominio`
  - cada vez que se modifique alguna tabla (p.e. añadiendo un nuevo usuario), debemos hacer **make** para actualizar los mapas NIS
6. Ejecutar `/usr/lib/yp/ypinit -m` para que el sistema se configure como servidor maestro
  - no añadir ningún servidor NIS más (Ctrl-D)
7. Podemos comprobar que funciona bien haciendo un `ypcat` de alguno de los mapas (p.e. `ypcat passwd`)
8. Por último, el servidor debe configurarse también como cliente

Página *www*

Página inicial



Página 45 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

- seguir los pasos de la siguiente sección

*Índice*

*Introducción*

*Acceso remoto y...*

*Sistemas de...*

*Servicios de directorio*

*Servicio de...*

*Compartición...*

*Página www*

*Página inicial*



*Página 46 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

[Página www](#)[Página inicial](#)[Página 47 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Cliente NIS

1. Eliminar de los ficheros locales los usuarios, grupos y otra información que queramos que sea accesible por NIS (sólo en los clientes)
2. Instalar el paquete `nis` e indicar el nombre del dominio NIS
3. Si se desea, cambiar `/etc/yp.conf` para especificar el servidor NIS concreto
  - por defecto, busca el servidor mediante un broadcast
4. Modificar el archivo `/etc/nsswitch.conf` para que busque `passwd`, `group` y `shadow` por NIS:

```
passwd:      files nis
group:       files nis
shadow:      files nis
```

- El formato y opciones de ese fichero lo vimos en el **Tema 5: Ficheros de configuración de red**

- Alternativamente, se puede dejar el modo `compat` añadiendo al final de los ficheros `passwd`, `shadow` y `group` del cliente un `+`, para indicar que vamos a usar NIS
- este método permite incluir/excluir determinados usuarios
- ver [NIS-HOWTO: Setting up a NIS Client using Traditional NIS](#)

*Página www*

*Página inicial*



*Página 48 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

[Página www](#)[Página inicial](#)[Página 49 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Fichero `/etc/netgroup`

NIS introduce el concepto de *netgroups*

- grupos de usuarios, máquinas y redes que pueden ser referenciadas como un conjunto
- se definen en el fichero `/etc/netgroup`, en principio, sólo en el servidor NIS maestro

Formato de una entrada en `netgroup`

```
netgroup_name (host, user, NIS_domain), ...
```

- `host` nombre de una máquina en el grupo
- `user` nombre de login de un usuario de la máquina `host`
- `NIS_domain` dominio NIS nombre del dominio NIS

Pueden dejarse entradas en blanco o con un guión:

- Una entrada en blanco implica cualquier valor, p.e.
  - ★ (doc19, , ) indica todos los usuarios del host doc19
- Una entrada con un guión (-) implica campo sin valor, p.e.
  - ★ (-, pepe, ) indica el usuario pepe y nada más

Ejemplo de un fichero /etc/netgroup

```

sysadmins      (-,pepe,) (-,heidis,) (-,jnguyen,) (-,mpham,)
servers        (numark,-,) (vestax,-,)
clients        (denon,-,) (technics,-,) (mtx,-,)
research       (-,boson,) (-,jyom,) (-,weals,) (-,jaffe,)
allusers       sysadmins research
allhosts       servers clients
    
```

[Página www](#)[Página inicial](#)[Página 51 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

Estos *netgroups* pueden usarse en varios ficheros del sistema para definir permisos:

- con NIS en modo `compat`, p.e. añadiendo `+@sysadmins` en `/etc/passwd` daríamos permiso de acceso a los usuarios definidos como `sysadmins`
- en el fichero `/etc/exports`, para indicar grupos de máquinas a las que exportar un directorio por NFS

```
/home          allhosts(rw,root_squash,sync)
```

- en los ficheros `/etc/hosts.allow` y `/etc/hosts.deny` de los TCP Wrappers
- etc.

[Página www](#)[Página inicial](#)[Página 52 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## 5. Servicio de directorio: LDAP

LDAP: Protocolo Ligero de Acceso a Directorios (*Lightweight Directory Access Protocol*)

- Protocolo de red para consulta y modificación de datos de directorios X.500
  - ★ X.500: Estándares de la ITU-T para servicios de directorio
  - ★ Define, entre otros, un protocolo de acceso a directorios llamado DAP (*Directory Access Protocol*)
  - ★ DAP definido sobre la pila completa de niveles OSI: costoso y complejo
- LDAP es una alternativa ligera al protocolo DAP
  - ★ Opera directamente sobre TCP/IP
  - ★ Actualmente, la mayoría de servidores de directorio X.500 incorporan LDAP como uno de sus protocolo de acceso

[Página www](#)[Página inicial](#)[Página 53 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- Diferentes implementaciones del protocolo LDAP

- ★ Microsoft Server Active Directory
- ★ NetIQ eDirectory
- ★ Oracle Internet Directory
- ★ IBM Security Directory Server
- ★ Apache Directory Server
- ★ 389 Directory Server
- ★ Red Hat Directory Server
- ★ OpenLDAP

- Más información sobre LDAP

1. LDAP Linux HOWTO
2. Sección 19.3 del libro “Unix and Linux System Administration Handbook” (4a ed.)
3. IBM RedBooks: Understanding LDAP - Design and Implementation
4. Recursos, ayudas, ... : [ldapman.org](http://ldapman.org)

[Página www](#)[Página inicial](#)[Página 54 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## OpenLDAP

- Implementación *open source* del protocolo LDAP
- Basado en software desarrollado en la [Universidad de Michigan](#)
- Incluye cuatro componentes principales
  - ★ [slapd](#) - demonio LDAP stand-alone (servidor)
  - ★ [slurpd](#) - demonio de replicación y actualización de LDAP
  - ★ [librerías](#) que implementan el protocolo LDAP
  - ★ utilidades, herramientas, y clientes básicos
- Más información sobre la configuración de OpenLDAP en el [OpenLDAP Administrator's Guide](#)

[Página www](#)[Página inicial](#)[Página 55 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Modelo de datos de LDAP

Un directorio es una base de dato optimizada para lectura, navegación y búsqueda

- la información se almacena de manera jerárquica
- generalmente no se soportan transacciones complejas ni sistemas de recuperación
- las actualizaciones son cambios simples
- proporcionan respuestas rápidas a grandes volúmenes de búsquedas
- el directorio puede estar replicado y/o distribuido entre varios sistemas (p.e. DNS)

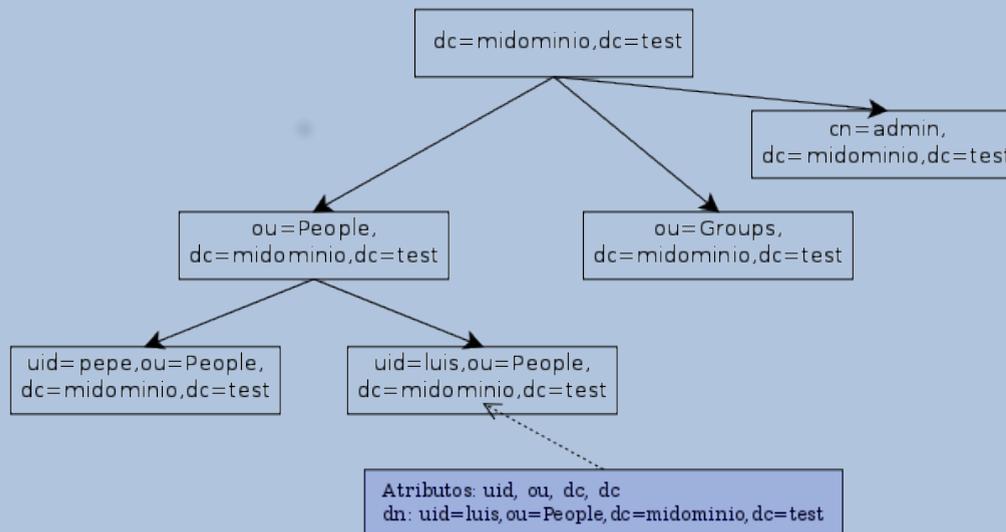
[Página www](#)[Página inicial](#)[Página 56 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

LDAP organiza el directorio como una estructura jerárquica de entradas (nodos) en forma de árbol

- Cada entrada posee un conjunto de atributos, que pueden ser de diferentes tipos
  - ★ cada atributo se identifica por su tipo y uno o más valores
  - ★ los tipos son normalmente palabras nemotécnicas, como **uid** (identificador de usuario), **cn** (*common name*), **c** (*country*), **dc** (*domain component*), etc.
  - ★ los diferentes atributos de un nodo están determinados por la clase a la que pertenece
  - ★ las clases permiten definir entradas con diferente información: clases para personas, para equipos, administrativas, etc.
  - ★ las clases se definen mediante ficheros de *esquema* (*schema*)

- Cada nodo debe poseer un nombre único: *nombre distinguido* o *dn (distinguished name)*
  - ★ el *dn* identifica de forma unívoca cada objeto en la base de datos

Ejemplo: árbol de usuarios y grupos en LDAP, basado en nombres de dominios de Internet:



[Página www](#)[Página inicial](#)[Página 58 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- cada nodo puede tener varios atributos, p.e. el nodo *uid=pepe* podría tener los siguientes atributos:

```
dn: uid=pepe,ou=People,dc=midominio,dc=test
objectClass: account
cn: Jose Pena
sn: Pena
description: alumno
mail: pepe@midominio.test
```

- el formato en el que se muestran los atributos del objeto se denomina LDIF (*LDAP Data Interchange Format*)
  - ★ formato de intercambio de datos para importar y exportar datos a un servidor LDAP

[Página www](#)[Página inicial](#)[Página 59 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Instalación de un servidor LDAP

Describiremos como montar un servidor LDAP simple que nos permita la gestión de usuarios y grupos

- el servidor mantendrá la lista de usuarios y grupos del dominio
- en los clientes la autenticación de usuarios y los permisos se basará en el servidor LDAP

### Pasos para la instalación del servidor en Debian

1. Instalar los paquetes `slapd` (servidor OpenLDAP) y `ldap-utils` (utilidades del paquete OpenLDAP: `ldapsearch`, `ldapadd`)
  - En la configuración, elegir una contraseña para el administrador del LDAP (no tiene que ser la contraseña de root del sistema)

[Página www](#)[Página inicial](#)[Página 60 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- El comando `slapcat` permite ver la estructura creada
  - ★ comprobar que la estructura creada por defecto es correcta
- Podemos hacer búsquedas sencillas con `ldapsearch`

```
# ldapsearch -x -b dc=midominio,dc=test cn=admin
```



## 2. Ficheros de configuración:

### (a) Fichero de opciones del demonio `/etc/default/slapd`

- Permite, entre otras cosas, especificar las interfaces donde se desea que escuche ldap (por defecto, todas las interfaces usando TCP puerto 389, URI `ldap://389`)
- La variable `SLAPD_SERVICES` indica los mecanismos de escucha de slapd
  - ★ Puede aceptar conexiones estándar (`ldap://`), conexiones seguras con **SASL** (*Simple Authentication and Security Layer*), usando `ldaps:///` o peticiones realizadas desde sockets UNIX (`ldapi:///`)
- Indicar que acepte conexiones estándar en la red del cliente (`ldap://172.25.25.1:389/`), y reiniciar el servicio (`service slapd restart`)

### (b) Fichero de configuración del servidor `/etc/ldap/sldap.conf`

- Fichero de configuración del demonio `slapd` (en principio, no es necesario cambiarlo)
- En versiones recientes, cambiado por ficheros LDIF

[Página www](#)[Página inicial](#)[Página 62 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

en el directorio `/etc/ldap/slapd.d`

- Para más info, ver `man slapd.conf`

(c) Fichero `/etc/ldap/ldap.conf`

- Fichero de configuración global para los clientes LDAP
- Permite especificar la base por defecto, el servidor LDAP, etc. (cambiarlo para poner nuestra configuración, comprobar que tiene permisos 644)
- Para más info, ver `man ldap.conf`

[Página www](#)[Página inicial](#)[Página 63 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

3. Crear la estructura de la base de datos: crearemos los nodos de **People** y **Group** del árbol LDAP

(a) Crear el siguiente fichero **estructura.ldif** en formato LDIF:

```
dn: ou=People,dc=midominio,dc=test
objectClass: top
objectClass: organizationalUnit
ou: People
```

```
dn: ou=Group,dc=midominio,dc=test
objectClass: top
objectClass: organizationalUnit
ou: Group
```

[Página www](#)[Página inicial](#)[Página 64 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

(b) Añadir los nodos a la base de datos:

```
# ldapadd -x -D 'cn=admin,dc=midominio,dc=test'  
-W -f estructura.ldif
```

- -x autenticación simple sin SASL
- -D nombre distinguido con el que nos conectamos a LDAP (ponemos el del administrador)
- -W pide la contraseña de forma interactiva
- -f fichero a cargar

*Página www*

*Página inicial*



*Página 65 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

#### 4. Añadir un usuario y un grupo a la base de datos

- (a) Crear un fichero como este, que tiene la información para un usuario y un grupo:

```
dn: uid=pepe,ou=People,dc=midominio,dc=test
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
uid: pepe
cn: Jose Pena
uidNumber: 2000
gidNumber: 2000
homeDirectory: /home/pepe
loginShell: /bin/bash
gecos: Jose Pena, Despacho 22,,
```

```
dn: cn=pepe,ou=Group,dc=midominio,dc=test
objectClass: top
objectClass: posixGroup
cn: pepe
gidNumber: 2000
```

- define un usuario pepe y un grupo pepe con información similar a la aparece en el fichero `/etc/passwd`
- puede añadirse más información, como la que aparece en el fichero `/etc/shadow`: campos `shadowMax`, `shadowExpire`, `shadowWarning`, etc.

(b) Añadir el fichero con:

```
# ldapadd -x -D 'cn=admin,dc=midominio,dc=test'
-W -f user-group.ldif
```

(c) Probarlo haciendo búsquedas, tipo:

```
# ldapsearch -x uid=pepe
```

(d) Añadir una contraseña al usuario: puede hacerse directamente metiendo un campo `userPassword` en el fichero anterior, pero es preferible hacerlo mediante el comando `ldappasswd`

```
# ldappasswd -x 'uid=pepe,ou=People,dc=midominio,dc=test'
-D 'cn=admin,dc=midominio,dc=test' -W -S
```

5. Por último, también podemos querer instalar el servidor como cliente, por lo que debemos seguir los pasos indicados en la sección de instalación de un cliente

*Índice*

*Introducción*

*Acceso remoto y...*

*Sistemas de...*

*Servicios de directorio*

*Servicio de...*

*Compartición...*

*Página www*

*Página inicial*



*Página 67 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

Página www

Página inicial

◀

▶

◀

▶

Página 68 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

## Migración desde ficheros o NIS

Si tenemos un sistema configurado mediante ficheros (`passwd`, `shadow`, etc.) o NIS, migrar a LDAP puede ser laborioso

- los scripts del paquete *MigrationTools* son de gran ayuda

## Migración desde ficheros

Pasos para migrar los usuarios definidos en `/etc/passwd` a LDAP

1. Instalar el paquete `migrationtools`
2. Modificar el fichero `/usr/share/migrationtools/migrate_common.ph` para indicar el dominio y la base LDAP

```
$DEFAULT_MAIL_DOMAIN = "midominio.test"
```

```
$DEFAULT_BASE = "dc=midominio,dc=test"
```

3. Descomentad y poned los valores adecuados en las variables `$IGNORE_UID...` y `$IGNORE_GID...` para que no considere los usuarios y grupos del sistema

4. Utilizar los diferentes scripts del directorio `/usr/share/migrationtools` para incorporar la información del sistema al directorio

- `migrate_base.pl` genera entradas correspondientes a las unidades organizativas por defecto: **People**, **Group**, **Hosts**, etc. (ya lo tenemos)
- Otros scripts generan entradas asociadas a diferentes ficheros del sistema

Script	Migra
<code>migrate_passwd.pl</code>	<code>/etc/passwd</code> y <code>/etc/shadow</code>
<code>migrate_group.pl</code>	<code>/etc/group</code>
<code>migrate_hosts.pl</code>	<code>/etc/hosts</code>
<code>migrate_fstab.pl</code>	<code>/etc/fstab</code>
...	

- Estos scripts generan ficheros LDIF que podemos añadir a la base LDAP con `ldapadd`

[Página www](#)[Página inicial](#)[Página 70 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- Ejemplo: migración de los usuarios

- ★ crear el fichero ldif con los usuarios con el comando

```
# ./migrate_passwd.pl /etc/passwd ./passwd.ldif
```

- ★ añadir el fichero generado a la base LDAP con

```
# ldapadd -x -D 'cn=admin,dc=midominio,dc=test'  
-W -f ./passwd.ldif
```

Página www

Página inicial



Página 71 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

## Instalación de un cliente LDAP

Describiremos como como configurar un cliente para que acceda a la información almacenada en el directorio de LDAP del servidor

- Tres pasos:
  1. Indicar en el fichero `/etc/ldap/ldap.conf` la información sobre el servidor LDAP y el URI
  2. Instalar y configurar el *Name Service Switch* (fichero `/etc/nsswitch.conf`)
  3. Instalar y configurar el módulo de autenticación (PAM, *Pluggable Authentication Modules*)
- Los módulos necesarios para esta configuración pueden descargarse de la página de PADL ([www.padl.com](http://www.padl.com)) o directamente como paquetes Debian
  - ★ Descargar los paquetes `libnss-ldap`, `libpam-ldap` (como dependencia, se instala también el paquete `nscd`)
  - ★ En la configuración indicar como URI `ldap://ip_del_servidor/`, el DN de la base del directorio LDAP, versión 3 de

LDAP, y el password del administrador LDAP

- ★ Estas configuraciones se guardan en los ficheros `/etc/libnss-ldap.conf` y `/etc/pam_ldap.conf` (ver `man libnss-ldap.conf` y `man pam_ldap.conf`)
- ★ En la instalación nos pide la clave del administrador de LDAP:
  - \* esta clave se guarda en plano en los ficheros `/etc/pam_ldap.secret` y `/etc/libnss-ldap.secret`, y se usa para que el `root` del sistema pueda modificar directamente las contraseñas de los usuarios
  - \* si no se quiere tener ese fichero con la clave, no indicar ninguna clave cuando nos la pide y en los ficheros `/etc/libnss-ldap.conf` y `/etc/pam_ldap.conf` comentar la línea que empieza por `rootbinddn`
- En algunas distros (RedHat) existe la herramienta `authconfig` que facilita esta configuración

Página [www](#)

Página inicial



Página 73 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

## Configurar el *Name Service Switch*

El NSS se encarga, entre otras, de realizar la correspondencia entre los números y nombres de usuario

- permite gestionar los permisos de acceso de usuarios a ficheros
- se configura a través del fichero `/etc/nsswitch.conf`
- la configuración de ese fichero la vimos en el [tema 3](#)

Pasos (después de haber instalado el paquete `libnss-ldap`):

1. Modificar las líneas de `passwd`, `group` y `shadow` del fichero `nsswitch.conf`

```
passwd:    files ldap
group:     files ldap
shadow:    files ldap
```

- esto indica al NSS que busque la información primero en los ficheros y, si no la encuentra, en el servidor LDAP

2. Probar que funciona:

- (a) Crear el directorio `/home/pepe` y cambiarle el propietario y grupo a `pepe`
- (b) Hacer un `su - pepe` para ver que podemos cambiar al usuario `pepe` en el cliente

*Página www*

*Página inicial*



*Página 74 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

[Página www](#)[Página inicial](#)[Página 75 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Configurar el módulo de autenticación

Aunque el usuario es válido en el cliente, todavía no podemos autenticarnos (entrar en la cuenta)

- Debemos configurar el servicio PAM

**PAM** (*Pluggable Authentication Module*) biblioteca de autenticación genérica que cualquier aplicación puede utilizar para validar usuarios, utilizando por debajo múltiples esquemas de autenticación alternativos (ficheros locales, Kerberos, LDAP, etc.)

- permite añadir nuevos mecanismos de autenticación (Kerberos, LDAP,...) sin tener que modificar los servicios de entrada al sistema como **login**, **ftp**, **ssh**, etc.

*Página www*

*Página inicial*



*Página 76 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

- PAM utiliza módulos que proporcionan autenticación en los servicios de entrada al sistema:
  - ★ Módulos de autenticación (**auth**): comprobación de contraseñas (utilizado por el proceso de **login** para averiguar si las credenciales tecleadas por el usuario (nombre y contraseña) son correctas)
  - ★ Módulos de cuentas (**account**): controlan que la autenticación sea permitida (que la cuenta no haya caducado, que el usuario tenga permiso de iniciar sesiones a esa hora del día, etc.)
  - ★ Módulos de contraseña (**password**): permiten cambiar contraseñas
  - ★ Módulos de sesión (**session**): configuran y administran sesiones de usuarios (tareas adicionales que son necesitadas para permitir acceso, como el montaje de directorios, actualización del fichero **lastlog**, etc.)



- Las librerías de PAM están en `/lib/security` y los ficheros de configuración en `/etc/pam.d/`
- Existe un fichero de configuración para cada servicio que usa PAM
- También existen ficheros comunes que son incluidos por los ficheros de configuración: `common-auth`, `common-account`, `common-password` y `common-session`
- Versiones recientes de PAM gestionan estos ficheros mediante el comando `pam-auth-update` y los ficheros en `/usr/share/pam-configs`

[Página www](#)[Página inicial](#)[Página 78 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

**Configuración de PAM para usar LDAP** el proceso de instalación del paquete `libpam-ldap` ya modifica de forma adecuada los ficheros de configuración de PAM, pero da algunos problemas

1. En el fichero `/usr/share/pam-configs/ldap`, borra la opción `use_authok` (con esta opción, no permite que los usuarios cambien su contraseña) y ejecuta `pam-auth-update`

Página www

Página inicial



Página 79 de 97

Regresar

Pantalla completa

Cerrar

Abandonar

- Si queremos crear directorios home “al vuelo” (el home del usuario se crea al entrar por primera vez en su cuenta), crear el fichero `/usr/share/pam-configs/my_mkhome` con el siguiente contenido

```
Name: activate mkhome
```

```
Default: yes
```

```
Priority: 900
```

```
Session-Type: Additional
```

```
Session:
```

```
        required        pam_mkhome.so umask=0022 skel=/etc/skel
```

y ejecutar `pam-auth-update` (Nota: el directorio solo se va a crear cuando el usuario se conecta en servidor)

- Probar que funciona:
  - Entrar en la cuenta como un usuario LDAP
  - Cambiar la contraseña del usuario

**Paquete `nscd`** Se trata del *Name Service Cache Daemon*

- Hace caché para los nombres leídos del servidor LDAP para aumentar la eficiencia

*Índice*

*Introducción*

*Acceso remoto y...*

*Sistemas de...*

*Servicios de directorio*

*Servicio de...*

*Compartición...*

*Página www*

*Página inicial*



*Página 80 de 97*

*Regresar*

*Pantalla completa*

*Cerrar*

*Abandonar*

[Página www](#)[Página inicial](#)[Página 81 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Configuración de LDAP con múltiples servidores

Podemos configurar varios servidores LDAP que mantengan imágenes sincronizadas de la información del directorio

- equilibran la carga de las consultas, y mejora la tolerancia a fallos

Esquema de maestro único y múltiples esclavos

- el *maestro* mantiene la copia principal sobre la que se hacen los cambios
  - ★ si un cliente intenta hacer un cambio en un esclavo, este lo redirige automáticamente al maestro
- cada vez que se produce un cambio en el directorio del maestro, el servicio **slapd** escribe dicho cambio, en formato LDIF, en un fichero de log
- el demonio **slurpd** lee dichos cambios e invoca las operaciones de modificación correspondientes en todos los esclavos

[Página www](#)[Página inicial](#)[Página 82 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

Para configurarlo debemos hacer que el maestro y los esclavos partan de un estado de directorio común

- copiar manualmente la base de datos LDAP del maestro a todos los esclavos

En el maestro y en los esclavos debemos modificar el fichero `/etc/ldap/slapd.conf`

- Maestro:

- ★ añadir una directiva `replica` por cada esclavo, donde se indique el nombre del esclavo y una cuenta con permiso de escritura en el LDAP del esclavo (`bindn`)

```
replica uri=ldap://esclavo.midominio.test:389
        binddn="cn=Replicator,dc=midominio,dc=test"
        bindmethod=simple credentials=CONTRASEÑA_PLANA
```

- ★ indicar el fichero de log donde se guardan los cambios

```
relogfile /var/lib/ldap/relog
```

[Página www](#)[Página inicial](#)[Página 83 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

- Esclavo:

- ★ añadir una línea `updatedn` indicando la cuenta con la que el servicio `slurpd` del servidor maestro va a realizar las modificaciones en la réplica del esclavo
- ★ esa cuenta debe tener permisos de escritura en la base de datos del esclavo, y debe coincidir con la indicada en el campo `binddn` del maestro
- ★ añadir una línea `updateref` para indicar al servidor esclavo que cualquier petición directa de modificación que venga de un cliente debe ser redireccionada al servidor maestro

```
updatedn    "cn=Replicator,dc=midominio,dc=test"  
updateref  ldap://maestro.midominio.test
```

Para más detalles ver: [OpenLDAP Administrator's Guide: Replication with slurpd](#)

[Página www](#)[Página inicial](#)[Página 84 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Herramientas de administración de LDAP

Administrar LDAP desde línea de comandos resulta muy engoroso

- Existen numerosas herramientas visuales que facilitan la gestión de LDAP
- Algunas de ellas son:
  1. [phpldapadmin](#) - interfaz basada en web para administrar servidores LDAP
  2. [gosa](#) - herramienta de administración, basada en PHP, para gestión de cuentas y sistemas en LDAP
  3. [ldap-account-manager](#) - webfrontend para gestión de cuentas en un directorio LDAP
  4. [gq](#) - cliente LDAP basado en GTK+/GTK2 (bastante simple)
  5. [cpu](#) - herramientas de gestión para consola: proporciona comandos tipo `useradd/userdel` para usar con LDAP

[Página www](#)[Página inicial](#)[Página 85 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## 6. Compartición Linux-Windows: Samba

Samba permite a un sistema UNIX conversar con sistemas Windows a través de la red de forma nativa

- el sistema Unix aparece en el “Entorno de red” de Windows
- los clientes Windows pueden acceder a sus recursos de red e impresoras compartidas
- el sistema UNIX puede integrarse en un dominio Windows, bien como Controlador Primario del Dominio (PDC) o como miembro del dominio

[Página www](#)[Página inicial](#)[Página 86 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

Veremos una configuración “básica” de Samba; para saber más:

1. [The Samba Homepage](#)
2. [The Official Samba-3 HOWTO and Reference Guide](#)
3. [The Unofficial Samba HOWTO](#)
4. [The Linux Samba-OpenLDAP Howto](#)
5. [Using Samba, 2nd Edition, O'Reilly & Associates](#)
6. [Integración de redes con OpenLDAP, Samba, CUPS y PyKota](#)
7. [Curso de Integración de Sistemas Linux/Windows](#)

[Página www](#)[Página inicial](#)[Página 87 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Funcionamiento de Samba

Samba implementa los protocolos NetBIOS y SMB

- NetBIOS: protocolo de nivel de sesión que permite establecer sesiones entre dos ordenadores
- SMB (*Server Message Block*): permite a los sistemas Windows compartir ficheros e impresoras (llamado *Common Internet File System*, CIFS por Microsoft)

Samba ofrece los siguientes servicios

- Servicios de acceso remoto a ficheros e impresoras
- Autenticación y autorización
- Servicio de resolución de nombres

[Página www](#)[Página inicial](#)[Página 88 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Demonios Samba

Samba utiliza dos demonios: **smbd** y **nmbd**

- **smbd** permite la compartición de archivos e impresoras sobre una red SMB, y proporciona autenticación y autorización de acceso para clientes SMB; ofrece los dos modos de compartición de recursos existentes en Windows
  - ★ modo basado en usuarios o modo *user* (propio de los dominios Windows NT o 2000)
  - ★ modo basado en recursos o modo **share** (propio de Windows 3.11/95)
- **nmbd** permite que el sistema Unix participe en los mecanismos de resolución de nombres propios de Windows (WINS), lo que incluye
  - ★ anuncio en el grupo de trabajo
  - ★ gestión de la lista de ordenadores del grupo de trabajo
  - ★ contestación a peticiones de resolución de nombres
  - ★ anuncio de los recursos compartidos

[Página www](#)[Página inicial](#)[Página 89 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Otras utilidades Samba

Adicionalmente a los dos programas anteriores, Samba ofrece varias utilidades; algunas de las más relevantes son:

- **smbclient** interfaz que permite a un usuario de un sistema Unix conectarse a recursos SMB y listar, transferir y enviar ficheros
- **swat** (*Samba Web Administration Tool*) utilidad para configurar Samba de forma local o remota utilizando un navegador web
- **smbfs** sistema de ficheros SMB para Linux: permite montar un recurso SMB ofrecido por un servidor SMB (un sistema Windows o un servidor Samba) en un directorio local
- **winbind** permite al sistema UNIX resolver nombres y grupos desde un servidor Windows

[Página www](#)[Página inicial](#)[Página 90 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Instalación básica de Samba

Veremos una instalación básica de Samba en nuestro sistema Debian:

- permitirá desde un Windows acceder a los directorios de usuarios

## Instalación de los paquetes

El paquete básico a instalar es **samba** que incluye los demonios de Samba

- instala también el paquete **samba-common**, que incluye utilidades como **smbpasswd** y **testparm**

Otros paquetes de Samba son:

- **smbclient** herramientas para el cliente Samba
- **smbfs** comandos para montar y desmontar **smbfs**
- **swat** *Samba Web Administration Tool*
- **winbind**

[Página www](#)[Página inicial](#)[Página 91 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

Sólo instalaremos samba y samba-common

- la instalación nos pide un nombre de Grupo de Trabajo/Dominio
  - ★ indicar un nombre, que debemos usar en el sistema Windows

[Página www](#)[Página inicial](#)[Página 92 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Configuración de Samba

La configuración de Samba se realiza en el fichero `/etc/samba/smb.conf`

- establece las características del servidor Samba, así como los recursos que serán compartidos en la red

Ejemplo sencillo:

```
[global]
  workgroup = MIGRUPO
[homes]
  comment = Home Directories
[pub]
  path = /espacio/pub
```

[Página www](#)[Página inicial](#)[Página 93 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Estructura del archivo `smb.conf`

El fichero `/etc/samba/smb.conf` se encuentra dividido en secciones, encabezados por una palabra entre corchetes

- En cada sección figuran opciones de configuración, de la forma `etiqueta = valor`, que determinan las características del recurso exportado por la sección
- Existen tres secciones predefinidas: `global`, `homes` y `printers`
- Otras secciones (como `pub` en el ejemplo anterior) definen otros recursos para compartir

[Página www](#)[Página inicial](#)[Página 94 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Secciones predefinidas:

**[global]** define los parámetros de Samba a nivel global del servidor, por ejemplo, el programa utilizado para que un usuario pueda cambiar su clave (`passwd program`)

**[homes]** define automáticamente un recurso de red por cada usuario conocido por Samba; este recurso, por defecto, está asociado al directorio *home* del usuario en el ordenador en el que Samba está instalado

**[printers]** define un recurso compartido por cada nombre de impresora conocida por Samba

[Página www](#)[Página inicial](#)[Página 95 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

## Niveles de Seguridad

Samba ofrece dos modos de seguridad, correspondientes a los **dos modos de compartición de recursos** ya vistos

- Modo *share*: cada vez que un cliente quiere utilizar un recurso ofrecido por Samba, debe suministrar una contraseña de acceso asociada a dicho recurso
- Modo *user*: el cliente establece una sesión con el servidor Samba (mediante usuario y contraseña); una vez Samba valida al usuario, el cliente obtiene permiso para acceder a los recursos ofrecidos por Samba

El nivel de seguridad se especifica con la opción `security`, la cual pertenece a la sección `[global]`

```
security = share | user | server | domain | ADS
```

[Página www](#)[Página inicial](#)[Página 96 de 97](#)[Regresar](#)[Pantalla completa](#)[Cerrar](#)[Abandonar](#)

Los niveles **user**, **server**, **domain** y **ADS** corresponden todos ellos al modo de seguridad **user**

- Nivel **user**: el encargado de validar al usuario es el sistema Unix donde Samba se ejecuta; es necesario que existan los mismos usuarios y con idénticas contraseñas en los sistemas Windows y en el servidor Samba
- Nivel **server**: Samba delega la validación del usuario en otro ordenador, normalmente un sistema Windows 2000 (método no recomendado)
- Nivel **domain**: el ordenador en el que se delega la validación debe ser un Controlador de Dominio (DC), o una lista de DCs; el sistema Samba actúa como miembro de un dominio
- Nivel **ADS**: en Samba-3 permite unirse a un dominio basado en Active Directory como miembro nativo

El modo por defecto es **user**

## Otros comandos Samba

La suite Samba incluye otros comandos, como son:

- **testparm** permite chequear el fichero **smb.conf** para ver si es correcto
- **net** herramienta básica para administrar Samba y servidores SMB remotos; funciona de forma similar al comando **net** de DOS
- **smbpasswd** permite cambiar la contraseña usada en las sesiones SMB; si se ejecuta como root también permite añadir y borrar usuarios del fichero de contraseñas de Samba
- **smbstatus** muestra las conexiones Samba activas
- **smbclient** permite a un usuario de un sistema Unix conectarse a recursos SMB y listar, transferir y enviar ficheros