SMB HOWTO

David Wood, *dwood@plugged.net.au* v1.3, 20 aprile 2000

Questo è lo SMB HOWTO. Il documento descrive come usare il protocollo Session Message Block (SMB), chiamato anche protocollo NetBIOS o LanManager, con Linux. Traduzione italiana e manutenzione: Andrea Girotto (*andrea.girotto@era-net.it*)

Indice

1 Licenza **1**

2 Introduzione **2**

3 Altre informazioni **3**

4 Installazione **3**

5 Eseguire i demoni **4**

6 Configurazione generale (`/etc/smb.conf`). **6**

7 Condividere un disco Linux con macchine Windows **7**

8 Condividere un disco Windows con macchine Linux **9**

9 Condividere una stampante Linux con macchine Windows **11**

10 Condividere una stampante Windows con macchine Linux **13**

11 Eseguire copie di sicurezza (backup) di macchine Windows su un server Linux **22**

12 Usare Samba attraverso reti instradate **26**

13 Ringraziamenti **27**

1 Licenza

2 Introduzione

Questo è lo SMB HOWTO. Il documento descrive come usare il protocollo *Session Message Block* (SMB), chiamato anche protocollo NetBIOS o LanManager, con Linux usando Samba. Nonostante tale documento sia basato su Linux, è possibile eseguire Samba sulla maggior parte dei sistemi operativi Unix e simili.

Questo documento è gestito da David Wood (*dwood@plugged.net.au*). Aggiunte, modifiche o correzioni per la versione successiva di questo documento, possono essere spedite (in inglese) all'indirizzo indicato.

Prima di richiedere aiuto, si prega di consultare il sito di Samba localizzato presso: *http://www.samba.org/* `http://www.samba.org/` , che dispone di molta più documentazione. È presente una quantità enorme di informazioni. Si può anche cercare di leggere il newsgroup comp.protocols.smb.

Il protocollo SMB è usato da Microsoft Windows 3.11, NT e 95/98 per condividere dischi e stampanti. Usando la suite di programmi Samba di Andrew Tridgell (*Andrew.Tridgell@anu.edu.au*), macchine Unix (compreso Linux) possono condividere dischi e stampanti con host Windows. Paal-Kr. Engstad (*engstad@intermetrics.com*) e Volker Lendecke (*lendecke@namu01.gwdg.de*) hanno sviluppato lo strumento smbfs che consente a macchine Unix di montare condivisioni SMB da macchine Windows oppure host Samba.

Il pacchetto **Samba** sostanzialmente consente di:

1. Condividere un disco Linux con macchine Windows.
2. Condividere un disco Windows con macchine Linux.
3. Condividere una stampante Linux con macchine Windows.
4. Condividere una stampante Windows con macchine Linux.

Tutte queste possibilità sono discusse in questo documento.

Attenzione: Le procedure, i programmi e gli script riportati, o si sono dimostrati funzionanti per l'autore, o sono stati riportati esserlo dalle persone che li hanno forniti; con configurazioni differenti potrebbero non funzionare; se ci si trova in tale situazione, è possibile spedire un *email* all'autore con i suggerimenti per migliorare questo documento.

Si prega di porre attenzione al fatto che macchine Windows 3.x, affinché possano accedere a condivisioni SMB, devono avere uno stack TCP/IP e le DLL Win32s. Entrambi questi pacchetti sono disponibili presso il sito di Microsoft (*http://www.microsoft.com* `http://www.microsoft.com`). Al momento in cui si scrive questa versione dell'HOWTO, è riportato che Microsoft richiede la sottoscrizione a Microsoft Software Developers Network (MSDN) al fine di prelevare lo stack TCP/IP-32 per Windows 3.x dal loro sito. Dal momento che tale software era distribuito gratuitamente, dovrebbero essere diffuse diverse copie meno recenti probabilmente presso amici o tramite gruppi di utenti.

3 Altre informazioni

Questo HOWTO spiega come configurare servizi minimi di file e di stampa per SMB su una macchina Linux. **Samba** è un pacchetto piuttosto completo e complesso. Non sarà fatto alcun tentativo di duplicare tutta la documentazione **Samba**.

Per maggiori informazioni, consultare i seguenti documenti:

- La documentazione **Samba**, disponibile come parte del pacchetto. La distribuzione è disponibile presso: *ftp://ftp.samba.org/*
- Il *Printing HOWTO* di Linux (*in versione italiana presso PLUTO*).
- Protocol Standard For A NetBIOS Service On A TCP/UDP Transport.

  RFC 1001
  > RFC 1001 - Concepts and Methods.

  RFC 1002
  > RFC 1002 - Detailed Specifications.

4 Installazione

Per prima cosa, al fine di poter usare Samba sulle proprie macchine è necessario utilizzare un singolo segmento LAN ethernet ed usare il protocollo TCP/IP. Samba non funzionerà con altri protocolli di rete. Generalmente è facile soddisfare questa richiesta, dal momento che Linux e Windows 95/98/NT sono distribuiti con il TCP/IP. Tuttavia, se si usa Windows 3.x è necessario aggiungere il supporto TCP/IP. Una delle domande poste più di frequente è perché Samba "non funziona" con sistemi Windows senza il protocollo TCP/IP.

Per installare TCP/IP con Windows 95/98, si deve selezionare Pannello di controllo | Rete, aggiungere e configurare Microsoft TCP/IP. Con Windows NT, si selezioni Pannello di Controllo | Rete | Protocolli.

Per prelevare l'ultima versione di **Samba** si deve selezionare il mirror più vicino, dalla lista disponibile presso:

ftp://ftp.samba.org/

Nella maggior parte dei casi, le distribuzioni Linux dispongono già dei pacchetti di una versione recente di Samba.

Il pacchetto **Samba** richiede due demoni. Tipicamente sono installati in `/usr/sbin` ed eseguiti o in fase di avvio dagli *script di inizializzazione* del sistema o da **inetd**. Script di esempio sono elencati in 5 (Eseguire i demoni).

```
smbd (Il demone SMB)
nmbd (Fornisce supporto di NetBIOS nameserver per i client)
```

Si noti che il servizio dei nomi fornito dal demone nmbd è differente dal servizio fornito da Domain Name Service (DNS). Il servizio di nomi NetBIOS usato da SMB è "stile-Windows". In altre parole, avere un DNS attivo, non ha nulla a che vedere con la possibilità di Samba di risolvere i nomi dei computer.

Di solito (la posizione è opzionale) i seguenti programmi di **Samba** sono installati in `/usr/bin` oppure `/usr/local/samba/bin`:

smbclient
Client SMB per macchine UNIX

smbprint

script per usare la stampante di un host SMB

smbprint.sysv

Come `smbprint` per macchine con UNIX SVR4

smbstatus

Elenca le connessioni SMB correnti per il computer locale

smbrun

Facilita esecuzioni di applicazioni su un host SMB

La distribuzione in formato binario per il supporto del file system smbfs, è discussa successivamente in questo documento.

Inoltre, è incluso in questo HOWTO uno script chiamato 10 (print), da usare come interfaccia per `smbprint`.

Il pacchetto **Samba** è semplice da installare. Per prima cosa è necessario recuperare il sorgente dal sito elencato precedentemente e leggere il file `README` della distribuzione. C'è anche un file chiamato `docs/INSTALL.txt` che fornisce istruzioni passo-a-passo.

Seguendo l'installazione, si pongono i demoni in `/usr/sbin` ed i programmi in `/usr/bin`. Le pagine del manuale in `/usr/local/man`.

Compilando il pacchetto **Samba**, si specifica nel `Makefile` la locazione del file di configurazione `smb.conf`, generalmente `/etc`, ma può essere una qualsiasi. Per questo documento si suppone che tale file di configurazione sia `/etc/smb.conf`; altri file di configurazione: `/var/log/samba-log.%m` e *lock directory* `/var/lock/samba`.

Si installi il file di configurazione, `smb.conf`. Nella directory dove si è compilato Samba è presente una sottodirectory `examples/simple` e si legga il file README. Si copi il file `smb.conf` in `/etc`. ATTENZIONE! Se si ha una distribuzione Linux con già Samba installato, potrebbe già essere presente un file di configurazione in `/etc`, sarebbe meglio iniziare con quello.

Se non si desidera avere il file di configurazione in `/etc`, può essere messo in una qualsiasi posizione ed eventualmente collegato con un *link* simbolico in `/etc`:

```
ln -s /percorso/di/smb.conf /etc/smb.conf
```

5 Eseguire i demoni

I due demoni SMB sono: `/usr/sbin/smbd` e `/usr/sbin/nmbd`. Nella maggior parte delle distribuzioni, questi possono essere fatti partire, interrompere o riavviare attraverso script di avvio come `/etc/rc.d/init.d/smb` e collegati in modo simbolico agli appropriati runlevel.

Nel caso si decida di non usare i script di avvio, è possibile eseguire i demoni di Samba da inetd o come singoli processi. Samba risponderà leggermente più velocemente come demone isolato rispetto a quando eseguito tramite inetd.

In entrambi i casi, è necessario verificare che il file `/etc/services` presenti linee simili a queste:

```
netbios-ns      137/tcp     nbns
netbios-ns      137/udp     nbns
netbios-dgm     138/tcp     nbdgm
netbios-dgm     138/udp     nbdgm
netbios-ssn     139/tcp     nbssn
```

Si faccia attenzione che non siano commentate. A seconda della distribuzione usata, potrebbe anche essere necessario aggiungerle. Samba non sarà in grado di collegarsi alle porte appropriate a meno che il file `/etc/services` non presenti tali linee.

Per eseguire i demoni da `inetd`, si deve aggiungere nel file di configurazione, `/etc/inetd.conf`:

```
# Servizi NetBIOS SAMBA (per condivisione di file e stampanti con PC)
netbios-ssn stream tcp nowait root /usr/sbin/smbd smbd
netbios-ns dgram udp wait root /usr/sbin/nmbd nmbd
```

Quindi far ripartire il demone di `inetd` con il comando:

```
kill -HUP `cat /var/run/inetd.pid`
```

Per eseguire i demoni dagli script di inizializzazione del sistema, si possono mettere le istruzioni seguenti in un file chiamato `/etc/rc.d/init.d/smb` (nella maggior parte delle distribuzioni) e fare collegamenti simbolici a file indicati nei commenti:

```
#!/bin/sh

#
# /etc/rc.d/init.d/smb - inizia e termina i servizi SMB.
#
# file che seguono devono essere link simbolici a questo file:
# symlinks: /etc/rc.d/rc1.d/K35smb  (termina i servizi SMB durante
#                                     shutdown)
#           /etc/rc.d/rc3.d/S91smb  (inizia i servizi SMB in
#                                     multiuser mode)
#           /etc/rc.d/rc6.d/K35smb  (Termina i servizi SMB prima
#                                     di reboot)
#

# Sorgente per funzioni di libreria.
. /etc/rc.d/init.d/functions

# Sorgente per la configurazione di rete.
. /etc/sysconfig/network

# Verifica che la rete sia configurata.
[ ${NETWORKING} = "no" ] && exit 0

# Controlla il tipo di chiamata.
case "$1" in
  start)
    echo -n "Attivati servizi SMB: "
    daemon smbd -D
    daemon nmbd -D
    echo
    touch /var/lock/subsys/smb
    ;;
 stop)
    echo -n "Terminati servizi SMB: "
    killproc smbd
    killproc nmbd
    rm -f /var/lock/subsys/smb
```

```
        echo ""
        ;;
    *)
        echo "Uso: smb {start|stop}"
        exit 1
esac
```

Se all'avvio di Samba si presenta un errore che segnala l'impossibilità da parte del demone a collegarsi alla porta 139, allora probabilmente è presente in esecuzione un altro processo Samba che non è stato ancora interrotto. Verificare la lista dei processi (con 'ps auxww | grep mbd') per determinare se un altro servizio Samba è in esecuzione.

6 Configurazione generale (`/etc/smb.conf`).

La configurazione di **Samba** su Linux (o su altre macchine UNIX) è controllata da un file soltanto, `/etc/smb.conf`, che determina quali risorse di sistema si desidera condividere con il mondo esterno e quali restrizioni attivare.

Ai fini di introduzione, il primo file `smb.conf` mostrato, sarà il più semplice possibile per concentrare l'attenzione su come condividere dischi e stampanti Linux con macchine Windows. Nei paragrafi successivi si introdurranno altri concetti.

Ciascuna sezione del file inizia con un'intestazione come **[global]**, **[homes]**, **[printers]**, ecc.

La sezione **[global]** imposta le variabili usate da **Samba** per definire la condivisione di ogni altra risorsa.

La sezione **[homes]** consente ad un utente remoto accesso alla propria (e solo a quella) *home directory* sulla macchina locale (Linux). Quindi se un utente Windows cerca di connettersi a tale condivisione da una macchina Windows, verrà collegato alla propria *home directory*. Si noti che per poter accedere alla condivisione è necessario avere un *account* sulla macchina Linux.

Il file di esempio `smb.conf` fornito, consente ad utenti remoti di accedere alla propria *home directory* dalla macchina locale e di scrivere in una directory temporanea. Affinché un utente Windows veda queste condivisioni, è necessario che la macchina Linux sia sulla rete locale. Quindi l'utente semplicemente connette un disco di rete dal File Manager di Windows oppure da Windows Explorer.

Nelle sezioni che seguono saranno fornite specificazioni aggiuntive per la configurazione al fine di condividere maggiori risorse.

```
; /etc/smb.conf
;
; Dopo aver modificato questo file assicurarsi di far ripartire il
; server, ad esempio:
; /etc/rc.d/init.d/smb stop
; /etc/rc.d/init.d/smb start

[global]
; Attivare questa linea se si desidera un account generico
; guest account = nobody
   log file = /var/log/samba-log.%m
   lock directory = /var/lock/samba
   share modes = yes

[homes]
   comment = Directory Personali
   browseable = no
```

```
   read only = no
   create mode = 0750

[tmp]
   comment =  Spazio per file temporanei.
   path = /tmp
   read only = no
   public = yes
```

Avendo scritto un nuovo smb.conf, è utile verificare la correttezza. È possibile usare l'utility 'testparm' per controllare un file smb.conf (pagina del manuale: testparm); se non viene riportato alcun problema, smbd caricherà correttamente il file di configurazione.

Un aiuto prezioso: se il server Samba dispone di più di una interfaccia ethernet, il demone smbd potrebbe collegarsi a quella sbagliata. In tal caso, è possibile forzare il comportamento desiderato, aggiungendo nella sezione [global] una linea simile a:

```
interfaces = 192.168.1.1/24
```

sostituendo al posto dell'indirizzo IP quello assegnato all'interfaccia ethernet corretta. Il valore "24" è corretto per una rete di classe C, ma dovrebbe essere ricalcolato nel caso di sottoreti. Tale numero si riferisce alla maschera di rete. Numeri per altre classi o reti sono disponibilie nell'IP-Masquerade mini-HOWTO.

È disponibile anche un tool grafico per la configurazione di Samba: GtkSamba. Si consulti *http://www.open-systems.com/gtksamba.html* `http://www.open-systems.com/gtksamba.html` .

7 Condividere un disco Linux con macchine Windows

Come dimostrato nel precedente `smb.conf`, è alquanto semplice condividere dischi Linux con utenti Windows. Come ogni altra cosa in **Samba**, è possibile controllare con notevole precisione il comportamento. Ecco alcuni esempi. Per condividere pubblicamente una directory, si può creare una copia della sezione **[tmp]** aggiungendo al `smb.conf` qualcosa come:

```
[public]
   comment = Pubblicamente Accessibile
   path = /home/public
   public = yes
   writable = yes
   printable = yes
```

Le seguenti modifiche rendono la directory precedente pubblicamente leggibile e scrivibile solo dalle persone del gruppo *staff*:

```
[public]
   comment = Pubblicamente Accessibile
   path = /home/public
   public = yes
   writable = yes
   printable = no
   write list = @staff
```

In passato era così semplice. Tuttavia, recentemente Microsoft ha reso la vita leggermente più difficile per chi usa Samba. Windows 98, Windows NT (service pack 3 e successivi) e le ultime versioni di Windows 95, usano in modo predefinito password criptate. Samba invece è impostato per password non criptate. Non è possibile consultare i server in modo anonimo, quando sia client che server usino password in codice, perché non è possibile fare una connessione anonima.

È possibile verificare se è presente un disaccordo sul tipo di password tra server e client, quando provando a connettersi ad una condivisione si presenta un dialog box con un messaggio del tipo "Non è possibile accedere a questo account da questa macchina".

È possibile configurare il server Samba per usare password criptate, o, in alternativa, configurare i sistemi Windows per usare password in chiaro.

Al fine di ottenere che Windows usi password in chiaro, usando l'editor di registro (regedit):

Windows 95/98

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\VNETSUP

Windows NT

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Rdr\Parameters

Windows 2000

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanWorkStation\Parameters

e si aggiunga un nuovo valore DWORD:

Value Name:

EnablePlainTextPassword

Data:

0x01

Una volta apportate queste modifiche, si faccia ripartire la macchina Windows e si cerchi di mappare un drive di rete sul server Samba. Dovrebbe funzionare, posto che il server Samba usi password in chiaro.

Per configurare Samba al fine di usare password criptate, si aggiungano nella sezione [global] del file `/etc/smb.conf` le seguenti linee:

```
encrypt passwords = yes
smb passwd file = /etc/smbpasswd
```

Prima di intraprendere una tale modifica si consiglia vivamente di leggere i file ENCRYPTION.txt, Win95.txt e WinNT.txt della documentazione Samba!

Se i client ed il server usano password criptate, non sarà possibile elencare le condivisioni sul server, fino a che non ne sia stata fatta una con appropriata autenticazione. Per avere la connessione iniziale, o si digiti manualmente il nome della condivisione nel File Manager di Windows, o nel dialog box di Explorer, nella forma '\\<nomecomputer>\<nomecondivisione>'. A questo punto ci si connetta al server con un username ed una password che siano validi per quella macchina.

Se si sospetta che il servizio dei nomi NetBIOS non sia correttamente configurato (magari perché si ottiene un errore del tipo 'host not found', quando si cerca di connettersi), si provi ad usare solo l'indirizzo IP del server: '\\<indirizzi>\<nomecondivisione>'.

Per ottenere che i nomi dei file appaiano correttamente, potrebbe anche essere necessario impostare alcune opzioni nelle appropriate sezioni di condivisione. Quelle che si indicano funzionano con Windows 95/98/NT ma probabilmente sarà necessario modificarle per i client Windows 3.x:

```
; Mangle case = yes seems to give the correct filenames
; for Win95/98/NT.
mangle case = yes

; se samba debba essere sensibile a maiuscole/minuscole
case sensitive = no

; impostazione predefinita per i file creati
default case = lower

; preserva maiuscole/minuscole per tutti i nomi di file
preserve case = yes

; preserva la capitalizzazione per nomi di file dos (8.3)
short preserve case = no
```

Consultare la documentazione **Samba** o le pagine del manuale per altre configurazioni particolari di condivisione.

8 Condividere un disco Windows con macchine Linux

Sistemi Linux (UNIX) possono anche sfogliare e montare condivisioni SMB. Si noti che questo è possibile sia con server Windows che con server Samba.

Con la distribuzione **Samba**, è incluso un programma *client* del protocollo SMB per macchine UNIX. Dispone di un'interfaccia a linea di comando simile ad `ftp`. Si può usare questo programma per trasferire file tra un *server* Windows ed un *client* Linux.

La maggior parte delle distribuzioni Linux ora include l'utile pacchetto smbfs, che consente di montare e smontare condivisioni SMB.

Per elencare le condivisioni disponibili su un dato *host*, è sufficiente eseguire:

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

dove `host` è il nome della macchina che si desidera interrogare. Questo comando ritorna una lista di nomi di '*servizi*', cioè nomi di dischi o stampanti che si possono condividere. A meno che il *server* SMB non sia configurato per nessuna sicurezza, sarà richiesto una password. Potrebbe essere essere la password per l'*account* `guest` di quella macchina o per il proprio personale.

Per esempio:

```
smbclient -L zimmerman
```

L'output di questo comando dovrebbe somigliare a qualcosa del tipo:

```
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
```

Il browse list elenca altri server SMB sulla rete con risorse da condividere.

Per eseguire il client:

```
/usr/sbin/smbclient servizio <password>
```

dove 'servizio' è il nome di macchina e servizio. Per esempio se si desidera raggiungere una directory condivisa come 'public' su una macchina chiamata 'zimmerman', il servizio dovrebbe essere chiamato: \\zimmerman\public. Per le restrizioni dell'interprete dei comandi (shell) potrebbe essere necessario raddoppiare le barre inverse, qualcosa come:

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

dove 'mypasswd' è la propria password.

A questo punto `smbclient` dovrebbe presentare la propria richiesta comandi:

```
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: \>
```

Scrivendo 'h' si ottiene aiuto per usare `smbclient`:

```
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: \>
```

Se già si conosce `ftp`, non dovrebbe essere necessario consultare le pagine del manuale di `smbclient`.

Nonostante si possa usare smbclient per verificare il funzionamento, è scomodo per l'uso comune. A questo punto potrebbe tornare utile il pacchetto smbfs. Smbfs dispone di due semplici utility, smbmount e smbumount che funzionano come mount e umount per condivisioni SMB.

Una cosa importante da notare: è necessario aver compilato il supporto SMB nel kernel per usare queste utility!

L'esempio seguente dimostra un tipico uso di smbmount per montare una condivisione SMB chiamata "customers" da una macchina di nome "samba1":

```
[root@postel]# smbmount "\\\\samba1\\customers" -U rtg2t -c 'mount /customers -u 500 -g 100'
Added interface ip=192.168.35.84 bcast=192.168.255.255 nmask=255.255.0.0
Got a positive name query response from 192.168.168.158 ( 192.168.168.158 )
Server time is Tue Oct  5 10:27:36 1999
Timezone is UTC-4.0
Password:
Domain=[IPM] OS=[Unix] Server=[Samba 2.0.3]
security=user
```

Il comando mount ora riporterà la condivisione montata, come se fosse un export NFS:

```
[root@postel]# mount
/dev/hda2 on / type ext2 (rw)
none on /proc type proc (rw)
none on /dev/pts type devpts (rw,mode=622)
//SAMBA1/CUSTOMERS on /customers type smbfs (0)
```

Si consultino le pagine del manuale per dettagli su smbmount e smbumount ed il relativo uso.

9 Condividere una stampante Linux con macchine Windows

Per condividere una stampante Linux con macchine Windows, è necessario accertarsi che la propria stampante sia correttamente configurata e funzioni con Linux. Se la stampa da Linux è possibile, è abbastanza semplice configurare una condivisione per stampante SMB.

Si noti che utenti Windows devono avere un account sulla macchina Linux/Samba al fine di poter stampare. Windows 95/98 cercherà di autenticarsi con il print server usando lo username e la password forniti per login all'avvio. Questo significa che se si seleziona 'Annulla' al LogOn di Windows, non sarà possibile stampare (o connettersi ad altri servizi SMB)! Windows NT consente di fornire username e password quando si usa una stampante.

Consultare il Printing HOWTO per configurare la stampa locale.

Aggiungere la configurazione di stampa al proprio `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 = All Printers
```

```
    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
```

Accertarsi che il percorso della stampante (in questo caso **[ljet]**) corrisponda alla *directory di spool* in `/etc/printcap`!

Le linee:

```
printcap name = /etc/printcap
load printers = yes
```

Controllano se tutte le stampanti in `/etc/printcap` debbano essere caricate in modo predefinito. In tal caso non c'è ragione per impostare le stampanti individualmente. La sezione [printers] specifica opzioni per le stampanti che si definiscono esplicitamente. Se il sottosistema di stampa in uso non funziona in questo modo (BSD), è necessario impostare un falso file printcap (oppure usare la tecnica del 'print command'). Per maggiori informazioni relativamente il sistema printcap, si consulti il Printing HOWTO.

Una tecnica utile per verificare le proprie connessioni è cambiare il comando di stampa in:

```
print command = cp %S /tmp/print.%P.%S
```

ed analizzare il file risultante.

NOTA: ci sono alcuni problemi per condividere stampanti di macchine UNIX con macchine Windows NT usando **Samba**. Un problema è che NT non vede correttamente la stampante condivisa. Per rimediare consultare le note nella distribuzione **Samba** nel file `docs/WinNT.txt`. Altri problemi hanno a che fare con le password. Vedere i commenti nello stesso file per una maggiore comprensione o in caso di problemi nella configurazione.

Oleg L. Machulskiy (*machulsk@shade.msu.ru*) ha suggerito un comando di stampa migliore da usare nell'esempio precedente:

```
print command = smb2ps %s | lpr -r -h -P %p
```

Dove 'smb2ps' è uno script che trasforma il file di spool ricevuto da Windows in un file Postscript usabile. È necessario eliminare le prime 3 linee e le ultime 2, perché contengono codice PJL oppure PCL.

Questo approccio è necessario solo se la macchina Windows stampa PCL e non realmente Postscript. L'autore ha scoperto che Windows 95/98/NT non hanno un generico driver Postscript, ma il "Digital turbo Printserver 20" si è dimostrato essere abbastanza generico per la maggior parte delle configurazioni. Inoltre pare che anche "Apple LaserWriter II NTX" funzioni correttamente per questo scopo.

Se si crea una directory di spool al posto di quella impostata dall'utility fornita con la distribuzione si faccia attenzione ai permessi. Neil Fraser (*neilf@necon.co.za*) suggerisce che i permessi della directory di spool (in questo caso `/var/spool/lpd/lpr` siano 4755 (notare il bit suid); funziona sul suo sistema quando il proprietario è 'root' ed il gruppo 'lp'.

Jeff Stern (*jstern@eclectic.ss.uci.edu*) riporta come impostazione della propria spool directory 777, per consentire ad utenti senza privilegi di stampare, anche se indica come avrebbe potuto aggiungere gli utenti al gruppo 'lp'. Questa decisione spetta al system administrator locale, se la sicurezza di stampa è necessaria, allora si limitino i permessi il più possibile. In un ambiente casalingo, probabilmente si desidera che tutti siano in grado di stampare.

Dr. Michael Langner (*langner@fiz-chemie.de*) indica come i problemi di permesso di scrittura in `/var/spool/lpd/` possono essere evitati se si usa qualcosa come "path = /tmp" e "print command = lpr -r -P%p %s".

A volte un errore di parsing del Postscript capita quando si stampa da macchine Windows e causa la stampa di una pagina extra. Tale pagina avrà sempre "%%[Lastpage]%%" in cima. Questo sembra accadere solo con Windows 95 e 98 a causa del Postscript malformato.

Un modo per gestire questa situazione è di usare uno script per rimuovere la sequenza Postscript errata. Un altro modo è cercare di trovare un driver Postscript migliore. Probabilmente il modo migliore è di usare LPRng invece del Postscript per stampare su un Server Samba.

Erik Ratcliffe (*erik@caldera.com*) di Caldera, riporta che LPRng comporta che qualsiasi sia il driver sulla macchina Windows, sul server Samba si usa un `/etc/printcap` simile a:

```
raw:\
        :rw:sh:
        :lp=/dev/lp1
        :sd=/var/spool/lpd/raw
        :fx=flp
```

LPRng non richiede un :\ alla fine di ogni riga. L'indicazione della stampante deve comunque essere presente in `/etc/smb.conf` per la stampante fisica. Il comando di stampa usa l'entry "raw" ed i dati sono mandati alla stampante in modo binario. Tale comando somiglia a:

```
print command = lpr -b -Praw %s
```

Potrebbe anche essere necessario disattivare lo spooling di stampa su Windows 95 al fine di stampare direttamente.

Se si ottiene sempre una pagina extra alla fine della stampa di client Windows, si provi ad aggiungere la direttiva sf in `/etc/printcap`. In questo modo si sopprimono i form feed per separare i job fi stampa, ma non si interferisce con i from feed dei documenti.

10 Condividere una stampante Windows con macchine Linux

Per condividere una stampante su una macchina Windows, è necessario:

1. Avere una configurazione corretta in `/etc/printcap` che corrisponda alla struttura delle directory locali (per le directory di spool, ecc.)
2. Usare lo script 10 (/usr/bin/smbprint). Questo script è disponibile con il sorgente di **Samba**, purtroppo non con tutte le distribuzioni in formato binario. Più avanti verrà presentata una versione leggermente diversa di 10 (smbprint).

3. Se si desidera convertire file ASCII in Postscript, è necessario avere `nenscript`, o qualcosa di equivalente. `nenscript` è un convertitore Postscript generalmente installato in `/usr/bin`.
4. Si potrebbe desiderare rendere la stampa con **Samba** più semplice tramite un semplice *front-end*. Di seguito è riportato un semplice script (10 (print)) in Perl per manipolare ASCII, Postscript...
5. Si potrebbe anche usare MagicFilter per ottenere il risultato. I dettagli necessari per configurare MagicFilter sono presenti nello script perl. MagicFilter ha dei punti di favore in quanto conosce automaticamente come convertire tra un gran numero di formati.

La successiva configurazione di `/etc/printcap` è valida per una stampante **HP 5MP** su un host Windows NT. Le linee hanno il seguente significato:

cm

commento

lp

dispositivo da aprire per output

sd

directory di spool della stampante sulla macchina locale

af

file per registrare le transazioni

mx

massima grandezza per un file (zero significa nessun limite)

if

nome del filtro di output (script)

Per maggiori informazioni consultare il *Printing HOWTO*

o le pagine del manuale per la voce *printcap*.

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

È necessario accertarsi che le directory di spool e per la registrazione delle transazioni (log) esistano e siano scrivibili. Assicurarsi che la linea 'if' contenga un percorso corretto allo script di **smbprint** (10 (dato oltre)) ed inoltre che il dispositivo puntato sia corretto (il file speciale `/dev`).

Di seguito lo script **smbprint**. Di solito è installato in `/usr/bin` ed è attribuibile, per quanto ne so, ad Andrew Tridgell, la persona che ha creato **Samba**. Viene fornito con la distribuzione in formato sorgente di **Samba** ma è riportato, essendo assente in certe distribuzioni in formato binario.

Potrebbe essere utile studiarlo attentamente, alcune piccole modifiche hanno dimostrato essere di enorme utilità.

```
#!/bin/sh -x

# Questo script è un filtro di input per la stampa via printcap da una
# macchina UNIX. Usa il programma smbclient per stampare il file sul
# server e servizio specificati usando il protocollo smb.
# Per esempio è possibile avere una linea del printcap del tipo:
#
# smb:lp=/dev/null:sd=/usr/spool/smb:sh:if=/usr/local/samba/smbprint
#
# Che dovrebbe creare una stampante UNIX chiamata "smb" che stamperà
# attraverso questo script. E' necessario creare la directory di spool
# /usr/spool/smb con permessi, proprietario e gruppo appropriati per il
# sistema.

# Assegnare i valori che seguono in modo da accordarsi al server e
# e servizio su cui si desidera stampare. In questo esempio si suppone
# di avere un PC con WfWg chiamato "lapland" che ha una stampante
# esportata chiamata "printer" senza password.
#
# Script modificato da hamiltom@ecnz.co.nz (Michael Hamilton)
# in modo da poter leggere server, servizio e password dal file
# /usr/var/spool/lpd/PRINTNAME/.config
#
# Affinché tutto funzioni è necessario che esista una linea in
# /etc/printcap che includa il file di accounting (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:
#
# Il file /usr/var/spool/lpd/PRINTNAME/.config dovrebbe contenere:
#   server=PC_SERVER
#   service=PR_SHARENAME
#   password="password"
#
# Esempio:
#   server=PAULS_PC
#   service=CJET_371
#   password=""

#
# File per i messaggi di debug, cambiare in /dev/null se si preferisce.
#
logfile=/tmp/smb-print.log
# logfile=/dev/null


#
# L'ultimo parametro del filtro è il file per la registrazione delle
# transazioni.
```

```
#
spool_dir=/var/spool/lpd/lp
config_file=$spool_dir/.config

# Le seguenti variabili dovrebbero essere assegnate nel file di
# configurazione:
#   server
#   service
#   password
#   user
eval `cat $config_file`

#
# Supporto per il debugging si può cambiare >> con > per risparmiare
# spazio.
#
echo "server $server, service $service" >> $logfile

(
# NOTA Si potrebbe desiderare di aggiungere la linea `echo translate'
# per avere la conversione automatica CR/LF quando si stampa.
    echo translate
    echo "print -"
    cat
) | /usr/bin/smbclient "\\\\$server\\$service" $password -U $user -N -P >> $logfile
```

La maggior parte delle distribuzioni Linux include `nenscript` per convertire documenti ASCII a Postscript. Lo script Perl che segue rende la vita più semplice agendo da semplice interfaccia per la stampa di Linux attraverso `smbprint`.

```
Usage: print [-a|c|p] <filename>
       -a prints <filename> as ASCII
       -c prints <filename> formatted as source code
       -p prints <filename> as Postscript
        If no switch is given, print attempts to
        guess the file type and print appropriately.
```

Usare `smbprint` per stampare file ASCII comporta a volte troncare le linee lunghe. Lo script che segue interrompe, se possibile, le linee lunghe su uno spazio (invece che in mezzo ad una parola).

La formattazione del codice sorgente è fatta con `nenscript`. Il file ASCII viene formattato su 2 colonne con un'intestazione (data, nome del file, ecc.) inoltre numera le linee. Usandolo come esempio, è possibile realizzare altri tipi di formattazione.

I documenti Postscript sono già correttamente formattati, quindi passano senza essere modificati.

```
#!/usr/bin/perl

# Script:   print
# Autore:   Brad Marshall, David Wood
#           Plugged In Communications
# Data:     960808
#
# Script per stampare su oreilly che è attualmente connessa a zimmerman
# Scopo:  Dati diversi tipi di file come argomenti, li elabora
```

```
# correttamente per mandarli in pipe ad uno script di stampa Samba.
#
# Tipi di file correntemente supportati:
#
# ASCII      - Verifica che le linee più lunghe di $line_length caratteri
#              si interrompano su spazio.
# Postscript - Intrapresa nessuna azione.
# Codice     - Formatta in Postscript (usando nenscript) per rendere al
#              al meglio (in orizzontale, font, ecc.)
#

# Lunghezza massima concessa per ciascuna linea di testo ASCII.
$line_length = 76;

# Nome e percorso dello script di stampa Samba
$print_prog = "/usr/bin/smbprint";

# Nome e percorso di nenscript (converte ASCII-->Postscript)
$nenscript = "/usr/bin/nenscript";

unless ( -f $print_prog ) {
    die "Non trovo: $print_prog!";
}
unless ( -f $nenscript ) {
    die "Non trovo: $nenscript!";
}

&ParseCmdLine(@ARGV);

# DBG
print "Il file e' di tipo: $filetype\n";

if ($filetype eq "ASCII") {
    &wrap($line_length);
} elsif ($filetype eq "code") {
    &codeformat;
} elsif ($filetype eq "ps") {
    &createarray;
} else {
    print "Spiacente...file di tipo sconosciuto.\n";
    exit 0;
}
# Pipe the array to smbprint
open(PRINTER, "|$print_prog") || die "Impossibile aprire $print_prog: $!\n";
foreach $line (@newlines) {
    print PRINTER $line;
}
# Spedisce un linefeed extra nel caso in cui il file abbia l'ultima linea
# incompleta.
print PRINTER "\n";
close(PRINTER);
print "Finito.\n";
exit 0;

# ----------------------------------------------------- #
```

```
#   Da questo punto in poi ci sono solo subroutine    #
# --------------------------------------------------- #

sub ParseCmdLine {
    # Analizza la linea di comand, cercando di riconoscere il tipo di
    # file.

        # Se esistono imposta $arg e $file agli argomenti.
    if ($#_ < 0) {
        &usage;
    }
    # DBG
#   foreach $element (@_) {
#       print "*$element* \n";
#   }

    $arg = shift(@_);
    if ($arg =~ /\-./) {
        $cmd = $arg;
    # DBG
#   print "\$cmd trovato.\n";

        $file = shift(@_);
    } else {
        $file = $arg;
    }

    # Definisce il tipo di file
    unless ($cmd) {
        # Nessun argomento

        if ($file =~ /\.ps$/) {
            $filetype = "ps";
        } elsif ($file =~ /\.java$|\.c$|\.h$|\.pl$|\.sh$|\.csh$|\.m4$|\.inc$|\.html$|\.htm$/) {
            $filetype = "code";
        } else {
            $filetype = "ASCII";
        }

        # Elabora $file in base al suo tipo e ritorna $filetype
    } else {
        # Il tipo che e' viene restituito in $arg
        if ($cmd =~ /^-p$/) {
            $filetype = "ps";
        } elsif ($cmd =~ /^-c$/) {
            $filetype = "code";
        } elsif ($cmd =~ /^-a$/) {
            $filetype = "ASCII"
        }
    }
}

sub usage {
    print "
Uso: print [-a|c|p] <nomefile>
```

```
        -a stampa <nomefile> come ASCII
        -c stampa <nomefile> formattato come codice sorgente
        -p stampa <nomefile> come Postscript
         Se non viene fornito alcun parametro, cerca di
         indovinare il tipo e stamparlo adeguatamente.\n
";
    exit(0);
}

sub wrap {
        # Crea un array di linee del file, dove ciascuna linea e' <
        # del numero di caratteri specificato, e termina solo su spazi.

        # Recupera il numero di caratteri a cui limitare la linea.
    $limit = pop(@_);

    # DBG
    #print "Entra subroutine wrap\n";
    #print "La lunghezza limite per la linea e' $limit\n";

    # Leggi il file, analizzalo e mettilo nell'array.
    open(FILE, "<$file") || die "Impossibile aprire: $file: $!\n";
    while(<FILE>) {
        $line = $_;

        # DBG
        #print "La linea e':\n$line\n";

        # Se la linea e' oltre il limite vai a capo.
        while ( length($line) > $limit ) {

            # DBG
            #print "Limita...";

            # Prendi i primi $limit +1 caratteri.
            $part = substr($line,0,$limit +1);

            # DBG
            #print "La linea parziale e':\n$part\n";

                        # verifica se l'ultimo carattere e' spazio
            $last_char = substr($part,-1, 1);
            if ( " " eq $last_char ) {
                # Se lo e' stampa il resto.

                # DBG
                #print "L'ultimo carattere era spazio\n";

                substr($line,0,$limit + 1) = "";
                substr($part,-1,1) = "";
                push(@newlines,"$part\n");
            } else {
                 # se non lo e', cerca l'ultimo spazio nella
                 # sottolinea e stampa fino a li'.
```

```
                # DBG
                #print "L'ultimo carattere non era spazio\n";

                 # RImuove i caratteri oltre $limit
                 substr($part,-1,1) = "";
                 # Rovescia la linea per rendere semplice da trovare
                 # l'ultimo carattere.
                 $revpart = reverse($part);
                 $index = index($revpart," ");
                 if ( $index > 0 ) {
                   substr($line,0,$limit-$index) = "";
                   push(@newlines,substr($part,0,$limit-$index)
                       . "\n");
                 } else {
                   # Non c'erano spazi cosi' stampa fino a $limit
                   substr($line,0,$limit) = "";
                   push(@newlines,substr($part,0,$limit)
                       . "\n");
                 }
            }
        }
        push(@newlines,$line);
    }
    close(FILE);
}

sub codeformat {
    # Chiama la funzione wrap e poi filtra il risultato attraverso
    # nenscript
    &wrap($line_length);

    # Manda il risultato attraverso nenscript per creare un file
    # Postscript che abbia un formato accettabile di stampa per
    # il codice sorgente (stile orizzontale, font Courier, numeri
    # di linea)
    # Per prima cosa stampa su un file temporaneo.
    $tmpfile = "/tmp/nenscript$$";
    open(FILE, "|$nenscript -2G -i$file -N -p$tmpfile -r") ||
        die "Non posso aprire nenscript: $!\n";
    foreach $line (@newlines) {
        print FILE $line;
    }
    close(FILE);

    # Legge il file temporaneo inun array per passarlo allo script di
    # stampa di Samba
    @newlines = ("");
    open(FILE, "<$tmpfile") || die "Impossibile aprire $file: $!\n";
    while(<FILE>) {
        push(@newlines,$_);
    }
    close(FILE);
    system("rm $tmpfile");
}
```

```
sub createarray {
    # Crea l'array per Postscript
    open(FILE, "<$file") || die "Impossibile aprire $file: $!\n";
    while(<FILE>) {
        push(@newlines,$_);
    }
    close(FILE);
}
```

A questo punto si può esaminare MagicFilter. Ringraziamenti ad Alberto Menegazzi (*flash.egon@iol.it*) per queste informazioni.

Alberto scrive:

1. Installare MagicFilter con il filtro per le stampanti in `/usr/bin/local` ma *non* si compili il file `/etc/printcap` con i suggerimenti dati dalla documentazione di MagicFilter.
2. Si scriva >`/etc/printcap` in questo modo (l'esempio si riferisce alla LaserJet 4L dell'autore):

```
lp|ljet4l:\
        :cm=HP LaserJet 4L:\
        :lp=/dev/null:\                        # or /dev/lp1
        :sd=/var/spool/lpd/ljet4l:\
        :af=/var/spool/lpd/ljet4l/acct:\
        :sh:mx#0:\
        :if=/usr/local/bin/main-filter:
```

   È necessario spiegare che 'lp=/dev/...' viene aperta e bloccata (locked) quindi è necessario utilizzare un dispositivo virtuale per ciascuna stampante remota che si debba usare. È possibile crearli ad esempio con: 'touch /dev/ljet41'

3. Si scriva un filtro `/usr/local/bin/main-filter` come quello suggerito, usando **ljet4l-filter** invece di bf/cat/. Ad esempio:

```
#! /bin/sh
logfile=/var/log/smb-print.log
spool_dir=/var/spool/lpd/ljet4l
(
  echo "print -"
  /usr/local/bin/ljet4l-filter
) | /usr/bin/smbclient "\\\\SHIR\\HPLJ4" -N -P >> $logfile
```

C'è anche un riferimento dal Print2Win mini-Howto relativamente il locking dei dispositivi e sul perché conviene creare stampanti virtuali.

```
Suggerimento da Rick Bressler:
A titolo di suggerimento, la configurazione che segue non è
una buona idea:

   :lp=/dev/null:\

in quanto lpr apre in modo 'esclusivo' il file specificato
tramite lp=.  È necessario per prevenire il tentativo di
diversi processi di stampare contemporaneamente sulla
stessa stampante.
```

```
Un effetto collaterale è che in tal caso, eng e colour non
possono stampare contemporaneamente, (di solito è più o
meno trasparente dal momento che stampano velocemente, e
dal momento che usano una coda, probabilmente non si nota),
tuttavia ogni altro processo che cerca di scrivere a
/dev/null sarà interrotto!

Non si tratta di un grosso problema in un sistema per utente
singolo.  L'autore ha un sistema con più di 50 stampanti.
Sarebbe un problema in questo caso.

La soluzione è quella di creare una stampante fittizia per
ciascuna di quelle fisiche, esempio con il comando:

touch /dev/eng
```

11 Eseguire copie di sicurezza (backup) di macchine Windows su un server Linux

Adam Neat (*adamneat@ipax.com.au*) ha gentilmente contribuito a fornire lo script che segue per fare backup di macchine Windows usando server Linux come supporto, tramite l'utility smbclient. Adam è solito fare backup di macchine Windows 3.x ed NT su un sistema Linux che dispone di un drive SCSI DAT.

Adam non è soddisfatto dello stile di codifica utilizzato, tuttavia funziona. Come l'autore ama commentare. Se funziona ed è stupido, allora non è stupido.

Un altro script di backup per Windows, è stato fornito da Dan Tager (*dtager@marsala.com*). Lo script di Dan è in grado di fare backup di macchine Unix via rsh e può essere facilmente modificato per usare ssh.

In questo script, la stringa 'agnea1' è il nome dell'utente sulla macchina Linux che fa il backup.

```
#!/bin/bash

clear
echo Inizializzazione ...
checkdate=`date | awk '{print $1}'`

if [ -f "~agnea1/backup-dir/backup-data" ]; then

        echo "ERRORE: non esiste il file di configurazione per oggi!"
        echo "FATALE!"
        exit 1
fi

if [ -d "~agnea1/backup-dir/temp" ]; then

        echo "ERRORE: Manca la directory temporanea!"
        echo
        echo "Tentativo di creazione"
        cd ~agnea1
        cd backup-dir
        mkdir temp
        echo "Creata directory - temp"
```

```
fi

if [ "$1" = "" ]; then

        echo "ERRORE: manca il nome della macchina (es: cdwriter)"
        exit 1
fi

if [ "$2" = "" ]; then

        echo "ERRORE: Manca la risorsa SMB (Lan Manager) (es: work)"
        exit 1
fi

if [ "$3" = "" ]; then

        echo "ERRORE: Manca indirizzo IP di $1 (es: 130.xxx.xxx.52)"
        exit 1
fi




###############################################################################
# Main Section
#
###############################################################################

cd ~agnea1/backup-dir/temp
rm -r ~agnea1/backup-dir/temp/*
cd ~agnea1/backup-dir/

case "$checkdate"
in
   Mon)
      echo "Backup di Lunedì"
      cat backup-data | /usr/local/samba/bin/smbclient \\\\$1\\$2 -I$3 -N
      echo "Completato"

      if [ -d "~agnea1/backup-dir/Monday" ]; then
        echo "Directory Monday non trovata ...  creazione"
        mkdir ~agnea1/backup-dir/Monday
      fi
      echo "Archiviazione ..."
      cd ~agnea1/backup-dir/temp
      tar -cf monday.tar *
      echo "fatto ..."
      rm ~agnea1/backup-dir/Monday/monday.tar
      mv monday.tar ~agnea1/backup-dir/Monday
      ;;

   Tue)
      echo "Backup di Martedì"
      cat backup-data | /usr/local/samba/bin/smbclient \\\\$1\\$2 -I$3 -N
      echo "Completato"
```

```
    if [ -d "~agnea1/backup-dir/Tuesday" ]; then
      echo "Directory Tuesday non trovata ... creazione"
      mkdir ~agnea1/backup-dir/Tuesday
    fi
    echo "Archiviazione ..."
    cd ~agnea1/backup-dir/temp
    tar -cf tuesday.tar *
    echo "fatto ..."
    rm ~agnea1/backup-dir/Tuesday/tuesday.tar
    mv tuesday.tar ~agnea1/backup-dir/Tuesday
    ;;

 Wed)
    echo "Backup di Mercoledì"
    cat backup-data | /usr/local/samba/bin/smbclient \\\\$1\\$2 -I$3 -N
    echo "Completato"

    if [ -d "~agnea1/backup-dir/Wednesday" ]; then
      echo "Directory Wednesday non trovata ... creazione"
      mkdir ~agnea1/backup-dir/Wednesday
    fi
    echo "Archiviazione ..."
    cd ~agnea1/backup-dir/temp
    tar -cf wednesday.tar *
    echo "fatto ..."
    rm ~agnea1/backup-dir/Wednesday/wednesday.tar
    mv wednesday.tar ~agnea1/backup-dir/Wednesday
    ;;

 Thu)
    echo "Backup di Giovedì"
    cat backup-data | /usr/local/samba/bin/smbclient \\\\$1\\$2 -I$3 -N
    echo "Completato"

    if [ -d "~agnea1/backup-dir/Thursday" ]; then
      echo "Directory Thrusday non trovata ... creazione"
      mkdir ~agnea1/backup-dir/Thursday
    fi
    echo "Archiviazione..."
    cd ~agnea1/backup-dir/temp
    tar -cf thursday.tar *
    echo "fatto ..."
    rm ~agnea1/backup-dir/Thursday/thursday.tar
    mv thursday.tar ~agnea1/backup-dir/Thursday
    ;;

 Fri)
    echo "Backup di Venerdì"
    cat backup-data | /usr/local/samba/bin/smbclient \\\\$1\\$2 -I$3 -N
    echo "Completato"

    if [ -d "~agnea1/backup-dir/Friday" ]; then
      echo "Directory Friday non trovata ... creazione"
      mkdir ~agnea1/backup-dir/Friday
```

```
      fi
      echo "Archiviazione ..."
      cd ~agnea1/backup-dir/temp
      tar -cf friday.tar *
      echo "fatto ..."
      rm ~agnea1/backup-dir/Friday/friday.tar
      mv friday.tar ~agnea1/backup-dir/Friday
      ;;

     *)
      echo "ERRORE FATALE: giorno non corretto"
      exit 1
      ;;
esac


###########
```

Lo script di Dan per eseguire il backup

```
#!/bin/bash

BACKDIR=3D/backup
WINCMD=3D/usr/bin/smbclient

function CopyWinHost(){

# usa tar e gzip per archiviare "condivisioni windows" in una
# directory locale usando l'utility smbclient della suite samba.
# argomento 1 è il nome del computer windows remoto
# argomento 2 è il nome della condivisione di cui fare backup

   echo $1,$2,$3
   REMOTE=3D$1
   SHARE=3D$2
   DEST=3D$3

 # crea un file tar gzip usando samba per copiare direttamente
 # dal pc con windows
 # 12345 è la password.  È necessaria anche non e' definita alcuna
 # sulla macchina remota

   $WINCMD \\\\$REMOTE\\$SHARE 12345 -Tc -|gzip > $DEST
   echo `date`": Completato backup da "$REMOTE" a "$DEST
   echo
}

function CopyUnixHost(){

# tar e gzip una directory usando rsh
# argomento 1 è il nome del computer remoto
# argomento 2 è il percorso completo della directory sorgente
#             remota
# argomento 3 è il nome del file tar-gzip locale. Il nome
#             del giorno più .tgz sono aggiunti in coda
#             all'argomento 3
```

```
   REMOTE=3D$1
   SRC=3D$2
   DEST=3D$3


   if rsh $REMOTE tar -cf - $SRC |gzip > $DEST; then
      echo `date`": Completato backup da "$REMOTE":"$SRC" a "$DEST
   else
     echo `date`": Errore durante il backup  da "$REMOTE":"$SRC" a "$DEST
   fi

}

# $1: win=backup macchina windows, unix=backup macchina unix
case $1 in
   win)
      # $2= nome della macchina remota windows,
      # $3= nome della condivisione remota,
      # $4= nome della directory locale
      CopyWinHost $2 $3 $4;;
   unix)
      # $2= host remoto,
      # $3= directory remota,
      # $4= nomde della directory locale
      CopyUnixHost $2 $3 $4;;
esac
```

12 Usare Samba attraverso reti instradate

Andrew Tridgell afferma che la consultazione di host SMB attraversando router è problematica. Ecco i suoi suggerimenti per consentire tale operazione:

Per la consultazione attravero subnet instradate è necessario comportarsi come segue. Ci sono anche altri metodi ma sono molto più complessi:

1) tutti i computer che si desiderano visibili dovrebbero usare un solo server WINS (Samba o NT possono)

2) il master browser per ciascuna sotto rete deve essere NT o Samba (Win9X non comunica correttamente informazioni attraverso cross-subnet)

3) si dovrebbe usare lo stesso nome di gruppo di lavoro su tutte le sottoreti. Non è strettamente necessario ma è il modo più semplice per garantire il successo. Se non fosse possibile allora è necessario organizzarsi in modo che le informazioni di browse possano propagarsi attraverso le sottoreti (non si propagano tramite WINS). Ci sono due meccanismi: i) ciascun master nota gli annunci di workgroup da altri master sullo stesso dominio di broadcast ii) ciascun non Win9X master contatta il DMB globale per il gruppo di lavoro (tipicamente il domain controller o un sistema Samba configurato come domain master) e scambia periodicamente la lista di browse completa.

Inoltre Rakes Bhrania indica che i router Cisco possono essere configurati per inoltrare il traffico SMB in modo da consentire il browsing. Il suo suggerimento è di configurare l'interfaccia del router relativamente host SMB con un comando simile a:

```
ip helper-address x.x.x.x
```

dove x.x.x.x è l'indirizzo IP del server SMB.

13 Ringraziamenti

Ringraziamente particolari ad Andrew Tridgell (*tridge@linuxcare.com*) per aver dato inizio e diretto il progetto Samba e per aver mantenuto questo documento corretto.

Brad Marshall (*bmarshall@plugged.net.au*) e Jason Parker (*jparker@plugged.net.au*) hanno contribuito con tempo, pazienza, script e ricerca.

Adam Neat (*adamneat@ipax.com.au*) e Dan Tager (*dtager@marsala.com*) hanno fornito script bash usati per il backup di macchine Windows verso server Linux.

Matthew Flint (*matthew@philtrum.demon.co.uk*) ha suggerito l'uso dell'opzione 'interfaces' nel file smb.conf

Alberto Menegazzi (*flash.egon@iol.it*) ha contribuito con il setup di MagicFilter per consentire ad una macchina Linux di condividere una stampante Windows

Rakesh Bharania (*rbharani@cisco.com*) ha suggerito la configurazione dei router CISCO.

Rich Gregory (*rtg2t@virginia.edu*) ed altri hanno suggerito che questo documento presentasse alcuni dettagli del pacchetto smbfs

Andrea Girotto (*andrea.girotto@era-net.it*) ha contribuito con alcuni utili suggerimenti.

Grazie anche a tutto i traduttori internazionali che hanno distribuito questo HOWTO a chi non parla inglese: Takeo Nakano (*nakano@apm.seikei.ac.jp*), Klaus-Dieter Schumacher (*Klaus-Dieter.Schumacher@fernuni-hagen.de*), Andrea Girotto (*andrea.girotto@era-net.it*), Mathieu Arnold (*arn_mat@club-internet.fr*), Stein Oddvar Rasmussen (*Stein@kongsberg-energi.no*), Nilo Menezes (*nmenezes@n3.com.br*) e molti altri di cui non ho le informazioni per il contatto.