IURRETA INSTITUTUA
Goi-mailako Lanbide-Heziketako Berariazko Institutua

EUSKO JAURLARITZA
GOBIERNO VASCO
HEZKUNTZA UNIBERTSITATE ETA IKERKETA SAILA
DEPARTAMENTO DE EDUCACIÓN UNIVERSIDADES E INVESTIGACIÓN

Integración Linux - Windows

Parte2: Dominios Windows con Samba

Versión: 2.0.0

Alfredo Barrainkua Zallo

Julio del 2012

Índice

1. Introducción

Ya no queda empresa ni oficina que no dispongan de una red de área local. Ya sea por que se tiene gran cantidad de ordenadores, por que el almacenamiento de datos está centralizado, por que se desea compartir una impresora costosa, o simplemente la conexión a Internet, las redes de computadores se han hecho omnipresentes.

En el mundo empresarial, es típico encontrar redes de clientes Windows, con la autenticación centralizada en un controlador de dominio Windows, y carpetas compartidas en red. Esto hace que haya un coste económico importante asociado a las licencias de uso del software servidor.

Por otro lado, cada vez es más común encontrarse también en el mismo entorno, máquinas con Linux. Los entornos heterogéneos están haciéndose con el mundo de la informática.

Esto nos plantea varios problemas. Entre ellos, la compartición de información entre los distintos sistemas, y por el otro lado, el coste económico de las licencias.

La integración de máquinas Windows puede hacerse instalando software de cliente para NIS y NFS, por ejemplo. Esto nos trae el problema de tener que instalar y mantener software en un gran número de máquinas. Además este software también hay que pagarlo. Hay otra forma de hacer la integración. Hacer que una máquina Linux hable el protocolo nativo de los sistemas Windows: SMB (CIFS).

El software que tenemos para que máquinas Unix hablen SMB se llama **Samba**. Samba es un Samba es un servidor que implementa el protocolo SMB desarrollado inicialmente por IBM y Microsoft, para sus redes Lan Server y Lan Manager. Después, su desarrollo ha estado ligado a los productos Windows. Es un sistemas de red bastante “ruidoso” pero sencillo de configurar y con una funcionalidad bastante rica.

Vamos a instalar **Samba** como controlador de dominio windows sobre la distribución GNU/Linux **Ubuntu Server 12.04 LTS**. Como clientes de escritorio vamos a utilizar Windows XP Profesional. Se va a ver la configuración del controlador de dominio y de las carpetas compartidas a través de un ejemplo de una oficina. No se va a entrar en la funcionalidad de compartición de impresoras. En esta documentación, el dominio se denomina **NIREDOMAIN**, y el nombre del servidor es **NIRESERVER**. El dominio DNS será **nire-domain.net**. La IP del servidor será **192.168.33.44**.

No se va a utilizar servidor DNS, por lo que la resolución de nombres de host se realizará con wins.

2. Trabajos previos

Antes de instalar y configurar samba, realizaremos una serie de configuraciones en el servidor, que si bien no son específicas de samba, si son imprescindibles para su correcto funcionamiento. Son detalles que hay que cuidar en un servidor, para este servicio o para cualquier otro.

2.1 Nombre del servidor

Necesitamos poder resolver el nombre de nuestro servidor independientemente de que tengamos un servidor DNS. Al inicio del sistema, puede suceder que necesitemos resolver el nombre de nuestra máquina, y no tengamos aún cargada la red o el servicio de interrogación DNS, o simplemente puede suceder que no podamos acceder al servidor DNS. Para ello, vamos a verificar que el nombre del servidor esté correctamente establecido en los ficheros **/etc/hostname** y **/etc/hosts.** He aquí el contenido del fichero **/etc/hostname**:

```
txonpolo
```

El contenido del fichero **/etc/hosts**:

```
127.0.0.1    localhost
192.168.33.44txonpolo.nire-domain.net  txonpolo

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

2.2 Dirección IP

La IP deberá de ser fija. La habremos puesto durante la instalación y estará en el fichero **/etc/network/interfaces**. En este fichero aparecen todos los interfaces de red. Deberá contener algo similar a esto:

```
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
```

```
address 192.168.33.44
netmask 255.255.255.0
network 192.168.33.0
broadcast 192.168.33.255
gateway 192.168.33.1
# dns-* options are implemented by the resolvconf package, if installed
dns-nameservers 192.168.33.9 192.168.33.82
dns-search nire-institutua.net
```

2.3 Los imprescindibles?

Hay paquetes sin los cuales la vida se hace dura en un sistema en modo texto. Entre ellos se encuentran Midnight Commander y Lynx. Vamos a instalarlos.

```
aptitude install mc lynx
```

2.4 La hora, la hora, la hora

Debemos asegurarnos de que la hora esté correctamente configurada. Para ello, configuraremos el cliente **ntpdate** para que el reloj del sistema se sincronice con un servidor ntp. El paquete está instalado por defecto. En caso contrario instalarlo:

```
apt-get install ntpdate
```

Posteriormente, configuraremos el fichero **/etc/default/ntpdate** para que **ntpdate** sincronice la hora del sistema con el servidor de tiempo que deseemos. El fichero tendrá un contenido similar al siguiente:

```
# The settings in this file are used by the program ntpdate-debian, but not
# by the upstream program ntpdate.

# Set to "yes" to take the server list from /etc/ntp.conf, from package ntp,
# so you only have to keep it in one place.
NTPDATE_USE_NTP_CONF=no

# List of NTP servers to use  (Separate multiple servers with spaces.)
# Not used if NTPDATE_USE_NTP_CONF is yes.
NTPSERVERS="ntp.nire-domain.net ntp2.nire-domain.net"

# Additional options to pass to ntpdate
NTPOPTIONS=""
```

El comando ntpdate se ejecuta al inicio del sistema. Como un servidor está en marcha largos períodos de tiempo, conviene resincronizar el reloj del sistema una vez al día por si hubiera deslizamientos en el mismo. Para ello creamos un fichero ntpdate que sea ejecutado por el demonio cron una vez al día. Primero creamos el fichero:

```
touch /etc/cron.daily/ntpdate
```

Ponemos permisos de ejecución:

```
chmod +x /etc/cron.daily/ntpdate
```

Ponemos el siguiente contenido en su interior:

```
#!/bin/bash

ntpdate ntp.nire-domain.net
```

Lo probamos.

```
/etc/cron.daily/ntpdate
```

En Redes de Área Local, lo más probable es que haya un servidor ntp, que sea el que se sincroniza con un reloj que haya en Internet. En este caso, los demás equipos de la red (incluidos los equipos que den algún tipo de servicio) se sincronizarán con este servidor. Así conseguimos que sea un solo equipo el que tenga que comunicarse con un servidor de Internet, reduciendo el tráfico entre la LAN e Internet.

2.5 Deshabilitar el reinicio por teclado

La distribución Debian, por defecto tiene activado el reinicio del sistema cuando detecta la combinación de teclado **CTRL+ALT+DEL**. Debemos desactivarlo. De este modo, se deberá introducir el comando oportuno para reiniciar el sistema.

El el fichero **/etc/inittab**, debemos comentar (poner "**#**" delante) la siguiente línea, que quedará así:

```
#ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -a now
```

2.6 Acceso por la red : SSH

En un servidor es común acceder al mismo por la red. Esto nos permite acceder al mismo desde lugares remotos, o aunque estemos en la misma localización, no tener que instalar periféricos para el mismo. el acceso se puede realizar tanto en modo consola o gráfico.

Las comunicaciones conviene realizarlas cifradas. De este modo, no se transmitirán por la red nuestros datos (entre otros, contraseñas de acceso) en testo plano. Los datos se cifran con algún método criptográfico. No vamos a entrar aquí en teoría ni sistemas de certificados, ni cosas de esas. Vamos a ver simplemente un acceso de clave compartida.

El sistema de acceso que vamos a utilizar el **ssh**. Hay que mencionar que **ssh** también permite la copia de ficheros entre máquinas, de forma cifrada con **scp**. Vamos a instalarlo.

```
apt-get install ssh
```

Vamos a configurar el paquete. El fichero de configuración es **/etc/ssh/sshd_config**. Vamos a poner las siguientes líneas de esta forma:

```
Protocol 2
PermitRootLogin no
```

Si querremos securizar más el sistema, podemos evitar el acceso con password y que se usen certificados, limitarlo a unas máquinas, etc.

Para acceder al sistema como el usuario **elusuario**:

```
ssh elusuario@lamaquina.eldominio.net
```

Para copiar el fichero **mifile** de **/home/mihome** al directorio home del usuario **elusuario** en el sistema 10.22.1.59:

```
scp /home/mihome/mifile elusuario@10.22.1.59:/home/elusuario/
```

2.7 ACLs en las particiones de datos de Samba

Conviene proveer de la funcionalidad de Listas de Control de Acceso -**ACL**- POSIX a los directorios en las particiones donde se encuentren los datos que samba ha de servir. Para ello, a la hora de montar las particiones hay que especificar dicha opción. Vamos a poner lo siguiente en el fichero **/etc/fstab**:

```
/dev/lvolgroup1/lvol1     /datuak      reiserfs     acl    0      2
```

Después lo remontamos.

```
mount -o remount /datuak
```

Ya estamos preparados para comenzar la tarea.

3. Sobre SIDs y RIDs

Windows utiliza para identificar los objetos del dominio, identificadores de seguridad (Security Identifiers) o SIDs. No es el único sistema operativo que los utiliza, pero es el que conocemos.

A diferencia de Linux, que utiliza UIDs y GIDs, y que un objeto puede tener dos atributos con el mismo valor (root:0:0, por ejemplo), en Windows no puede haber dos SIDs iguales, por lo que dos objetos, aunque sean de diferente naturaleza (usuario, grupo), no pueden tener el mismo SID. Esto es MUY importante.

Un string SID viene a tener el siguiente aspecto:

`S-1-5-21-123456789-123456789-123456789-512`

El significado del formato SID viene dado en la siguiente tabla:

Significado de SIDs y RIDs en Windows	
S	Es un SID
1 (S-1)	Versión 1
5 (S-1-5)	Autoridad (5 -> NT)
21 (S-1-5-21)	SIDs no únicos. Utilizados para SIDs de dominios. Al SID S-1-5-21 le siguen 3 RIDs (96 biytes) que definen un dominio. E aspecto puede se algo así como S-1-5-21-123456789-123456789-123456789. Los tres números se crean de forma aleatoria al crearse el dominio. Al SID del dominio le sigue un RID identificando al objeto del dominio. Este RIS no es más que un número consecutivo que determina los onbjetos, pero hau una serie de RIDs bien conocidos.
512 (S-1-5-21-xxx-xxx-xxx-512)	RID correspondiente al grupo de administradores del dominio.
32 (S-1-5-32)	SIDs correspondientes a los grupos locales de un sistema Windows.
544 (S-1-5-32-544)	RID corrspondiente al grupo de administradores locales de un servidor Windows

Hay mucho más que decir de los SIDs y hay más SIDs en Windows pero no vamos a entrar en detalle. Además, tras la introducción de Vista, 7 y Server 2008, su número se ha multiplicado.

Los SID y RID más importantes / bien conocidos en Windows son los que vamos a ver a continuación.

SIDs Locales más Importantes en un Servidor Windows	
S-1-5-32-544	BUILTIN\Administrators
S-1-5-32-545	BUILTIN\Users
S-1-5-32-546	BUILTIN\Guests
S-1-5-32-547	BUILTIN\Power Users
S-1-5-32-548	BUILTIN\Account Operators
S-1-5-32-549	BUILTIN\Server Operators
S-1-5-32-550	BUILTIN\Print Operators
S-1-5-32-551	BUILTIN\Backup Operators
S-1-5-32-552	BUILTIN\Replicators

SIDs del Dominio más Importantes en una Red Windows	
S-1-5-21-xxx-xxx-xxx-512	Domain Admins
S-1-5-21-xxx-xxx-xxx-513	Domain Users
S-1-5-21-xxx-xxx-xxx-514	Domain Guests
S-1-5-21-xxx-xxx-xxx-515	Domain Computers
S-1-5-21-xxx-xxx-xxx-516	Domain Controllers
S-1-5-21-xxx-xxx-xxx-500	Administrator
S-1-5-21-xxx-xxx-xxx-501	Guest
S-1-5-21-xxx-xxx-xxx-502	KRBTGT

En las referencias podemos encontrar enlaces que nos condicen a más información sobre el tema.

4. Instalación y configuración de Samba

La instalación del software de Samba la haremos de la forma habitual. También necesitamos la herramienta **unix2dos**, que se encuentra en el paquete **dos2unix** y que Ubuntu 12.04 no instala por defecto:

```
apt-get install samba smbclient smbfs libpam-smbpass dos2unix acl
```

De cara a la configuración es importante conocer y / o recordar algunos de los ficheros y carpetas de los equipos Linux y de Samba:

/etc/passwd	Fichero con los usuarios locales
/etc/shadow	Fichero de contraseñas cifradas de los usuarios locales
/etc/group	Grupos locales y sus usuarios
/etc/nsswitch.conf	En la sección hosts de este fichero se indica el orden en el que se hará la resolución de nombres
/home	Directorios home de los usuarios
/etc/samba/smb.conf	Fichero de configuración de Samba
/var/lib/samba/passdb.tdb	Fichero con los usuarios de Samba. Puede ponerse en cualquier otro sitio
/etc/samba/smbusers	Fichero con los alias de los usuarios. Por ejemplo: **root = Administrator Administrador alfredo josean josu alfonso**
/var/lib/samba/netlogon	Aunque Ubuntu 12.04 sitúa este directorio en /home/samba/netlogon de manera predeterminada, antiguamente se situaba en /var/lib/samba/netlogon. A priori me parece un lugar más acertado.

Vamos a configurar primero un par de cosas para que el sistema funcione de forma más “ligera”. Vamos a indicar en el fichero **/etc/nsswitch.conf**, que la resolución de nombres la haga primero por los ficheros locales, luego con wins, y después a través del servicio DNS. La siguiente línea deberá estar así:

```
hosts: files wins dns
```

Ubuntu 12.04 no crea por defecto el fichero **/etc/samba/smbusers** para los alias de los usuarios. Vamos a crearlo, para acomodar un alias de root: Administrador. Este es el contenido:

```
root = Administrador
```

El fichero **/etc/samba/smb.conf** es el punto de referencia del servicio Samba. Todos los cambios que vayamos haciendo a nuestro sistema se verán reflejados en este fichero, y cada vez que hagamos un cambio habrá que reiniciar los demonios **smbd** y **nmbd** de samba o hacer que relean el fichero de configuración. Para ello utilizaremos los comandos **/etc/init.d/smbd restart** / reload o bien **/etc/init.d/nmbd reload / reload**.

Estos dos servicios están soportados por el sistema **Upstart** de Ubuntu, y por ello, también se pueden utilizar los comandos **restart** y **reload**. Como ejemplo: **restart nmbd** o **reload smbd**.

El fichero **/etc/samba/smb.conf** contiene entre otras, las tres secciones especiales: **global**, **homes** y **printers**. El resto de secciones lo conforman las carpetas compartidas o **shares**. Hay una carpeta compartida con un propósito especial. Es la carpeta **netlogon**. Este share se utiliza para compartir los scrips de inicio de sesión o **logon scripts**.

Ubuntu 12.04 tampoco crea el directorio destinado a los scripts de inicio. Lo crearemos en **/var/lib/samba/netlogon**.

```
mkdir -p /var/lib/samba/netlogon
```

He aquí una descripción somera de las secciones en el fichero **/etc/samba/smb.conf**:

global	En la sección global se definirá la configuración del servicio Samba.
homes	En la sección homes, se definirán las características que por defecto tendrán las carpetas personales de los usuarios.
printers	Equivalente a los homes de los usuarios, pero para las impresoras.
netlogon	En la sección netlogon, se definirá el directorio en el que estarán los scripts de inicio de sesión y algunas de las características de estos.
shares genéricos	Un share es una carpeta compartida en red a través del servidor Samba. Cada share tiene su propia sección. En dicha sección se definen sus condiciones de compartición.

Veamos los parámetros más habituales en la sección **global**:

workgroup	Define el nombre del dominio de la máquina, o el grupo de trabajo en ausencia de dominio.
netbios name	Nombre NETBIOS del host
server string	Cadena que proporciona cuando se conecta a él. Meramente descriptiva.
os level	Nivel del sistema operativo. El las redes Lan Manager, la publicación de los recursos disponibles en los grupos de trabajo la realiza el Loca Master Browser. Para determinar quién va a hacerse cargo de dicho papel se realizan elecciones. Cada máquina se vota a sí misma, y cada sistema operativo

	tiene un número de votos. Un Win 2000 Server tiene más votos (32) que un Win 2000 PRO. Los Win 9x tienen menos. En caso de empate, se hace cargo del papel la máquina que más tiempo lleve encendida. Este parámetro indica el número de votos de nuestra máquina.
time server	Indica si hace la labor de servidor de tiempo para las máquinas del dominio SMB.
local master	Hace que se participe en las elecciones para ser el LMB.
preferred master / prefered master	Es el Local Master Browser preferido de nuestra subred. Al inicio va a intentar forzar elecciones para ser el LMB.
domain master	Hace que sea el controlador de dominio. Es decir es el LMB de toda la subnet, y el resto de LMB le ceden las listas de recursos. Se activa por defecto cuando se define como el servidor de logons de dominio “domain logons”. En caso de soportar “domain logons” y definir este parámetro como “no”, se estaría trabajando como un BDC.
domain logons	Provee el servicio netlogon a los clientes Win9x. Se supone que también hace de PDC.
enhanced browsing	Propagación de la anunciación de servicios entre subredes. No es estándar en los servicios de Microsoft.
unix extension	Extensiones para soportar enlaces simbólicos y otras características. No soportado por Microsoft.
security	Determina el modo en que se establece la seguridad de los recursos compartidos. user ads share server domain
encrypt passwords	Uso de contraseñas cifradas. Desde NT 4.0 SP3 y Win98, es la opción preferida.
passdb backend	Tipo y ubicación de la base de datos de los usuarios de Samba y de sus contraseñas.
password level	Numero de caracteres en mayúscula con los que se intenta contrastar el password. Viene derivado por un problema de Windows para grupos, con el protocolo LANMAN1. Aunque se especifique el nivel de protocolo COREPLUS!, intenta enviar las contraseñas en mayúsculas.
username map	Fichero que contiene mapeos entre usuarios del sistema y de Samba. root = Administrador

logon script	Script que se ejecutará al autentificarse el usuario. %U se sustituye por el nombre de usuario.
logon drive	Unidad a la que se mapea la carpeta de cada usuario.
map to guest	Indica lo que hay que hacer con un intento de login inválido. Típicamente por que no se ha suministrado el nombre de usuario Linux correctamente, o el password, o no existe... Never Bad user Bad password Bad uid
guest account	El usuario de sistema del cual tomarán los privilegios los invitados. nobody
log level	El nivel de registro. Cuanto más alto, más detallados serán los mensajes registrados y más crecerán los ficheros de registro. Un nivel aceptable es 2.
max log size	Tamaño máximo de los ficheros de registro. Se rotarán cuando se alcance dicho tamaño.
add machine script	Script que se ejecuta al añadir una máquina al dominio. Se crea la cuenta de sistema de dicha máquina. Si no funciona la habremos de crear a mano.
add user script	Script a ejecutar al añadir un usuario a Samba. Solamente es útil si se utiliza en entornos con un PDC WinNT.
delete user script	Script a ejecutar al borrar un usuario de Samba.
add group script	Script a ejecutar al añadir un grupo a Samba. Solamente es útil en entornos que utilicen las herramientas administrativas de dominio de NT.
delete group script	Script a ejecutar al borrar un grupo de Samba.
add user to group script	Script a ejecutar al añadir un usuario a un grupo de Samba. Solamente es útil en entornos que utilicen las herramientas administrativas de dominio de NT.
logon home	Lugar donde se sitúa el perfil móvil de los usuarios de máquinas Win95 / Win98.
logon path	Lugar donde se sitúa el perfil móvil de los usuarios de máquinas WinNT / Win2000 / WinXP. Si lo dejamos vacío, se desactivan los perfiles móviles.
hosts allow	Hosts o redes a las que se les permite acceder aun recurso. Si se indica en la sección **global** es para todos los recursos del sistema.
hosts deny	Lo contrario a la opción anterior.

spnego	Hace que se use el mecanismo Simple and Protected NEGOtiation.

La mayoría de éstos parámetros tienen unos valores por defecto. Si no indicamos dicho parámetro, toma el valor por defecto. En la mayoría de casos, estos valores nos son útiles, y no necesitamos definirlos. En este manual vamos a poner todos los relevantes, en aras de conocer su existencia.

En diversos lugares se hace uso de variables. Esto permite insertar en el lugar correspondiente, el valor de la variable en tiempo de ejecución. Las variables más importantes son:

%U	Se sustituye por el nombre del usuario va a iniciar la sesión.
%G	Nombre del grupo primario del usuario %U.
%L	Nombre NETBIOS del servidor donde se ha realizado el inicio de sesión.
%m	Nombre NETBIOS de la máquina cliente donde se ha realizado el inicio de sesión.
%h	Nombre DNS del servidor donde corre Samba.
%D	Nombre del dominio Windows.
%u	Nombre del usuario del servicio. Se sustituye una vez realizado el logon.

Aquí vemos el aspecto que va a ofrecer nuestra sección **global**:

```
[global]
       workgroup = MIDOMINIO
       netbios name = MISERVIDOR
       server string = Servidor Samba para MIDOMINIO
       os level = 33

       time server = yes
       local master = yes
       preferred master = yes
       domain master = yes
       domain logons = yes

       enhanced browsing = no
       unix extensions = yes
       security = user

       encrypt passwords = yes
       passdb backend = tdbsam:/var/lib/samba/passdb.tdb
       password level = 1
       username map = /etc/samba/smbusers

       logon script = %U.bat
       logon drive = U:
       map to guest = Bad user
       guest account = nobody

       log level = 2
```

```
	max log size = 7000

	add machine script = /usr/sbin/useradd -g domcmputers -c 'Domeinuko Ordenagailu
Kontua' -s /bin/false -d /var/lib/samba %u
	add user script = /usr/sbin/useradd -m %u
	delete user script = /usr/sbin/userdel -r %u
	add group script = /usr/sbin/groupadd %g
	delete group script = /usr/sbin/groupdel %g
	add user to group script = /usr/sbin/usermod -g %g %u

	logon path =
# \\%L\%U\profile
```

Parámetros de la sección **homes**:

comment	Meramente descriptivo
valid users	Usuarios que tienen acceso al sistema de carpetas home
browsable / browseable	Si el directorio puede ser visto en la red
writable / writeable	Si se puede escribir en él
create mask	Los permisos que tendrán por defecto los ficheros creados en los directorios home
directory mask	Los permisos que tendrán por defecto los directorios creados en los directorios home
guest ok	Indica si el directorio debe ser visible a los usuarios de la red sin proveer contraseña.

Este es el aspecto de nuestra sección **homes**:

```
[homes]
	comment = Carpetas personales de los usuarios
	valid users = %S
	browseable = no
	writeable = yes
	create mask = 0640
	directory mask = 0750
	guest ok = no
```

Parámetros de la sección **netlogon**:

comment	Meramente descriptivo
path	Lugar donde se sitúan los scripts de inicio de sesión
browsable / browseable	Si el directorio puede ser visto en la red
writable / writeable	Si se puede escribir en él

Este es el aspecto de nuestra sección **netlogon**:

```
[netlogon]
```

```
        comment = Servicio logon de la red
        path = /var/lib/samba/netlogon
        browseable = no
        writeable = no
```

Vamos a ver ahora los parámetros más usuales de una carpeta compartida en red:

comment	Meramente descriptivo
path	Lugar donde se sitúan los scripts de inicio de sesión
valid users	Usuarios que tienen acceso a la carpeta compartida
admin users	Lista de usuarios con privilegios administrativos. Todas las operaciones las realizan como root.
write list	Lista de usuarios y grupos que tienen derecho de escritura en la carpeta. Separar con comas, y a los grupos añadir el carácter “@” delante.
force user	Nombre del usuario Linux que se le va a asignar al usuario conectado a esta carpeta.
force group	Nombre del grupo Linux primario que se le va a aplicar al usuario conectado a esta carpeta. Ello significa que los ficheros escritos en ella son propiedad de ese grupo.
browsable / browseable	Si el directorio puede ser visto en la red
writable / writeable	Si se puede escribir en él
create mask	Los permisos que tendrán por defecto los ficheros creados en dicho directorio
directory mask	Los permisos que tendrán por defecto los directorios creados en dicho directorio
guest ok	Indica si el directorio debe ser visible a los usuarios de la red sin proveer contraseña.

Como ejemplo, vamos a poner una carpeta compartida llamada **documentos**. Esta carpeta está situada en **/home/carpetas/documentos**. El grupo de **managers** puede escribir en ella, y también el usuario **jefe**. Todos pueden leer su contenido. Hay un usuario **boss** que es el administrador de la carpeta. La carpeta no ha de ser vista en la red.

Este es el aspecto que tendría la configuración de esta carpeta compartida:

```
[documentos]
        comment = Documentos para todos
        path = /home/carpetas/documentos
        admin users = boss
        write list = @managers, boss
        browseable = no
        guest ok = no
```

Una de las tareas que se han de realizar al instalar un servidor es mapear los grupos habituales en Windows con los grupos del sistema Linux. Esto solamente se realiza una vez. Después, cuando creamos grupos de usuarios, también tendremos que hacerlo con los nuevos grupos. Hay que tener en cuenta que no todos los sistemas Linux tienen los mismos grupos por defecto, por lo que algunos los tendremos que crear primero. Suponemos un sistema Ubuntu 12.04 LTS. Vamos a ver los comandos a ejecutar para ello:

Grupos nuevos del sistema:

```
groupadd -g 512 domadmins
groupadd -g 513 domusers
groupadd -g 514 domguests
groupadd -g 515 domcomputers
groupadd -g 516 domcontrollers
groupadd powerusers
```

Grupos del dominio:

```
net groupmap add ntgroup="Domain Admins" unixgroup=domadmins rid=512 type=d
net groupmap add ntgroup="Domain Users" unixgroup=domusers rid=513 type=d
net groupmap add ntgroup="Domain Guests" unixgroup=domguests rid=514 type=d
net groupmap add ntgroup="Domain Computers" unixgroup=domcomputers rid=515 type=d
net groupmap add ntgroup="Domain Controllers" unixgroup=domcontrollers rid=516 type=d
```

Grupos locales de NT (BUILTIN):

```
net groupmap add ntgroup="Administrators" unixgroup=sys sid=S-1-5-32-544 type=l
net groupmap add ntgroup="Users" unixgroup=users sid=S-1-5-32-545 type=l
net groupmap add ntgroup="Guests" unixgroup=nogroup sid=S-1-5-32-546 type=l
net groupmap add ntgroup="Power Users" unixgroup=powerusers sid=S-1-5-32-547 type=l
net groupmap add ntgroup="Account Operators" unixgroup=staff sid=S-1-5-32-548 type=l
net groupmap add ntgroup="Server Operators" unixgroup=daemon sid=S-1-5-32-549 type=l
net groupmap add ntgroup="Print Operators" unixgroup=lp sid=S-1-5-32-550 type=l
net groupmap add ntgroup="Backup Operators" unixgroup=bin sid=S-1-5-32-551 type=l
net groupmap add ntgroup="Replicators" unixgroup=kmem sid=S-1-5-32-552 type=l
```

Ahora creamos unos usuarios del dominio que aunque no son estrictamente necesarios, es posible que algunos paquetes de software los busquen, con lo que aumentemos la compatibilidad con los controladores de dominio Windows.

```
useradd -c 'Domeinuko Administratzailea' -g domusers -u 500 -m -s /bin/false
administrator
useradd -c 'Domeinuko Gonbidatua' -g domguests -u 501 -m -s /bin/false guest
useradd -c 'Kerberos Tiket Banatzailea' -g domusers -u 502 -m -s /bin/false krbtgt

pdbedit -a -U 500 -f "Administrator" -c "[UX ]" -u administrator
pdbedit -r -c "[X ]" -u administrator
pdbedit -a -U 501 -f "Guest" -u guest
pdbedit -r -c "[D ]" -u guest
pdbedit -a -U 502 -f "KRBTGT" -u krbtgt
pdbedit -r -c "[D ]" -u krbtgt
```

Ahora solo nos queda asignar permisos especiales al grupo de administradores del dominio y a los grupos locales de control de impresoras y de copias de seguridad. Esto último tampoco es estrictamente necesario, pero es posible que algún software de copias

de seguridad se siente más cómodo si estos usuarios del servidor tienen los susodichos privilegios.

```
      net rpc rights grant -U root "IURRETA3\Domain Admins" SeMachineAccountPrivilege
SePrintOperatorPrivilege SeAddUsersPrivilege SeDiskOperatorPrivilege
SeRemoteShutdownPrivilege

      net rpc rights grant -U root "BUILTIN\Print Operators" SePrintOperatorPrivilege
      net rpc rights grant -U root "BUILTIN\Backup Operators" SeBackupPrivilege
SeRestorePrivilege
```

Vamos a probar el sistema. Primero verificamos la correctitud del fichero de configuración:

```
      testparm
```

Vamos a reiniciar el servicio:

```
      restart smbd
      restart nmbd
```

Ahora debemos asegurarnos de que responde tanto en el interface loopback como en el interface re red.

```
      smbclient -L localhost -U%
      smbclient -L NOMBRESERVIDOR -U%
```

Con estos pasos ya tenemos un sistema básico operativo para hacer las labores de controlador de un dominio de clientes windows. Vamos ahora a realizar un pequeño toque en el fichero **/etc/sudoers** para que el grupo de administradores del dominio pueda realizar cualquier acción en el sistema, introduciendo su contraseña, al igual que los miembros del grupo sudo. Añadimos la siguiente línea al final del fichero:

```
%domadmins   ALL=(ALL:ALL) ALL
```

Ahora tenemos que realizar la aplicación requerida. Eso será en el próximo capítulo.

5. Un ejemplo

Como ejemplo puede valer la red de una pequeña empresa, en la cual somos los administradores. Vamos a crear un controlador de dominio con Samba. instalación, configuración, servicios, usuarios, grupos,....

Esta empresa llamada **SUPERSOFT** tendrá dos tipos de trabajadores: supervisores y programadores. Los programadores tendrán distintas funciones, como será programar en c, html, python y java. Los supervisores supervisarán los trabajos de algunos de los programadores, no de todos.

De esta manera, cada grupo de la siguiente tabla, tendrá un directorio en el servidor, al que podrán acceder tanto los programadores de ese lenguaje como sus supervisores. Además de estos directorios habrá uno personal por cada usuario, donde tan solo los usuarios propietarios tienen acceso. También habrá un directorio general para la oficina, donde los supervisores tendrán permisos de lectura y escritura y los programadores tan sólo de lectura. Los supervisores tendrán un directorio de grupo al que sólo ellos tienen acceso.

Tendremos los siguientes usuarios:

NOMBRE	GRUPO	USUARIO	CONTRASEÑA
Mikel Aretxabaleta Maiz	supervisor	mikelare	1234mail
Josune Goitia Lopez	supervisor	josunego	9493zept
Karlos Brown Artea	supervisor	karlosbr	7321rera
Ander Hirigoien Gartzia	cprog	anderhir	8828idoa
Miren Uria Mendiola	cprog	mirenuri	9019olaa
Miren Garcia Sertutxa	javaprog	mirengar	7836ylkk
Janire Urkizu Agirre	htmlprog	janireur	4321dkll
Lorea Sanchez Agirre	htmlprog	loreasan	1236vasd
Jon Garcia Garcia	pythonprog	jongarci	4473mdfg

Tenemos, por tanto, un grupo de supervisores y dentro de los programadores, distintas funciones. Además, tendremos en cuenta que los supervisores supervisan más de un lenguaje de programación, por lo que serán miembros de esos grupos. Estos son los supervisores y los grupos que supervisan:

Mikel Aretxabaleta	cprog, htmlprog, javaprog
Josune Goitia	cprog, javaprog, pythonprog
Karlos Brown	htmlprog, javaprog, pythonprog

Cuando un programador entra en el dominio, se le montarán automáticamente distintas carpetas: carpeta personal, carpeta de grupo y carpeta de la empresa. Cuando un supervisor se “loguea”, se le montarán la carpeta personal, las de los grupos que supervisa, la de los supervisores y la de la empresa (**empresa**).

Los supervisores tendrán permisos de lectura y escritura en las carpetas de los grupos donde supervisan. En la carpeta de supervisores, los supervisores podrán leer y escribir. En la carpeta de la empresa, los supervisores podrán leer y escribir y los programadores sólo leer.

Las unidades de red a asignar serán las siguientes:

U	Carpeta personal de cada usuario
T	Carpeta del grupo primario
X	Carpeta general de toda la empresa
P,Q,R	Carpetas de los grupos a supervisar

Vamos a ponernos manos a la obra. Nuestro jefe nos ha comprado una máquina nueva. Su nombre será **SUPERSERVER**. Nuestro dominio será **SUPERSOFT**.

El software ya lo hemos instalado, por lo que nos queda configurarlo y crear los usuarios, grupos y carpetas...

5.1 Configuración global

Vamos a realizar la configuración básica del fichero de configuración de Samba **/etc/samba/smb.conf**. Quedará de esta manera:

```
[global]
       workgroup = SUPERSOFT
       netbios name = SUPERSERVER
       server string = Servidor Samba para SUPERSOFT
       os level = 33

       time server = yes
       local master = yes
       preferred master = yes
       domain master = yes
       domain logons = yes
       enhanced browsing = no
```

```
        unix extensions = yes
        security = user

        encrypt passwords = yes
        passdb backend = tdbsam:/var/lib/samba/passdb.tdb
        password level = 1
        username map = /etc/samba/smbusers

        logon script = %U.bat
        logon drive = U:
        map to guest = Bad user
        guest account = nobody
        log level = 2
        max log size = 7000

        add machine script = /usr/sbin/useradd -g maquinas -c 'Máquina del dominio' -s
/bin/false -d /var/lib/nobody '%u'

        logon path = \\%N\%U\profile

[homes]
        comment = Carpetas personales de los usuarios
        valid users = %S The current service name is sustituted for %S
        browseable = no
        writeable = yes
        create mask = 0640
        directory mask = 0750
        guest ok = no

[netlogon]
        comment = Servicio logon de la red
        path = /var/lib/samba/netlogon
        browseable = no
        writeable = no
```

Vamos a crear el alias de root:

```
        echo “root = Administrador” > /etc/samba/smbusers
```

Los mapeos de grupos habituales Windows a grupos Linux están realizados ya, por lo que no es necesario volver a realizarlo.

5.2 Creación de grupos

Vamos a crear los grupos. Para crear los grupos utilizaremos el comando **groupadd**. Después los mapearemos a grupos Windows del mismo nombre:

```
        groupadd supervisor
        groupadd cprog
        groupadd htmlprog
        groupadd javaprog
        groupadd pythonprog

        net groupmap add ntgroup=”supervisor” unixgroup=supervisor
        net groupmap add ntgroup=”cprog” unixgroup=cprog
        net groupmap add ntgroup=”htmlprog” unixgroup=htmlprog
        net groupmap add ntgroup=”javaprog” unixgroup=javaprog
```

```
net groupmap add ntgroup="pythonprog" unixgroup=pythonprog
```

Podemos ver lo que tenemos mapeado con el siguiente comando:

```
net groupmap list
```

5.3 Creación de usuarios y scrips de inicio de sesión

Para que un usuario pueda utilizar una máquina de la red, debe de tener una cuenta de usuario. Con este nombre de usuario y su correspondiente contraseña, se autentificará, pudiendo acceder al dominio y a sus recursos. Vamos a crear los usuarios. Utilizaremos el comando **useradd**:

```
useradd -g supervisor -c 'Mikel Aretxabaleta Maiz' -s /bin/bash -m mikelare
useradd -g supervisor -c 'Josune Goitia Lopez' -s /bin/bash -m josunego
useradd -g supervisor -c 'Karlos Brown Artea' -s /bin/bash -m karlosbr
useradd -g cprog -c 'Ander Hirigoien Gartzia' -s /bin/bash -m anderhir
useradd -g cprog -c 'Miren Uria Mendiola' -s /bin/bash -m mirenuri
useradd -g javaprog -c 'Miren Garcia Sertutxa' -s /bin/bash -m mirengar
useradd -g htmlprog -c 'Janire Urkizu Agirre' -s /bin/bash -m janireur
useradd -g htmlprog -c 'Lorea Sanchez Agirre' -s /bin/bash -m loreasan
useradd -g pythonprog -c 'Jon Garcia Garcia' -s /bin/bash -m jongarci
```

Ahora vamos a asignarles las contraseñas:

```
passwd mikelare
passwd josunego
passwd karlosbr
passwd anderhir
passwd mirenuri
passwd mirengar
passwd janireur
passwd loreasan
passwd jongarci
```

Los usuarios creados están en el fichero **/etc/passwd**, y los grupos en **/etc/group**. Hemos dicho en el enunciado de la práctica, que algunos usuarios van a ser a supervisar el trabajo de otros, y para ello necesitan permisos en las carpetas de esos otros grupos. Para ello podemos editar el fichero **/etc/group** y poner los usuarios en lops grupos correspondientes. También se puede utilizar el comando **usermod**. Vamos a añadir a los supervisores a los grupos a los que supervisan.

```
usermod -a -G cprog,htmlprog,javaprog mikelare
usermod -a -G cprog,javaprog,pythonpro josunego
usermod -a -G htmlprog,javaprog,pythonprog karlosbr
```

Veamos a verificar cómo han quedado los grupos y los usuarios del sistema. Para ello vamos a utilizar los siguientes comandos:

```
getent group
getent passwd
```

El siguiente paso es hacer que los usuarios del sistema lo sean también de samba. Para

ello, tenemos que crear los usuarios en el fichero que hayamos indicado en el parámetro passwd backend del fichero de configuración de Samba. En este caso será **/var/lib/samba/passdb.tdb**. El comando a utilizar será **pdbedit**:

```
pdbedit -a -u mikelare
pdbedit -a -u josunego
pdbedit -a -u karlosbr
pdbedit -a -u anderhir
pdbedit -a -u mirenuri
pdbedit -a -u mirengar
pdbedit -a -u janireur
pdbedit -a -u loreasan
pdbedit -a -u jongarci
```

Vamos a ver si los usuarios han sido creados correctamente:

```
pdbedit -L
```

Ahora ya solamente nos queda crear los scripts de inicio de sesión. Cuando un usuario se autentifica, el controlador de dominio, entre otra información le indica a la máquina cliente cuál es el nombre del fichero de scrips de inicio de sesión. La máquina cliente se conecta a la carpeta compartida con nombre **netlogon**, en el servidor de inicio de sesión y busca dicho fichero. Lo lee y lo ejecuta en la máquina local en nombre del usuario recién conectado. Nosotros vamos a utilizar este mecanismo, para montar automáticamente las carpetas que van a usar nuestros usuarios.

Los scripts de inicio de sesión (logon script) son ficheros que se guardan normalmente en **/var/lib/samba/netlogon**. En el caso de Debian, esta carpeta no existe, por lo que habrá que crearla. También se puede elegir otra ubicación. Vamos a crearla:

```
mkdir /var/lib/samba/netlogon
```

Hemos dicho que la finalidad de nuestros scripts de inicio de sesión va a ser montar carpetas compartidas de red. Las carpetas que se han de montar son la personal, la del grupo primario, la de la empresa y las de los otros grupos de los cuales también somos miembros. Este es el contenido de cada línea del fichero:

net use V: \\nombredelservidor\recurso

El nombre del servidor indicara donde se encuentran los recursos a montar, y el recurso será lo que habrá que montar. Dependiendo del usuario los recursos a montar serán distintos. Las letras indican la unidad, y en un mismo script no se pueden repetir. Así mismo, no se pueden utilizar letras que sean utilizadas por el sistema para referenciar discos duros, unidades de CD o de otro tipo.

Es importante que al acabar la línea introduzcamos un salto de línea entendible por Windows, sino esa línea no se tendrá en cuenta. Cada sistema operativo utiliza un carácter distinto para indicar el final de línea. El estilo DOS de salto de línea se corresponde con los caracteres <CR><LF>, mientras que en el estilo Unix es <LF>. CR es Carriage Return y LF Line Feed. En el mundo Macintosh el salto de línea es <CR>.

Nosotros crearemos los ficheros y los editaremos en un entorno Linux, pero la ejecución se dará en un equipo NT. Si el fin de línea no es entendible por Windows, todo lo que haya dentro del fichero se considerará como una sola línea, por lo que los comandos serán erróneos. Para convertir el salto de línea estilo Unix al estilo DOS utilizaremos el comando **unix2dos**. Los logon script o ficheros de inicio de sesión se guardan como **nombre_usuario.bat.**

Veamos el aspecto del fichero **mikelare.bat** correspondiente al usuario **mikelare**:

```
net use U: \\SUPERSERVER\mikelare
net use T: \\SUPERSERVER\supervisor
net use X: \\SUPERSERVER\empresa
net use P: \\SUPERSERVER\cprog
net use Q: \\SUPERSERVER\htmlprog
net use R: \\SUPERSERVER\javaprog
```

Para cambiar los finales de línea al estilo del DOS:

```
unix2dos /var/lib/samba/netlogon/mirentxu.bat
```

Con el resto de usuarios deberemos de hacer lo mismo.

5.4 Creación de carpetas compartidas

Vamos a crear un directorio que contenga las carpetas de grupos, y a cada grupo le creamos una carpeta de grupo:

```
mkdir /grupos
mkdir /grupos/empresa
mkdir /grupos/supervisor
mkdir /grupos/cprog
mkdir /grupos/htmlprog
mkdir /grupos/javaprog
mkdir /grupos/pythonprog
```

Tendremos que crear un grupo de nombre **maquinas** y hacer que las máquinas que metamos al dominio sean miembros de ese grupo. Esto último se realizará a través del script **add machine script** del fichero **/etc/samba/smb.conf**.

```
groupadd maquinas
net groupmap add ntgroup=”maquinas” unixgroup=maquinas
```

Ahora deberemos cambiar el grupo propietario de las carpetas y adecuar los permisos. Lo primero se debe a que las carpetas han sido creadas por root, y su grupo primario es root. Vamos con el cambio de grupo:

```
chgrp supervisor /grupos/empresa
chgrp supervisor /grupos/supervisor
chgrp cprog /grupos/cprog
chgrp htmlprog /grupos/htmlprog
chgrp javaprog /grupos/javaprog
```

```
chgrp pythonprog /grupos/pythonprog
```

También vamos a cambiar los permisos de la carpeta de los grupos, para que solamente sus miembros tengan acceso a ellas. La de la empresa vamos a dejarla con acceso de lectura para todos.

```
chmod 775 /grupos/empresa
chmod 770 /grupos/supervisor
chmod 770 /grupos/cprog
chmod 770 /grupos/htmlprog
chmod 770 /grupos/pythonprog
```

Vamos a modificar el fichero **/etc/smb.conf** para poner las carpetas compartidas en red. Lo siguiente se lo habremos de añadir al fichero.

```
[empresa]
       comment = Carpeta general de la empresa
       path = /grupos/empresa
       browsable = yes
       create mask = 0664
       directory mask = 0775
       write list = @supervisor
       force group = supervisor

[supervisor]
       comment = Carpeta de los supervisores
       path = /grupos/supervisor
       browsable = no
       create mask = 0660
       directory mask = 0770
       valid users = @supervisor
       write list = @supervisor
       force group = supervisor

[cprog]
       comment = Carpeta de cprog
       path = /grupos/cprog
       browsable = no
       create mask = 0660
       directory mask = 0770
       valid users = @cprog
       write list = @cprog
       force group = cprog

[htmlprog]
       comment = Carpeta de htmlprog
       path = /grupos/htmlprog
       browseable = no
       create mask = 0660
       directory mask = 0770
       valid users = @htmlprog
       write list = @htmlprog
       force group = htmlprog

[javaprog]
       comment = Carpeta de javaprog
       path = /grupos/javaprog
       browsable = no
       create mask = 0660
```

```
	directory mask = 0770
	valid users = @javaprog
	write list = @javaprog
	force group = javaprog

[pythonprog]
	comment = Carpeta de pythonprog
	path = /grupos/pythonprog
	browsable = no
	create mask = 0660
	directory mask = 0770
	valid users = @pythonprog
	write list = @pythonprog
	force group = pythonprog
```

Vamos a verificar que el fichero de configuración es corecto:

```
	testparm
```

Una vez hechos los cambios en smb.conf, y verificada su correctitud, indicamos a samba que relea el fichero de configuración:

```
	/etc/init.d/samba reload
```

IMPORTANTE: Hay una cierta inconsistencia en este ejercicio. Hemos supuesto que en la carpeta general de la empresa solamente escribe el grupo supervisor. Normalmente, la realidad será más compleja. Para tener una granularidad mayor, deberemos de configurar la carpeta compartida de la siguiente forma:

```
[empresa]
	comment = Carpeta general de la empresa
	path = /grupos/empresa
	browsable = yes
	create mask = 0664
	directory mask = 0775
	admin users = @supervisor
	valid users = @domusers
	write list = @domusers
	force group = +supervisor
```

Después, afinaremos los permisos con los acl-s. Usando el comando **`setfacl`**.

5.5 Otro ejercicio

Suponiendo que eres en administrador de sistemas de una empresa. Implanta un controlador de dominio realizado con Samba y Linux.

Esta empresa va a tener tres grupos de usuarios. Los grupos son **compras**, **ventas** y **gerencia**. Como ejemplo, he aquí unos empleados de la empresa.

Nombre	Grupo	Usuario	Contraseña
Juan Zurreta Cid	compras	juanzu	cid44732
Pepe Garai Yurrebeso	compras	pepega	yurre724

Maria Lopez Zaragoza	compras	marialo	zara4224
Josefa Hernández Zarrabeitia	ventas	josefahe	zarra332
José Auroberri Ansolegi	ventas	josean	anso1272
Gorka Fernández Areitio	gerencia	gorkafe	arei4212
Arantza Ortuzar Ortiz	gerencia	arantzaor	orti7762

Grupos:

. compras
. ventas
. gerencia

Cuando un usuario se autentifica en el dominio, se le montarán automáticamente las carpetas compartidas personal, la de su grupo, la de la empresa y las de los grupos en las que está autorizado. En la carpeta personal, solamente puede leer y escribir el usuario correspondiente. En las carpetas **gerencia**, **compras** y **ventas** solamente los miembros de esos grupos pueden leer y escribir. Además los miembros del grupo de **gerencia** pueden leer en las mismas. En la carpeta empresa todos tienen el derecho de leer, pero solamente pueden escribir los miembros de la **gerencia**.

Letras para el mapeo de las carpetas de red:

U	La carpeta personal de cada usuario
T	Carpeta del grupo
X	Carpeta general de la empresa
R	Carpeta compras para el grupo de gerencia
S	Carpeta ventas para el grupo de gerencia

Este ejercicio lo vamos a realizar con la ayuda de los scripts del siguiente apartado.

6. Scrips que nos ahorrarán trabajo

Hemos visto que muchas de las tareas realizadas son repetitivas. Para hacernos la vida un poco más fácil podemos realizar unos scripts. Estos scripts los guardaremos en la carpeta del administrador (**root**). Vamos a crear dos scripts para automatizar la creación de grupos y la de usuarios. El primero se llama **creargrupo.sh** y como se ve a continuación pide el nombre de grupo, lo crea, crea la carpeta y le da a la carpeta como grupo propietario el nombre del grupo. Este es su contenido:

```
#!/bin/bash
#
# Alfredo Barrainkua, Junio 2007
#
echo -n "Nombre del grupo: "
read GRUPO

groupadd $GRUPO
net groupmap add ntgroup="$GRUPO" unixgroup=$GRUPO

mkdir -p /grupos/$GRUPO
chgrp $GRUPO /grupos/$GRUPO
chmod 770 /grupos/$GRUPO

echo "
[$GRUPO]
       comment = Carpeta del grupo $GRUPO
       path = /grupos/$GRUPO
       browsable = no
       create mask = 0660
       vreate directory = 0770
       valid users = @$GRUPO
       write list = "$GRUPO
       force group = $GRUPO
" >> /etc/samba/smb.conf
```

Tendremos que hacer el script ejecutable:

```
chmod +x creargrupo.sh
```

La forma de utilizarlo (como root):

```
       /root/creargrupo.sh
```

El segundo script, de nombre **crearusuario.sh**, crea usuarios para el sistema y para samba. Primero pide el nombre completo del usuario, el nombre de usuario y el grupo al que pertenece (por eso es conveniente crear antes los grupos que los usuarios). A

continuación crea el usuario y le asigna una contraseña que será solicitada por pantalla. Después se creará el usuario para samba, por lo que volveremos a introducir otras dos veces la contraseña. Finalmente crea el scripts de inicio de sesión básico para el usuario.

```
#!/bin/bash
#
# Alfredo Barrainkua, Junio 2007
#
# Directorio donde se ubican los logon-script
LOGONDIR=/var/lib/samba/netlogon/

# Suponemos quue el nombre DNS y NETBIOS del servidor es el mismo
SRVNAME=`hostname`

echo -n "Nombre completo del usuario: "
read NOMBRE

echo -n "Nombre de usuario: "
read USUARIO

echo -n "Password: "
read PASSWORD

echo -n "Grupo al que pertenece: "
read GRUPO

echo "Grupos en los que también trabaja (separados por comas y sin espacios): "
read GRUPOS

echo "$PASSWORD" > passfile.tmp
PASSMD5=`openssl passwd -1 -stdin -noverify < passfile.tmp`

useradd -g $GRUPO -c "$NOMBRE" -s /bin/bash -m -p $PASSMD5 $USUARIO
usermod -a -G $GRUPOS $USUARIO


echo "$PASSWORD" >> passfile.tmp
pdbedit -a -t -u $USUARIO < passfile.tmp
rm passfile.tmp

if [ ! -d $LOGONDIR ]; then
       mkdir -p $LOGONDIR
fi
echo "net use t: \\\\$SRVNAME\\$GRUPO" >> $LOGONDIR$USUARIO.bat
echo "net use x: \\\\$SRVNAME\\empresa" >> $LOGONDIR$USUARIO.bat
unix2dos $LOGONDIR$USUARIO.bat

echo "
Recuerde añadir los grupos adicionales en el logon-script."
```

Como se ve al final del fichero, al usuario se le montarán además de su carpeta personal, la de su grupo primario y una genérica perteneciente a su empresa, en al inicio de sesión. El fichero ha de ejecutarse en un sistema con fines de línea de ficheros de texto, al estilo DOS, por lo que hay que convertirlo.

Por defecto hemos configurado que a los usuarios se les mapean las carpetas de usuario y la de grupo, pero muchas veces, y en el ejemplo ocurre así, no todos los usuarios tendrán

mapeadas solo esas carpetas. En este caso los supervisores tendrán mapeadas las carpetas que supervisan, y tanto supervisores como programadores la carpeta de la empresa. Estas personalizaciones deberán ser hechas a mano. O en caso de un uso intensivo tendremos que crear un script más complejo.

Al igual que con el script de creación de grupos, habremos de hacerlo ejecutable. La forma de utilizarlo como **root**, sería:

```
/root/crearusuario.sh
```

Si no hemos hecho el script ejecutable, tendremos que pedir a bash que lo ejecute:

```
bash ./crearusuario.sh
```

7. Metiendo máquinas al dominio

Si todo ha ido bien, este paso no debería de dar ningún problema. Meter los equipos NT al dominio Samba es igual que meterlos en un dominio Windows. Botón derecho en **Mi PC**, Propiedades, Nombre del equipo, Identificador de Red y aquí indicamos cual es el dominio. Aceptamos y nos pedirá el nombre y la contraseña de un administrador con derechos de meter máquinas al dominio.

NOTA: En caso de tener problemas para la búsqueda del controlador de dominio, ponemos su dirección IP en la solapa **WINS** de la configuración del interface de red. Debemos tener en cuenta que en este supuesto no estamos utilizando un servidor DNS para la resolución de nombres.

7.1 Máquinas Windows 7

En el caso de hosts Windows 7, debemos de añadir un par de entradas al registro de Windows para que sean capaces de unirse a un dominio Linux/Samba con Samba 3.6.3.

Deberemos de ejecutar el siguiente script en el cliente Windows 7:

```
Windows Registry Editor Version 5.00

; Windows 7 ostalariak Samba domeinuan sartzeko
[HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\LanManWorkstation\Parameters]
"DNSNameResolutionRequired"=dword:00000000
"DomainCompatibilityMode"=dword:00000001
```

También podemos modificar las siguientes entradas del registro de windows para mejorar la velocidad del cliente windows

```
; Abiadura ezarpenak
[HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\System]
"SlowLinkDetectEnabled"=dword:00000000
"DeleteRoamingCache"=dword:00000001
"WaitForNetwork"=dword:00000000
"CompatibleRUPSecurity"=dword:00000001
```

Deshabilitar avisos idiotas para el usuario.

```
; Kendu abisu tontoak
       [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System]
       "EnableLUA"=dword:00000000'
```

Listo! Lo podemos ejecutar.

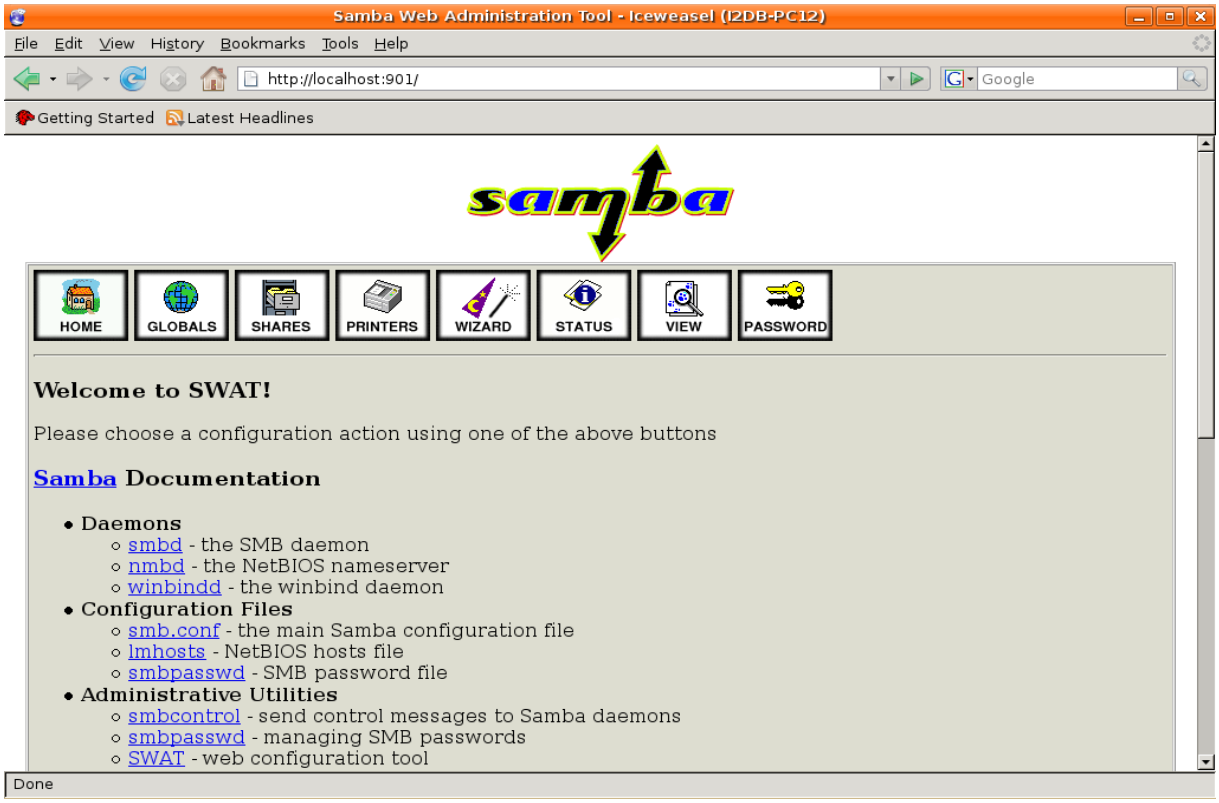
8. Trabajando con herramientas gráficas

swat es una herramienta para configurar samba desde un entrono gráfico vía web. Lo primero es instalarlo:

```
apt-get install swat
```

Para utilizarlo nos tenemos que conectar (como **root**) al puerto 901 de nuestro equipo:

http://localhost:901

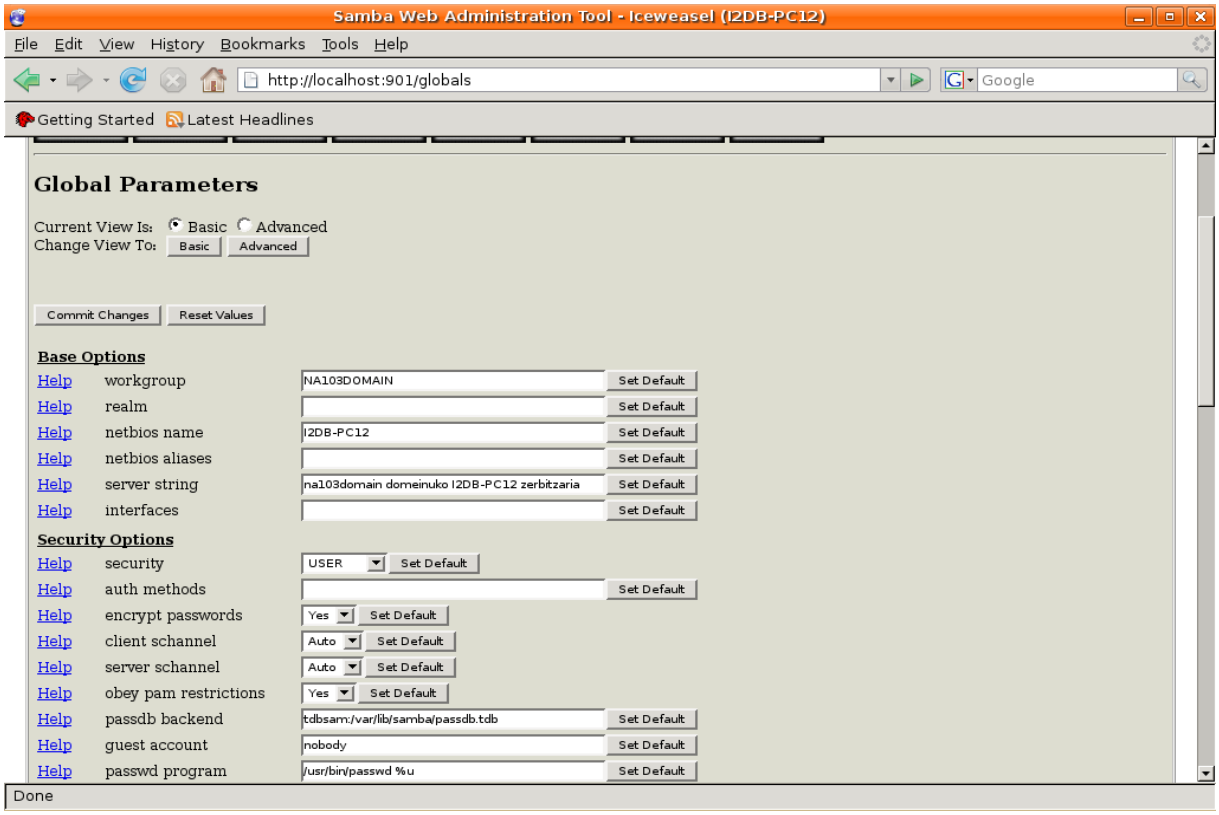


Para acceder al entorno gráfico, aparecerá una ventana con nombre de usuario y contraseña. Evidentemente es el nombre y contraseña de quien tiene privilegios para configurar samba: **root**.

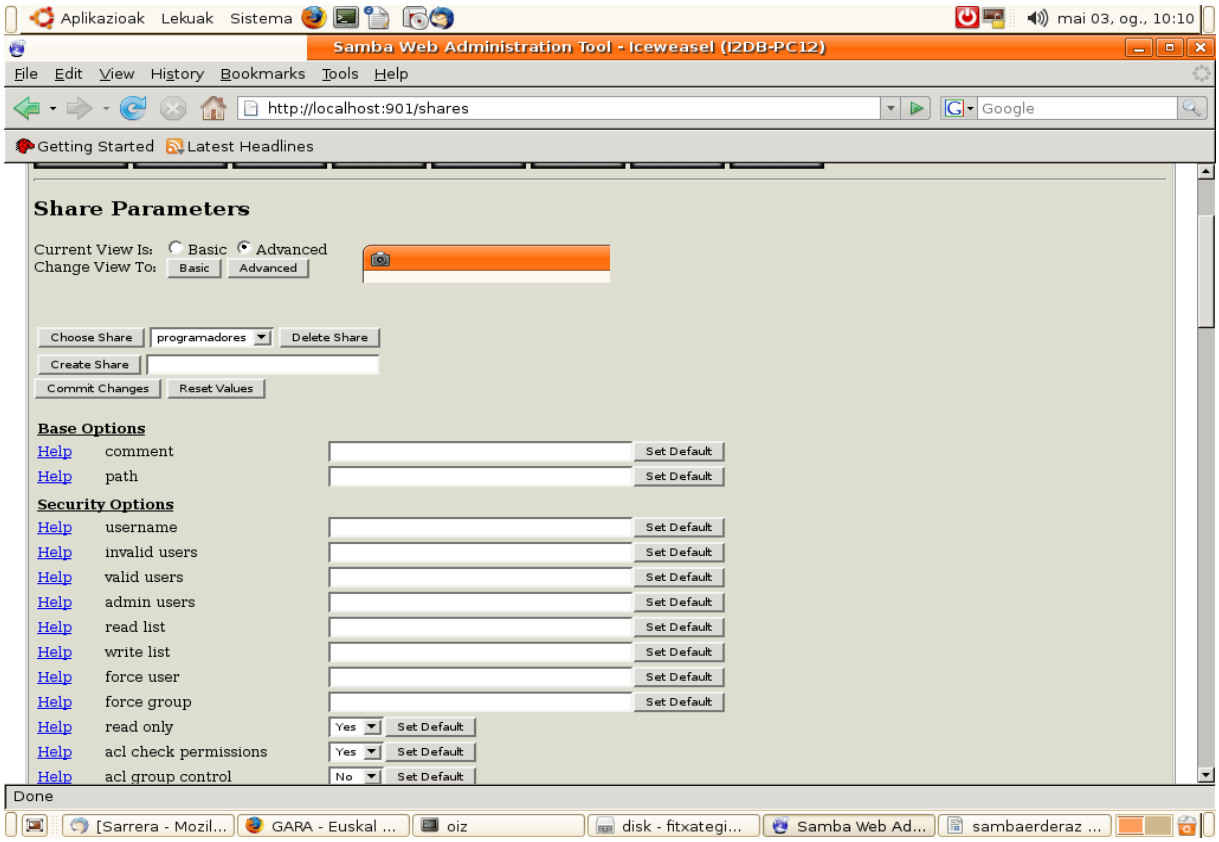
Una vez que sabemos como instalar y acceder a swat y los fundamentos de samba, la utilización de swat no tiene ningún misterio.

En la sección home, tenemos documentación sobre diferentes ficheros, configuraciones, utilidades,... de samba. No es más que una suma de páginas man.

En la siguiente imagen vemos el contenido de la sección global, que se corresponde con la sección global del fichero de configuración de samba smb.conf:

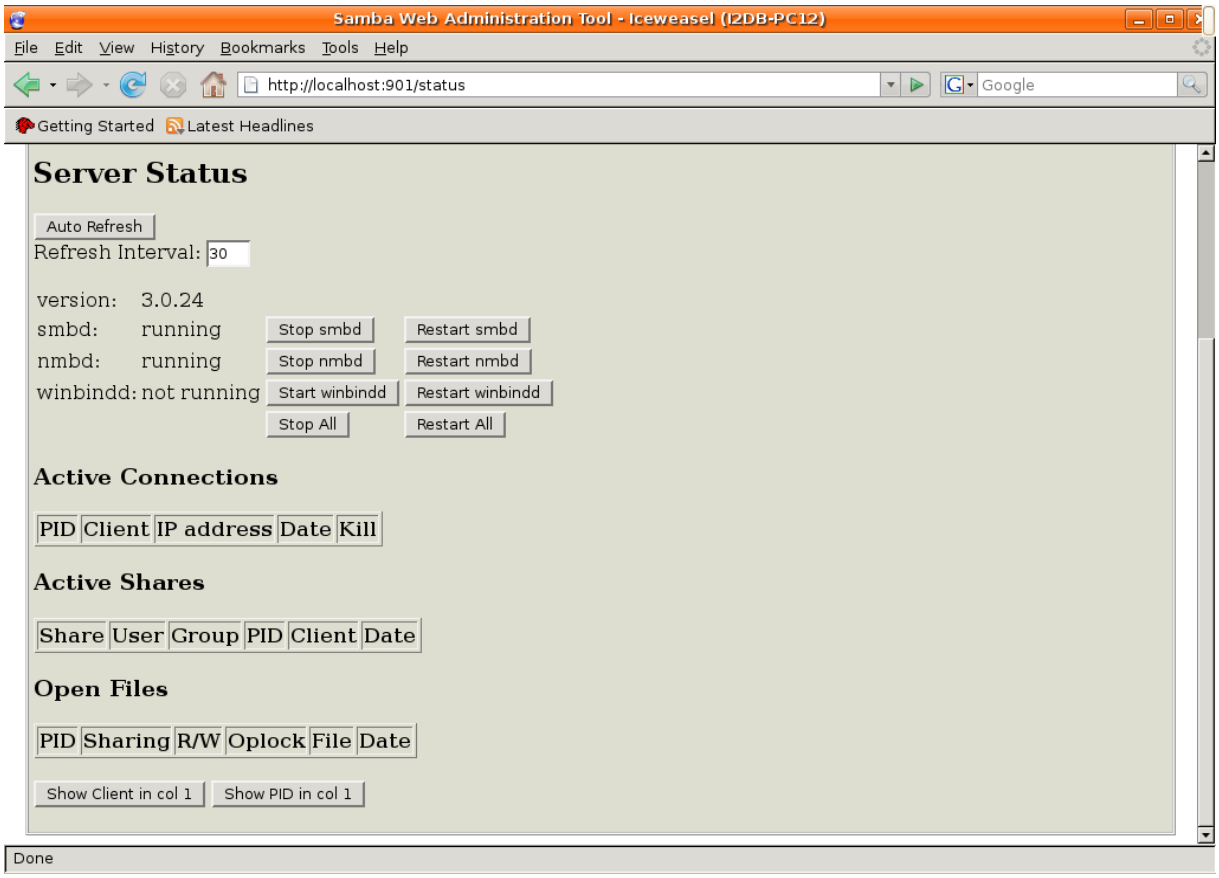


La siguiente sección es shares, donde se definirán las carpetas compartidas y sus características:



La sección printers, evidentemente, nos ayuda a configurar las impresoras y la sección wizard nos ayuda a configurar samba de una manera supuestamente más gráfica y fácil.

Con status, podemos visualizar cual es la situación de los demonios smbd y nmbd, también se pueden parar y reiniciar. Podemos ver cuales son las conexiones activas, los shares que están siendo utilizados y quién los está utilizando.



swat es una herramienta utilizable a gusto del usuario, pero es imprescindible “saber” samba. Se podría decir que no aporta especialmente nada más que un entorno vía web para trabajar con samba.

9. Referencias

Un libro práctico y ... divertido?!:

Samba-3 By Example 2. Edition. John H. Terpstra

Lo que nunca hay que olvidar:

man smb.conf

En la web:

http://us1.samba.org/samba/docs/using_samba/toc.html
http://us1.samba.org/samba/docs/man/Samba-HOWTO-Collection/
http://samba.org/samba/docs/man/Samba-HOWTO-Collection/AccessControls.html#id397568
https://help.ubuntu.com/12.04/serverguide/samba-fileserver.html
https://help.ubuntu.com/12.04/serverguide/samba-printserver.html
https://help.ubuntu.com/12.04/serverguide/samba-fileprint-security.html
https://help.ubuntu.com/12.04/serverguide/samba-dc.html
https://help.ubuntu.com/community/Samba
http://support.microsoft.com/en-us/library/windows/desktop/aa379649(v=vs.85).aspx
http://support.microsoft.com/default.aspx?scid=kb;en-us;243330

10. Autor

Alfredo Barrainkua Zallo

Iurreta Institutuko IKT Arduraduna

alfredobz@iurreta-institutua.net