Instalación y Configuración de Samba en el Laboratorio F1

Oscar de la Torre Robles
Universidad de León, España

oscarlane@eresmas.com

Este documento ilustra la manera en la que debe configurarse correctamente un servidor Samba, dando servicio a la compartición de archivos e impresoras entre equipos con distintos sistemas operativos (Windows 2000,95,98,XP,Me,GNU/Linux,Mac...). Tambien explica cómo se encuentran montados y configurados en la actualidad los servicios Samba. Ademas de Servidor, Samba tambien es un servicio cliente, es decir, contiene un programa para poder acceder a recursos remotos de distintas plataformas. En resumen, este documento describe cómo usar el protocolo Session Message Block (SMB), también llamado protocolo NetBIOS o LanManager, con GNU/Linux.

Introducción

El protocolo SMB es usado por Microsoft Windows 3.11, NT y 95 para compartir discos e impresoras. Usando el paquete de herramientas Samba creado por Andrew Tridgell, las máquinas UNIX (incluyendo Linux) pueden compartir discos e impresoras con servidores Windows.

Hay cuatro cosas que uno puede hacer con Samba:

Compartir una unidad de Linux con máquinas Windows.

Compartir una unidad de Windows con máquinas Linux.

Compartir una impresora de Linux con máquinas Windows.

Compartir una impresora de Windows con máquinas Linux.

Todas estas posibilidades han sido probadas en el laboratorio F1. Al no disponder de impresoras físicas se han utilizado impresoras virtuales. Todos estos puntos están cubiertos en este manual.

Instalación

Instalación de Samba

La última versión del código de Samba está disponible en:

ftp://nimbus.anu.edu.au/pub/tridge/samba/

De todas maneras, si has instalado la distribución RedHat de Linux, tienes la opción de instalarlo como paquete. Algunas otras distribuciones también incluyen los ejecutables de Samba.

Se requieren los dos demonios siguientes para el paquete Samba. Se suelen instalar en /usr/sbin y se pueden ejecutar tanto desde los scripts de arranque del sistema como desde inetd. Algunos scripts de ejemplo los puedes ver en Ejecutando los demonios.

smbd: El demonio de SMB.

nmbd: Provee un nameserver de NetBIOS para soporte de clientes.

Habitualmente, se instalan en /usr/bin los siguientes ejecutables de Samba, aunque la localización (como de costumbre) es opcional.

smbclient: Un cliente SMB para maquinas UNIX.

smbprint: Un script para imprimir a una impresora en un servidor SMB.

smbprint.sysv: Como el de encima, pero para máquinas UNIX SVR4.

smbstatus: Lista de las conexiones SMB en marcha en el servidor local.

smbrun: Un script 'cola' para facilitar la ejecución de aplicaciones en servidores.

Adicionalmente, se incluye en este documento un script llamado 'print', que sirve como un útil front end para el script smbprint.

El paquete Samba es sencillo de instalar. Simplemente consigue el código fuente del servidor que nombramos antes, y lee el fichero README de la distribución. Hay también un fichero llamado docs/INSTALL.txt en la distribución que te da un sencillo conjunto de instrucciones paso a paso.

Siguiendo con la instalación, pon los demonios en /usr/sbin y los ejecutables en /usr/bin. Instala las páginas del manual en /usr/local/man.

Cuando compiles el paquete Samba, deberías especificar en el Makefile la localización del fichero de configuración, smb.conf. Generalmente debería estar en /etc, pero puedes ponerlo donde quieras. A estas alturas, presumimos que especificaste la localización del fichero de configuración como /etc/smb.conf, el fichero de registro como /var/log/samba-log.%m y el directorio de bloqueo como /var/lock/samba.

Instala el fichero de configuración, smb.conf. Ve al directorio donde se compiló el Samba. Mira en el directorio examples/simple y lee el fichero README. En ese directorio encontrarás el fichero smb.conf. Cópialo en /etc. Hemos de tener una distribución de Linux que tiene el Samba instalado ya, puede que ya tengas un fichero de configuración en /etc. Probablemente deberías usar el antiguo.

Si no quieres que tu configuración esté en /etc, ponla donde quieras y luego pon un enlace simbólico en /etc:

```
ln -s /path/to/smb.conf /etc/smb.conf
```

Demonios

Arrancando los demonios

Los dos demonios de SMB son /usr/bin/smbd y /usr/sbin/nmbd.

Puedes ejecutar los demonios de Samba desde inetd o como procesos independientes . Si estás configurando un servidor de ficheros permanente, deberían ejecutarse desde inetd para que sean reejecutados si 'mueren'. Si solo quieres usar los servicios SMB de vez en cuando o como ayuda a la administración del sistema, puedes ejecutarlos con un script en /etc/rc.d/init.d o incluso a mano cuando los necesites.

Para ejecutar los demonios desde inetd, pon las siguientes líneas en el fichero de configuración de inetd, /etc/inetd.conf:

```
# Servicios SAMBA NetBIOS (para compartición de ficheros e impresoras en PC)
netbios-ssn stream tcp nowait root /usr/sbin/smbd smbd
netbios-ns dgram udp wait root /usr/sbin/nmbd nmbd
```

Entonces reejecuta inetd con el siguiente comando:

```
kill -HUP 1
```

Para ejecutarlos desde los scripts de inicio del sistema, pon las siguientes líneas en /etc/rc.d/init.d/smb y hazle un enlace simbólico con los ficheros indicados en los comentarios:

```
#!/bin/sh
#
# /etc/rc.d/init.d/smb - comienza y termina los servicios SMB.
#
# Se deben crear los siguientes ficheros como enlaces simbolicos a este fichero:
# symlinks: /etc/rc.d/rc1.d/K35smb  (Termina los servicios SMB al cerrar el sistema)
#           /etc/rc.d/rc3.d/S91smb  (Comienza los servicios SMB en modo multiusuario)
#           /etc/rc.d/rc6.d/K35smb  (Termina los servicios SMB al hacer un reboot)
#

# Libreria de funciones
. /etc/rc.d/init.d/functions

# Configuracion de red
. /etc/sysconfig/network

# Asegurarse que la red esta a punto
[ ${NETWORKING} = "no" ] and exit 0

# Comprobar como fuimos llamados
case "$1" in
```

```
  empezar)
    echo -n "Poniendo en marcha los servicios SMB: "
    daemon smbd -D
    daemon nmbd -D
    echo
    touch /var/lock/subsys/smb
    ;;
  parar)
    echo -n "Terminando los servicios SMB: "
    killproc smbd
    killproc nmbd
    rm -f /var/lock/subsys/smb
    echo ""
    ;;
  *)
    echo "Modo de uso: smb {empezar|parar}"
    exit 1
esac
```

Configuración

Configuración General (/etc/smb.conf)

La configuración de Samba en un Linux (u otra máquina UNIX) es controlada por un solo fichero, /etc/smb.conf. Este fichero determina qué recursos del sistema quieres compartir con el mundo exterior y que restricciones deseas poner en ellos.

Como las siguientes secciones 'direccionarán' la compartición de unidades e impresoras de Linux con máquinas Windows, el fichero smb.conf mostrado en esta sección es lo más simple posible, solo para propósitos introductorios.

No te preocupes por los detalles, aún. Otras secciones más adelante introducirán los conceptos más importantes.

Cada sección del fichero empieza con una cabecera como [global], [impresoras], etc.

La sección [global] define unas pocas variables que Samba usará para definir la compartición de todos los recursos.

La sección [homes] permite a los usuarios remotos acceder a sus respectivos directorios principales en la máquina Linux local (cada uno al suyo nada más). Esto es, si un usuario de Windows intenta conectar a este recurso desde su máquina Windows, será conectado a su directorio personal. A tener en cuenta que para hacer esto, tiene que tener una cuenta en la máquina Linux. ;-)

El fichero smb.conf que viene debajo es el que está funcionando en el laboratorio F1. Permite a los usuarios remotos acceder a su directorio principal en la máquina local y escribir en un directorio temporal. Para que un usuario de Windows vea estos recursos, la máquina Linux debe estar en la red local. Entonces el usuario simplemente conecta una unidad de red desde el Explorador de Windows o el Windows File Manager.

Fíjate que en las siguientes secciones, se darán entradas adicionales a este fichero para permitir la compartición de más recursos.

```
;
; /etc/samba/smb.conf
;
; Sample configuration file for the Samba suite for Debian GNU/Linux
;
; Please see the manual page for smb.conf for detailed description of
; every parameter.
;

[global]
;   printing = bsd
;   printcap name = /etc/printcap
;   load printers = yes
;   guest account = nobody
   invalid users = root

; Sólo permite conexiones de la red 192.168.2.x
; ¡¡ OJO !!  Si se pone, no se puede acceder. Hay que revisar por qué.

;  bind interfaces only = yes
;  interfaces = 192.168.2.2/24
;  hosts allow = 192.168.2.0/24

; "security = user" is always a good idea. This will require a Unix account
; in this server for every user accessing the server.
   security = user

; Change this for the workgroup your Samba server will part of
workgroup = SISTEMASOP

   server string = %h Servidor linux Litio

; If you want Samba to log though syslog only then set the following
; parameter to 'yes'. Please note that logging through syslog in
; Samba is still experimental.
   syslog only = no

; We want Samba to log a minimum amount of information to syslog. Everything
; should go to /var/log/{smb,nmb} instead. If you want to log through
; syslog you should set the following parameter to something higher.
   syslog = 0;

; This socket options really speed up Samba under Linux, according to my
; own tests.
   socket options = IPTOS_LOWDELAY TCP_NODELAY SO_SNDBUF=4096 SO_RCVBUF=4096

; Passwords are encrypted by default. This way the latest Windows 95 and NT
; clients can connect to the Samba server with no problems.
   encrypt passwords = yes
```

```
; It's always a good idea to use a WINS server. If you want this server
; to be the WINS server for your network change the following parameter
; to "yes". Otherwise leave it as "no" and specify your WINS server
;  below (note: only one Samba server can be the WINS server).
; Read BROWSING.txt for more details.
   wins support = no

; If this server is not the WINS server then specify who is it and uncomment
; next line.
;   wins server = 172.16.0.10

; Please read BROWSING.txt and set the next four parameters according
; to your network setup. There is no valid default so they are commented
; out.
;   os level = 0
;   domain master = no
;   local master = no
;   preferred master = no

; What naming service and in what order should we use to resolve host names
; to IP addresses
   name resolve order = lmhosts host wins bcast

; This will prevent nmbd to search for NetBIOS names through DNS.
   dns proxy = no

; Name mangling options

   preserve case = yes
   short preserve case = yes

; This boolean parameter controlls whether Samba attempts to sync. the Unix
; password with the SMB password when the encrypted SMB password in the
; /etc/samba/smbpasswd file is changed.
   unix password sync = false

; the passwd program in Debian Potato).
   passwd program = /usr/bin/passwd %u
   passwd chat = *Enter\snew\sUNIX\spassword:* %n\n *Retype\snew\sUNIX\spassword:* %n\n .

; The following parameter is useful only if you have the linpopup package
; installed. The samba maintainer and the linpopup maintainer are
; working to ease installation and configuration of linpopup and samba.
   message command = /bin/sh -c '/usr/bin/linpopup "%f" "%m" %s; rm %s'

; The default maximum log file size is 5 MBytes. That's too big so this
; next parameter sets it to 1 MByte. Currently, Samba rotates log
; files (/var/log/{smb,nmb} in Debian) when these files reach 1000 KBytes.
; A better solution would be to have Samba rotate the log file upon
; reception of a signal, but for now on, we have to live with this.
   max log size = 1000
```

```
[homes]
   comment = Home Directories
   browseable = no

; By default, the home directories are exported read only. Change next
; parameter to "no" if you want to be able to write to them.
   read only = yes

; File creation mask is set to 0700 for security reasons. If you want to
; create files with group=rw permissions, set next parameter to 0775.
   create mask = 0700

; Directory creation mask is set to 0700 for security reasons. If you want to
; create dirs. with group=rw permissions, set next parameter to 0775.
   directory mask = 0700

;[printers]
;   comment = All Printers
;   browseable = no
;   path = /tmp
;   printable = yes
;   public = no
;   writable = no
;   create mode = 0700

; A sample share for sharing your CD-ROM with others.
;[cdrom]
;   comment = Samba server's CD-ROM
;   writable = no
;   locking = no
;   path = /cdrom
;   public = yes
;
; The next two parameters show how to auto-mount a CD-ROM when the
; cdrom share is accesed. For this to work /etc/fstab must contain
; an entry like this:
;
;       /dev/scd0   /cdrom  iso9660 defaults,noauto,ro,user   0 0
;
; The CD-ROM gets unmounted automatically after the connection to the
;
; If you don't want to use auto-mounting/unmounting make sure the CD
; is mounted on /cdrom
;
;   preexec = /bin/mount /cdrom
;   postexec = /bin/umount /cdrom


[archivos_en_Litio]
comment = Directorio para almacenar archivos desde otros ordenadores
writable = yes
path = /var/ftp-samba/samba/subir_archivos_litio/
```

```
browseable = yes
valid users = sergio citizenx oscar joseluis luis esther jon
create mask = 0664
directory mask = 2775


[archivos_personales]
comment = Archivos personales
writable = yes
path = /var/ftp-samba/samba/archivos_personales/
browseable = yes
valid users = sergio jon oscar esther citizenx joseluis luis

[copias_seguridad]
comment = Copias de seguridad
writable = yes
patch = /var/ftp-samba/samba/copias_seguridad/
browseable = yes
valid user = jon oscar sergio citizenx joseluis luis

[pdf]
comment = Impresora virtual para imprimir en PDF
path = /tmp
printable = yes
guest ok = yes
print command = /usr/local/bin/printpdf.sh %s
lpq command =
lprm command =

[almacen-pdf]
comment = Ficheros impresos por la impresora virtual
path = /var/ftp-samba/samba/servicio-pdf
browseable = yes
writeable = yes
guest ok = yes
```

Compartir Archivos

Si todo ha ido bien, tenemos ya instalado el software servidor y las herramientas necesarias para compartir archivos. En nuestro supuesto práctico, hablamos de usar un disco duros como soporte físico para almacenar la información.

Compartir archivos en GNU/Linux con Máquinas Windows

Como se muestra en el fichero smb.conf anterior, compartir una unidad Linux con usuarios Windows es fácil. De todas maneras, como todo lo demás con Samba, puedes tener las cosas MUY controladas. Aquí tienes unos pocos ejemplos:

Para compartir un directorio con todo el mundo, crea una copia de la sección [tmp] añadiendo algo como esto al smb.conf:

```
[public]
   comment = Cosas publicas
   path = /home/public
   public = yes
   writable = yes
   printable = yes
```

Para que este directorio lo pueda leer todo el mundo, pero que sólo lo puedan cambiar gente del grupo 'laborales', modifica la entrada de esta manera:

```
[public]
   comment = Cosas publicas
   path = /home/public
   public = yes
   writable = yes
   printable = no
   write list = @laborales
```

Para aprender otros truquillos con que jugar con las unidades compartidas, mira la documentación de Samba o las páginas del man.

Compartir Una Unidad Windows Con Máquinas Linux

Se incluye un programa cliente de SMB para máquinas UNIX con la distribución de Samba. Provee un interfaz estilo ftp para la línea de comandos. Puedes usar esta utilidad para transferir ficheros entre un 'servidor' Windows y un cliente unix.

Para ver qué recursos están disponibles en un host dado, ejecuta:

```
/usr/sbin/smbclient -L host
```

donde 'host' es el nombre de la máquina que quieres 'ver'. Esto devolverá un lista de nombres de 'servicios' --esto es, nombres de unidades o impresoras que puede compartir contigo--. A menos que el servidor SMB no tenga la seguridad configurada, te preguntará por una clave. Dale la clave de la cuenta de 'invitados' o de tu cuenta personal en esa máquina.

Por ejemplo:

```
smbclient -L zimmerman
```

La salida de este comando debería ser algo parecido a esto:

```
Server time is Sat Aug 10 15:58:27 1996
Timezone is UTC+10.0
Password:
Domain=[WORKGROUP] OS=[Windows NT 3.51] Server=[NT LAN Manager 3.51]

Server=[ZIMMERMAN] User=[] Workgroup=[WORKGROUP] Domain=[]

        Sharename      Type      Comment
        ---------      ----      -------
        ADMIN$         Disk      Remote Admin
        public         Disk      Public
        C$             Disk      Default share
        IPC$           IPC       Remote IPC
        OReilly        Printer   OReilly
        print$         Disk      Printer Drivers


This machine has a browse list:

        Server               Comment
        ---------            -------
        HOPPER               Samba 1.9.15p8
        KERNIGAN             Samba 1.9.15p8
        LOVELACE             Samba 1.9.15p8
        RITCHIE              Samba 1.9.15p8
        ZIMMERMAN
```

La lista muestra otros servidores SMB con recursos para compartir con la red.

Para usar el cliente, ejecuta:

```
/usr/sbin/smbclient servicio password
```

donde 'servicio' es una máquina y un servicio. Por ejemplo, si estás intentando entrar en un directorio que ha sido compartido como 'public' en una máquina llamada zimmerman, el servicio debería llamarse \\zimmerman\public. De todas maneras, debido a restricciones del shell, necesitarás poner las barras invertidas con secuencias de escape, por lo que al final saldrá algo parecido a esto:

```
/usr/sbin/smbclient \\\\zimmerman\\public miclave
```

donde 'miclave' es una cadena literal con tu password.

Entonces te aparecerá el 'prompt' del smbclient:

```
Server time is Sat Aug 10 15:58:44 1996
Timezone is UTC+10.0
Domain=[WORKGROUP] OS=[Windows NT 3.51] Server=[NT LAN Manager 3.51]
smb: \>
```

Escribe 'h' para obtener una ayuda de como usar el cliente:

```
smb: \> h
ls              dir             lcd             cd              pwd
get             mget            put             mput            rename
more            mask            del             rm              mkdir
md              rmdir           rd              prompt          recurse
translate       lowercase       print           printmode       queue
cancel          stat            quit            q               exit
newer           archive         tar             blocksize       tarmode
setmode         help            ?               !
smb: \>
```

Si sabes usar el ftp, no deberías necesitar las páginas del man del smbclient.

Compartir Impresoras

Ahora compartiremos una impresora de cualquier tipo desde cualquier arquitectura con otros equipos.

Compartir Impresoras en Máquinas GNU/Linux a Máquinas Windows

Para compartir una impresora Linux con máquinas Windows, necesitas asegurarte de que la impresora está preparada para trabajar bajo Linux. Si puedes imprimir desde Linux, preparar una 'compartición' SMB de la impresora es automático.

Mírate el COMO Imprimir (Printing HOWTO) para poner a punto la impresora con Linux.

Añade la configuración de la impresora a tu smb.conf:

```
[global]
   printing = bsd
   printcap name = /etc/printcap
   load printers = yes
   log file = /var/log/samba-log.%m
   lock directory = /var/lock/samba

[printers]
   comment = Todas las impresoras
   security = server
   path = /var/spool/lpd/lp
   browseable = no
   printable = yes
   public = yes
   writable = no
   create mode = 0700

[ljet]
   security = server
   path = /var/spool/lpd/lp
   printer name = lp
   writable = yes
   public = yes
   printable = yes
   print command = lpr -r -h -P %p %s
```

¡Asegúrate de que el 'path' de la impresora (en este caso bajo [ljet]) se corresponde al directorio de 'spool' en /etc/printcap!

NOTA: Hay algunos problemas compartiendo impresoras conectadas a UNIX con máquinas Windows NT usando Samba. Un problema es que NT 'vea' la impresora compartida correctamente. Para conseguirlo, mírate las notas en la distribución de Samba en el fichero docs/WinNT.txt. El otro va con problemas con las claves. Mírate los comentarios en ese mismo fichero para conseguir una molesta ganancia de conocimientos y fallos para arreglar el problema.

Compartir Impresoras en Máquinas Windows a Máquinas GNU/Linux

Para compartir una impresora en una máquina Windows, debes hacer lo siguiente:

a) Debes tener las entradas adecuadas en /etc/printcap y deben corresponderse a la estructura de directorios local (el directorio de spool, etc)

b) Debes tener el script /usr/bin/smbprint. Viene con las fuentes de Samba, pero no con la distribución de ejecutables del Samba. Más abajo comentamos una copia ligeramente modificada.

c) Si quieres convertir ficheros ASCII a PostScript, debes tener el 'nenscript' o su equivalente. nenscript es un conversor de PostScript y habitualmente está instalado en /usr/bin.

d) Puedes desear que las impresiones de Samba sean más sencillas teniendo un front end fácil de usar. Más abajo tienes un sencillo script en perl para manejar ASCII, PostScript o PostScript generado.

La entrada para /etc/printcap que tenemos debajo es para una impresora HP 5MP en un host Windows NT. Las entradas son las siguientes:

```
        cm - comentario
        lp - nombre del dispositivo a abrir para salida
        sd - el directorio de spool de la impresora (en la máquina local)
        af - el fichero de cuentas
        mx - el tamano maximo del fichero (cero es ilimitado)
        if - nombre del fichero de entrada (script)
```

Para más información, lee el COMO Imprimir (Printing HOWTO) o la página del man de printcap.

```
# /etc/printcap
#
# //zimmerman/oreilly via smbprint
#
lp:\
        :cm=HP 5MP PostScript OReilly en zimmerman:\
        :lp=/dev/lp1:\
        :sd=/var/spool/lpd/lp:\
        :af=/var/spool/lpd/lp/acct:\
        :mx#0:\
        :if=/usr/bin/smbprint:
```

Asegúrate de que los directorios de spool y cuentas existe y se puede escribir en ellos. Asegura también que la línea 'if' tiene el path adecuado para el script smbprint (que damos debajo) y que apunta al dispositivo adecuado. (el fichero /dev especial).

Lo siguiente es el propio script smbprint. Normalmente está en /usr/bin y es atribuible a Andrew Tridgell, la persona que creó el Samba (que yo sepa). Viene con la distribución de las fuentes del Samba, pero está ausente de algunas distribuciones de ejecutables, por lo que lo he recreado aquí.

Te podría interesar mirarlo con cuidado. Hay algunas pequeñas alteraciones que han demostrado ser útiles.

```
#!/bin/sh -x

# Este script es un filtro de entrada para la impresion de printcap en
# una maquina unix. Usa el programa smbclient para imprimir un fichero
# en el servidor y servicio basados en smb especificados.
# Por ejemplo, puedes tener una entrada en printcap como esta
#
# smb:lp=/dev/null:sd=/usr/spool/smb:sh:if=/usr/local/samba/smbprint
#
# que creara una impresora unix llamada "smb" que imprimira a traves de
# este script. Necesitarras crear el directorio de spool /usr/spool/smb
# con los permisos y pertenencias apropiados para tu sistema.

# Ahora preparalos con el servidor y servicio en que quieras imprimir
# En este ejemplo tengo un PC WfWg llamado "lapland" que tiene una
# impresora exportada llamada "printer" sin password.

#
# Script alterado por hamilton@ecnz.co.nz (Michael Hamilton)
# para que servicio, servidor y clave puedan ser leidos desde un
# fichero /usr/var/spool/lpd/PRINTNAME/.config
#
# Para que esto funcione la entrada en /etc/printcap debe incluir un
# fichero de cuentas (af=...):
#
#   cdcolour:\
#       :cm=CD IBM Colorjet on 6th:\
#       :sd=/var/spool/lpd/cdcolour:\
#       :af=/var/spool/lpd/cdcolour/acct:\
#       :if=/usr/local/etc/smbprint:\
#       :mx=0:\
#       :lp=/dev/null:
#
# El fichero /usr/var/spool/lpd/PRINTNAME/.config deberia contener:
#   servidor=SERVIDOR_PC
#   servicio=NOMBRE_IMP
#   clave="clave"
#
```

```
# Ej.
#   servidor=PAULS_PC
#   servicio=CJET_371
#   clave=""

#
# Fichero de registro para correcciones, cambiar a /dev/null si se quiere
#
fichreg=/tmp/smb-print.log
# fichreg=/dev/null


#
# El ultimo parametro para el filtro es el nombre del fichero de
# cuentas
#
dir_spool=/var/spool/lpd/lp
fich_config=$dir_spool/.config

# Deberia leer las siguientes variables activadas en el fichero de
# configuracion:
#   servidor
#   servicio
#   clave
#   usuario
eval `cat $fich_config`

#
# Algo de ayuda en la correccion de errores. Cambia el >> por > si
# quieres salvar algo de espacio.
#
echo "servidor $servidor, servicio $servicio" >> $fichreg

(
# NOTA: Puede que quieras anadir la linea `echo translate' si quieres CR/LF
# automatico cuando imprimes.
        echo translate
        echo "print -"
        cat
) | /usr/bin/smbclient "\\\\$servidor\\$servicio" $clave -U $usuario -N -P >> $fichreg
```

La mayoría de las distribuciones de Linux vienen con el nenscript para convertir los documentos ASCII a Postscript. El siguiente script perl hace la vida más fácil dando un interfaz sencillo para que linux imprima a través de smbprint.

```
------------------------------------------------------------------------------

Forma de uso: print [-a|c|p] nombre_fichero
              -a imprime nombre_fichero como ASCII
              -c imprime nombre_fichero formateado como codigo fuente
              -p imprime nombre_fichero como Postscript
      Si no se pasa ningun parametro, print intenta
      averiguar el tipo de fichero e imprimirlo
      adecuadamente.


------------------------------------------------------------------------------
```

smbprint tiende a truncar las líneas demasiado largas cuando imprime ficheros ASCII. Este rompe las líneas largas donde haya un espacio en blanco (en lugar de en mitad de una palabra), si es posible.

El formateado de código fuente se hace con nenscript. Coge un fichero ASCII y lo formatea en 2 columnas con una cabecera mu' mona (fecha, nombre de fichero , etc). Además enumera las líneas. Usándolo como ejemplo, se pueden lograr otros tipos de formateado.

Los documentos Postscript también se imprimen correctamente, por lo que pasan directamente.

```
------------------------------------------------------------------------------


#!/usr/bin/perl

# Script:   print
# Autores:  Brad Marshall, David Wood
#           Plugged In Communications
# Fecha:    960808
# Cambios:  Ricardo Javier Cardenes Medina
# Razon:        Traduccion de comentarios y codigo a espanol para
#               mayor comprension.
# Fecha:    961109 (Sab 9 de Noviembre de 1996)
#
# Script para imprimir a Oreilly que esta actualmente en zimmerman.
# Proposito: Toma ficheros de varios tipos como parametros y
# los procesa apropiadamente para mandarlos al script de impresion de Samba.
#
# Tipos soportados actualmente:
#
# ASCII      - Asegura que las lineas con mas de $largo_linea caracteres seran
#              divididas aprovechando los espacios en blanco.
# PostScript - No hace nada.
# Codigo     - Lo formatea en PostScript (usando nenscript) para una mejor
#              presentacion (fuente, etc...).
```

```
#

# Maxima longitud permitida para cada linea de texto ASCII.
$largo_linea = 76;

# Path y nombre del script 'print' de Samba.
$prog_print = "/usr/bin/smbprint";

# Path y nombre del nenscript (el convertidor ASCII-->PostScript)
$nenscript = "/usr/bin/nenscript";

unless ( -f $prog_print ) {
        die "¡No encuentro $prog_print!";
}
unless ( -f $nenscript ) {
        die "¡No encuentro $nenscript!";
}

InterpLinCom(@ARGV);

# DBG
print "El tipo de fichero es $tipofich\n";

if ($tipofich eq "ASCII") {
        Rompe($largo_linea);
} elsif ($tipofich eq "codigo") {
        FormateaCodigo;
} elsif ($tipofich eq "postscript") {
        CreaTabla;
} else {
        print "Lo siento..tipo de fichero desconocido.\n";
        exit 0;
}
# Enviar el array a smbprint
open(IMPRESORA, "|$prog_print") || die "¡No puedo abrir $prog_print: $!\n";
foreach $linea (@newlines) {
        print IMPRESORA $linea;
}
# Enviar un avance de linea extra en caso de que el fichero tenga una
# ultima linea incompleta.
print IMPRESORA "\n";
close(IMPRESORA);
print "Terminado\n";
exit 0;

# ------------------------------------------------- #
#         Todo lo de debajo es una subrutina        #
# ------------------------------------------------- #

sub InterpLinCom {
        # Interpreta la linea de comando, averiguando el tipo de fichero

        # Toma $par y $fich como parametro (si existe) y nombre del
```

```
        # fichero.
        if ($_  0) {
               FormaUso;
        }
        # DBG
        #       foreach $elemento (@_) {
        #               print "*$elemento* \n";
        #       }

        $par = shift(@_);
        if ($par =~ /\-./) {
                $com = $par;
        # DBG
        #       print "\Encontrado $com.\n";

                $fich = shift(@_);
        } else {
                $fich = $par;
        }

        # Defining the file type
        unless ($com) {
                # No tenemos parámetro

                if ($fich =~ /\.ps$/) {
                        $tipofich = "postscript";
                } elsif ($fich =~ /\.java$|\.c$|\.h$|\.pl$|\.sh$|\.csh$|\.m4$|\.inc$|\.html$|\.
                        $tipofich = "codigo";
                } else {
                        $tipofich = "ASCII";
                }

                # Procesa $fich buscando el tipo y devuelve $tipofich
        } else {
                # Tenemos el tipo en $par
                if ($com =~ /^-p$/) {
                        $tipofich = "postscript";
                } elsif ($com =~ /^-c$/) {
                        $tipofich = "codigo";
                } elsif ($com =~ /^-a$/) {
                        $tipofich = "ASCII"
                }
        }
}

sub FormaUso {
        print "
Forma de uso: print [-a|c|p] nombre_fichero
              -a imprime nombre_fichero como ASCII
              -c imprime nombre_fichero formateado como codigo fuente
              -p imprime nombre_fichero como Postscript
      Si no se pasa ningun parametro, print intenta
      averiguar el tipo de fichero e imprimirlo
```

```
      adecuadamente.\n
";
        exit(0);
}

sub Rompe {
        # Crea una tabla con las lineas del fichero, donde cada linea es
        # menor que el numero de caracteres especificado, y las rompe
        # solo en los espacios en blanco

        # Toma el numero de caracteres a los que limitar la linea.
        $limite = pop(@_);

        # DBG
        #print "Entrando en la subrutina Rompe\n";
        #print "El limite de caracteres es $limit\n";

        # Lee en el fichero, lo interpreta y pone en la tabla.
        open(FICHERO, "$fich") || die "¡No puedo abrir $fich: $!\n";
        while(FICHERO) {
                $linea = $_;

                # DBG
                #print "La linea es:\n$linea\n";

                # Rompe la linea si se pasa del limite.
                while ( length($linea) > $limite ) {

                        # DBG
                        #print "Rompiendo...";

                        # Toma los primeros $limite +1 caracteres.
                        $cacho = substr($linea,0,$limite +1);

                        # DBG
                        #print "La linea parcial es:\n$cacho\n";

                        # Mira a ver si el ultimo caracter es un espacio.
                        $ultimo_car = substr($cacho,-1, 1);
                        if ( " " eq $ultimo_car ) {
                            # Si lo es, imprime el resto.

                            # DBG
                            #print "El ultimo caracter era un espacio\n";

                            substr($linea,0,$limite + 1) = "";
                            substr($cacho,-1,1) = "";
                            push(@newlines,"$cacho\n");
                        } else {
                             # Si no lo es, busca el ultimo espacio en la
                             # sub-linea e imprime hasta alli.

                            # DBG
```

```
                                #print "El ultimo caracter no era un espacio\n";

                                 # Borra el ultimo caracter que pasa de $limite
                                 substr($cacho,-1,1) = "";
                                 # Da la vuelta a la linea para hacer mas
                                 # sencillo buscar el espacio.
                                 $cachoreves = reverse($cacho);
                                 $indice = index($revpart," ");
                                 if ( $indice > 0 ) {
                                   substr($linea,0,$limite-$indice) = "";
                                   push(@newlines,substr($cacho,0,$limite-$indice)
                                       . "\n");
                                 } else {
                                     # No hay espacios en la linea, por lo que
                                     # se imprime hasta $limite.
                                   substr($linea,0,$limite) = "";
                                   push(@newlines,substr($cacho,0,$limite)
                                       . "\n");
                                 }
                            }
                    }
                    push(@newlines,$linea);
            }
            close(FICHERO);
    }

    sub FormateaCodigo {
            # Llama a la subrutina Rompe cuando filtra a traves de nenscript
            Rompe($largo_linea);

            # Manda los resultados a traves de nenscript para crear un
            # fichero Postscript que de una forma decente a nuestro codigo
            # fuente para imprimirlo (fuente Courier, numero de lineas, ...).
            # E imprime todo esto a un fichero temporal.
            $fichtmp = "/tmp/nenscript$$";
            open(FICHERO, "|$nenscript -2G -i$fich -N -p$fichtmpfich -r") ||
                    die "¡No pude abrir nenscript: $!\n";
            foreach $linea (@newlines) {
                    print FICHERO $linea;
            }
            close(FICHERO);

            # Vuelca el fichero temporal en una tabla para que pueda
            # ser pasado al script de impresion de Samba.
            @newlines = ("");
            open(FICHERO, "ichtmp") || die "¡No puedo abrir $fichtmp: $!\n";
            while(FICHERO) {
                    push(@newlines,$_);
            }
            close(FICHERO);
            system("rm $fichtmp");
    }
```

```
sub CreaTabla {
        # Crear la tabla para postscript
        open(FICHERO, "fich") || die "¡No puedo abrir $fich: $!\n";
        while(FICHERO) {
                push(@newlines,$_);
        }
        close(FICHERO);
}
```

Cómo Montar un PDC en Linux usando Samba

¿Por qué usar dominios? La respuesta no parece obvia hasta que miras detrás del telón, sobre todo con Windows 9X. Hay que remarcar que, con los grupos tradicionales, Windows 9X simplemente acepta cualquier usuario y contraseña que utilices al acceder al equipo. En Windows 9X no existen los usuarios no autorizados; cuando un usuario nuevo accede al sistema, éste simplemente le pide una contraseña nueva y lo autentifica contra esa misma contraseña. La única vez que Windows 9X intenta utilizar la contraseña es cuando te conectas a otro recurso.

Por otro lado, la autenticación en los dominios es similar a los sistemas Unix. Para poder acceder al dominio son necesarios un nombre de usuario y una contraseña válidos, que son autenticados a través de la base de datos de contraseñas del controlador primario de dominio. Si la contraseña no es válida, se notifica inmediatamente al usuario y este no puede acceder al dominio. Hay más buenas noticias: una vez que te has conectado al dominio, puedes acceder a cualquiera de los recursos de este para los que tengas derechos sin tener que reautenticarte. Dicho de otra forma, el PDC devuelve una señal al cliente que le permite acceder a cualquier recurso sin tener que consultar otra vez al PDC. Seguro que ya te has dado cuenta de la gran reducción que esto supone para el tráfico de la red, sin embargo puedes desactivar esto a través de la opción revalidate.

Configurando Samba para los Dominios Windows

Si tienes clientes Windows 2000 en tu red, hay varios pasos más que has de seguir para que Samba pueda actuar como PDC para ellos.

AVISO: Necesitaras usar Samba 2.1 o superior para asegurar que pueda funcionar como PDC para los clientes Windows 2000. Antes de Samba 2.1, para los clientes Windows 2000 solo estaba disponible una autenticación limitada. En el momento en que se imprimió este libro, la ultima versión de Samba era la 2.0.5, pero la 2.1 esta disponible a través de una descarga CVS.

Necesitamos asegurarnos de que Samba es el PDC para el grupo de trabajo actual y que esta usando la seguridad a nivel de usuario. Además, has de asegurarte de estar usando contraseñas encriptadas. Dicho de otra forma, modifica las opciones [global] del ejemplo anterior para incluir la opción encrypted passwords=yes:

```
[global]
workgroup = SIMPLE
encrypted passwords = yes
```

```
domain logons = yes
security = user
```

Crear cuentas de confianza para los clientes Windows 2000

Este paso es, exclusivamente, para los clientes Windows 2000. Todos los clientes Windows 2000 que se conectan a un PDC hacen uso de las cuentas de confianza. Estas cuentas permiten a una maquina conectarse con el PDC (y no con uno de sus recursos), lo que significa que el PDC puede verificar posteriores conexiones de los usuarios desde ese cliente. A nivel de utilización, una cuenta de confianza es idéntica a una cuenta de usuario. De hecho, vamos a utilizar cuentas de usuario estándar para simular cuentas de confianza.

El nombre de usuario de una cuenta de confianza para un equipo es el nombre del equipo con un signo de dólar añadido a el. Por ejemplo, si nuestra maquina windows 2000 se llama chimera el nombre de la cuenta será chimera$. La contraseña inicial de la cuenta será el nombre del equipo en minúsculas. Para reforzar la cuenta de confianza en el servidor Samba, necesitarás crear una cuenta Unix con el nombre de la maquina y una entrada con una contraseña encriptada en el fichero smbpasswd. Vamos a ver la primera parte. Aquí, sólo necesitaremos modificar el fichero /etc/passwd para soportar las cuentas de confianza; no hay necesidad de crear un directorio personal o asignar un shell al usuario porque solo estamos interesados en que se permita el acceso (login).

Por tanto, podemos crear una cuenta tonta con la siguiente entrada:

```
oscar$:*:1000:900:Trust Account:/dev/null:/dev/null
```

Fíjate en que hemos desactivado el campo de contraseña insertando un *. Esto es porque Samba usará el fichero smbpasswd para guardar la contraseña, y no deseamos que nadie haga un telnet a la maquina usando esa cuenta. De hecho, el único valor aparte del nombre de la cuenta es el UID de la misma para la base de datos de contraseñas (1000). Este numero debe apuntar a un único ID de recurso en el servidor Windows 2000 y no puede entrar en conflicto con ningún otro ID. Por ello, ningún usuario o grupo de Windows 2000 puede apuntar a este recurso o se producirá un error de red.

Ahora, añade la contraseña encriptada usando el comando smbpasswd, como sigue:

```
# smbpasswd –a –m oscar

Added user oscar$
Password changed for user oscar$
```

La opción -m especifica que se está creando una cuenta de confianza. El programa smbpasswd establecerá la contraseña encriptada inicial como el nombre NetBIOS del ordenador en minúsculas; no necesitas introducirla. Cuando utilices esta opción en la línea de comandos, no pongas el signo de dólar ($) después del nombre del ordenador, se añadirá automáticamente. Una vez que se añade la contraseña encriptada, Samba está preparado para manejar accesos al dominio desde un cliente Windows 2000.

Más Información

- Página web de Samba: www.samba.org (http://www.samba.org/)
- FAQ´s de Samba: Samba Faq-O-Matic (http://samba.sourceforge.net/fom-serve/cache/1.html)
- Proyecto S.O.B.L. (HOWTOS): http://www.samtek.es/howtos/samba/samba-howto.html
- Using Samba: O´Reilly Books "Using Samba" (http://www.oreilly.com/catalog/samba/chapter/book/)
- Documentación de la propia distribución.

Sobre este documento

Se otorga permiso para copiar, distribuir y/o modificar este documento bajo los términos de la Licencia de Documentación Libre GNU, versión 1.1 o cualquier versión posterior publicada por la Free Software Foundation. Puedes consultar una copia de la licencia en http://www.gnu.org/copyleft/fdl.html (http://www.gnu.org/copyleft/fdl.html)

Este documento ha sido escrito en formato XML utilizando la DTD de DocBook (http://www.docbook.org). Mediante este sistema, puede ser fácilmente transformado a múltiples formatos (HTML, TXT, PDF, PostScript, LaTeX, DVI, ...). Se recomienda su utilización como herramienta de documentación potente y libre.