SMB HOWTO

David Wood, *dwood@plugged.net.au*
Adaptation française par Mathieu Arnold `arn@multimania.com`, 28 Avril 1997. v1.0, 10 Août 1996

Voici le SMB-HOWTO, décrivant comment utiliser sous Linux le protocole SMB (« Session Message Block ») également appelé le protocole NetBIOS ou LanManager.

Contents

1 Introduction

Voici le SMB-HOWTO, décrivant comment utiliser sous Linux le protocole SMB (« Session Message Block ») également appelé le protocole NetBIOS ou LanManager.

Ce document est maintenu par David Wood (*dwood@plugged.net.au*). Veuillez envoyer vos ajouts, modifications ou corrections à cette adresse, pour qu'ils soient inclus dans la version suivante.

Le protocole SMB est utilisé par Microsoft Windows 3.11, NT et 95 pour partager des disques et des imprimantes. En utilisant les outils Samba d'Andrew Tridgell, les systèmes Unix (Linux inclus) peuvent également partager des disques et des imprimantes avec des hôtes Windows.

Vous pouvez faire quatre choses avec Samba :

1. Partager un disque Linux pour des machines Windows ;
2. Accéder à disque Windows depuis une machine Linux ;

3. Partager une imprimante Linux pour des machines Windows ;
4. Utiliser une imprimante Windows à partir d'un hôte Linux.

Tous ces points sont abordés dans ce document.

Mise en garde : Les procédures décrites et les scripts présentés fonctionnent pour l'auteur ou les personnes qui les ont écrits. Les informations données dans ce document peuvent ne pas fonctionner sur une configuration différente. Si vous rencontrez un problème, écrivez un mail à l'auteur avec vos suggestions pour l'amélioration de ce document, mais l'auteur ne garantit rien. Qu'attendiez-vous ? Après tout, l'auteur n'est qu'un consultant...

2 Pour plus d'informations

Cet HOWTO tente d'expliquer comment configurer de manière basique les services SMB de partage de fichier et d'imprimantes sous Linux. Samba est un ensemble de programme très complet, et très complexe également. Cela ne servirait à rien de recopier la documentation de Samba dans ce document.

Pour de plus amples informations, veuillez vous référer aux documents suivants :

- La documentation de Samba, disponible au sein de la distribution de Samba. Cette distribution est disponible à : *ftp://nimbus.anu.edu.au/pub/tridge/samba/* `ftp://nimbus.anu.edu.au/pub/tridge/samba/`
  NdT. : Vous pouvez récupérer les sources de Samba sur *ftp.ibp.fr*
- Le Printing-HOWTO.
- Le Print2Win Mini-HOWTO.

3 Installation

Les sources de la dernière version de Samba sont disponibles sur :

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

Cependant, si vous avez installé la distribution RedHat de Linux, vous pour pouvez l'installer comme paquetage. Certaines autres distributions proposent également un paquetage Samba (NdT : dont Slackware et Debian).

Les deux « daemons » suivant sont nécessaires pour Samba. Ils sont généralement installé dans le répertoire `/usr/sbin` et sont lancés soit au démarrage soit par inetd. Des scripts d'exemple sont présentés dans la section 4 (Lancer les daemons).

smbd

Le daemon SMB ;

nmbd

Propose le support du serveur de nom NetBIOS aux clients.

En général, les binaires Samba suivant sont installés dans `/usr/bin`, bien que leur emplacement soit variable.

smbclient

Un client SMB pour machines Unix ;

smbprint

Un script pour imprimer sur l'imprimante d'un hôte SMB ;

smbprint.sysv

Comme ci-dessus, mais pour des Unix SVR4 ;

smbstatus

Liste les connexions SMB présentes sur localhost ;

smbrun

Un script pour faciliter le lancement d'applications sur des hôtes SMB.

De plus, un script nommé « print » est inclus dans cet HOWTO. Il sert d'interface au script smbprint.

Le paquetage Samba est simple à installer. Il suffit de récupérer les sources depuis les sites mentionnés plus haut, et de lire les fichiers `README` fournis dans la distribution. Le fichier `docs/INSTALL.txt` fourni un ensemble d'instruction simple pour une configuration pas à pas.

Installez les daemons dans `/usr/sbin` et les autres programmes dans `/usr/bin`. Installez les pages *man* dans `/usr/local/man`.

Lors de la compilation du paquetage Samba, vous avez dû spécifier dans le Makefile l'emplacement du fichier de configuration, `smb.conf`. Il se trouve généralement dans `/etc`, mais vous pouvez le placer où bon vous semble. Dans la suite du document, nous considérerons que ce fichier de configuration est `/etc/smb.conf`, que le fichier de log est `/var/log/samba.log.%m` (valeur de « logfile » dans le Makefile) et que le répertoire des *locks* (verrous) est `/var/lock/samba` (« lock directory » dans le Makefile).

Installez le fichier de configuration, `smb.conf`. Allez dans le répertoire dans lequel vous avez compilé Samba. Consultez le fichier `README` dans le répertoire `examples/simple`. Copiez le fichier `smb.conf` de ce répertoire vers `/etc`. ATTENTION ! Si vous utilisez une distribution Linux et que vous avez déjà installé Samba, il doit déjà exister un fichier `smb.conf` dans `/etc`. Vous devriez sans doute commencer avec celui-là.

Si vous ne désirez pas placer votre fichier de configuration dans `/etc`, mettez le dans le répertoire que de votre choix, et faites un lien symbolique dans `/etc` :

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

4 Lancer les daemons

Les deux daemons SMB sont `/ust/sbin/smbd` et `/usr/sbin/nmb`.

Vous pouvez lancer les daemons Samba depuis l'inetd ou en tant que processus indépendants. Si vous êtes en train de configurer un serveur de fichier, ils devraient être lancés depuis l'inetd afin de pouvoir être redémarrez s'ils sont tués. Si vous comptez simplement utiliser les services SMB occasionnellement, vous pouvez démarrer les démons depuis un script `/etc/rc.d/init.d` ou même les lancer à la main lorsque vous en avez besoin.

Pour lancer les daemons depuis l'inetd, mettez les lignes suivantes dans le fichier de configuration d'inetd, `/etc/inetd.conf` :

```
# SAMBA NetBIOS services (for PC file and print sharing)
netbios-ssn stream tcp nowait root /usr/sbin/smbd smbd
netbios-ns dgram udp wait root /usr/sbin/nmbd nmbd
```

Et relancez le daemon inetd avec la commande :

```
killall -HUP inetd
```

Pour lancer les daemons depuis les scripts de démarrage, copier le script suivant sous le nom `/etc/rc.d/init.d/smb` et faites des liens depuis les fichiers spécifiés dans les commentaire sur ce script :

```
#!/bin/sh

#
# /etc/rc.d/init.d/smb - démarre et stoppe les services SMB.
#
# Les fichiers suivants doivent être des liens symboliques vers ce fichier~:
#    /etc/rc.d/rc1.d/K35smb  (Kille les services SMB à l'extinction)
#    /etc/rc.d/rc3.d/S91smb  (Démarre les services SMB en mode
#                             multi-utilisateur)
#    /etc/rc.d/rc6.d/K35smb  (Kille les services SMB au redémarrage)

# Charge la librairie de fonctions
. /etc/rc.d/init.d/functions

# Charge la configuration réseau
. /etc/sysconfig/network

# Vérifie que le réseau fonctionne
[ ${NETWORKING} = "no" ] && exit 0

# Traite les arguments
case "$1" in
   start)
      echo -n "Starting SMB services: "
      daemon smbd -D
      daemon nmbd -D
      echo
      touch /var/lock/subsys/smb
     ~;;
   stop)
      echo -n "Shutting down SMB services: "
      killproc smbd
      killproc nmbd
      rm -f /var/lock/subsys/smb
      echo ""
     ~;;
   *)
      echo "Usage: smb {start|stop}"
      exit 1
esac
```

5 Configuration générale (/etc/smb.conf)

La configuration de Samba sur une machine Linux (ou sur un autre Unix) est contrôlée par un seul fichier, `/etc/smb.conf`. Ce fichier indique quelle ressources système vous désirez partager avec le monde extérieur, et quelle restrictions vous voulez mettre dessus.

Puisque les chapitres suivants vont traiter du partage de fichier et d'imprimantes avec des machines Windows, le fichier `smb.conf` présenté dans ce chapitre est aussi simple que possible, en guise d'introduction.

Ne vous inquiétez pas pour les détails, en tout cas pas encore. Les chapitres suivants vont présenter les concepts fondamentaux.

Chaque partie du fichier commence par une entête, comme par exemple `[global]`, `[homes]`, `[printers]`, etc...

La section `[global]` définit quelques variables communes pour le partage de toutes les ressources.

La section `[homes]` permet à un utilisateur distant d'accéder à son répertoire d'accueil (et uniquement aux sien) sur la machine Linux locale. C'est-à-dire que si un utilisateur Windows essaye de se connecter à ce répertoire partagé depuis sa machine Windows, il sera connecté sur sa répertoire d'accueil personnel. Notez que pour cela, il doit posséder un compte sur l'hôte Linux.

Le fichier `smb.conf` suivant donné en exemple permet à des utilisateurs distants d'accéder à leur répertoire *home* sur la machine local et d'écrire dans un répertoire temporaire. Pour qu'un utilisateur Windows puisse voir ces partages, il faut que l'hôte Linux soit sur le réseau local. Ainsi l'utilisateur connecte un lecteur réseau depuis le gestionnaire de fichiers Windows ou l'Explorateur.

Veuillez noter que dans les chapitres suivants, des champs seront ajoutés à ce fichier pour permettre à plus de ressources d'être partagées.

```
; /etc/smb.conf
;
; Assurez vous de relancer le serveur après avoir fait des changement , dans
; ce fichier. Par exemple~:
; /etc/rc.d/init.d/smb stop
; /etc/rc.d/init.d/smb start

[global]
; décommentez cette ligne si vous désirez autoriser les invités (comptes
; "guest") à se connecter
; guest account = nobody
   log file = /var/log/samba-log.%m
   lock directory = /var/lock/samba
   share modes = yes

[homes]
   comment = Répertoire homes
   browseable = no
   read only = no
   create mode = 0750

[tmp]
   comment = Espace disque temporaire
   path = /tmp
   read only = no
   public = yes
```

6 Partager un lecteur Linux avec des machines Windows

Comme vous pouvez le constater avec le fichier `smb.conf` ci-dessus, partager des disques Linux avec des utilisateurs est simple. Néanmoins, comme pour tout avec Samba, vous pouvez tout contrôler finement.

Voici quelques exemples :

Pour partager un répertoire public, créez un clone de la section `[tmp]` ci-dessus en ajoutant le code suivant dans le `smb.conf` :

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

Pour que le répertoire ci-dessus soit en lecture pour tout le monde mais uniquement en écriture pour les personnes du groupe staff, modifiez l'entrée comme ci-dessous :

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

Pour d'autres trucs et astuces concernant les partages de répertoires, conférez vous à la documentation de Samba ou aux pages de manuel.

7 Partager un répertoire Windows pour des machines Linux

Un client SMB pour un hôte Unix est inclus dans la distribution de Samba. Il fournit une interface semblable au ftp, en ligne de commande. Vous pouvez utiliser cette utilitaire pour transférer des fichiers entre un « serveur » Windows et un client Linux.

Pour voir ce que partage une machine donnée, utilisez :

```
/usr/sbin/smbclient -L hôte
```

où « hôte » est le nom NetBIOS de la machine dont vous voulez voir les partages. Vous obtiendrez une liste des « services » en partage, c'est-à-dire le nom des répertoires ou des machines qu'il partage pour vous. A moins que le serveur SMB n'aient aucune protection, vous allez devoir saisir un mot de passe. Utilisez le mot de passe pour le compte d'invité ou pour votre compte personnel sur cette machine.

Par exemple :

```
smbclient -L zimmerman
```

Ce qui devrait donner quelque chose comme cela :

```
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 « browse list » (liste des machines) montre quels sont les autres serveurs SMB partageant des ressources sur le réseau.

Pour utiliser le client, lancez :

```
/usr/sbin/smbclient service <mot de passe>
```

ou « service » est une machine et un nom de partage. Par exemple, si vous essayez d'accéder à un répertoire en accès public sur une machine appelée zimmerman, le service sera nommé \\zimmerman\public. Néanmoins, à cause des restrictions de l'interpréteur de commandes (le *shell*), vous allez devoir redoubler les *backslashs*, pour obtenir la ligne suivante :

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

où « mon_mot_de_passe » est votre mot de passe écrit tel quel.

Vous allez obtenir la ligne de commande suivante :

```
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: \>
```

Tapez « h » pour obtenir de l'aide sur 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: \>
```

Si vous savez utiliser ftp, vous ne devriez pas avoir besoin de lire la page de manuel de smbclient.

8 Partager une imprimante Linux pour des stations Windows

Pour partager une imprimante Linux pour des stations Windows, vous devez être sûr que votre imprimante est configurée pour fonctionner sous Linux. Si vous savez imprimer depuis Linux, la mise en place d'un partage de l'imprimante par SMB est quasi-automatique.

Veuillez vous référer au Printing HOWTO pour la configuration de l'imprimante.

Puisque l'auteur utilise une imprimante connectée à une machine sous Windows NT, cette section ne doit pas être prise à la lettre, mais simplement comme une suggestion. Quiconque ayant des détails sur ce point est invité à les envoyer à *dwood@plugged.net.au* afin que ce chapitre puisse être complété.

Ajoutez la configuration d'impression à votre `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 = Toutes les imprimantes
   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
```

Assurez vous que le « path » (dans cet exemple, à l'intérieur de la section `[ljet]`) corresponde au répertoire de *spool* défini dans `/etc/printcap` !

NB : On constate quelques problèmes pour le partage d'imprimante sur des des serveurs Unix, utilisant Samba, pour des clients sous Windows NT. Un des problèmes est que NT ne voit pas l'imprimante partagée correctement. Pour résoudre ce problème, lisez les remarques dans le fichier `docs/WinNT.txt` de la distribution de Samba. Un autre problème existe avec les mots de passe. Voyez également dans ce fichier comment résoudre le problème.

9 Partager une imprimante Windows pour des clients Linux

Pour partager une imprimante sur une machine Windows, vous devez suivre les points suivants :

1. Vous devez avoir les entrées correspondantes à l'imprimante dans `/etc/printcap` et elles doivent correspondre à la structure locale des répertoires (pour le répertoire de *spool*, etc...).
2. Vous devez avoir le script `/usr/bin/smbprint`. Il est fournit avec les sources de Samba, mais pas avec toutes les distributions binaires. Une version légèrement modifiée de ce script est présentée plus loin.
3. Si vous voulez convertir des fichiers ASCII en Postscript, vous devez avoir `nenscript`, ou équivalent. `nenscript` est un convertisseur Postscript et se trouve généralement dans `/usr/bin`.
4. Vous voudrez peut-être simplifier l'impression à l'aide de Samba en utilisant une interface simple d'emploi. Un script simple, écrit en perl, pour gérer l'ASCII ou le PostScript est présenté ci-dessous.

L'entrée dans `/etc/printcap` est pour une imprimante HP 5 MP sur une machine utilisant Windows NT. Les entrées sont comme suit :

```
cm - commentaire
lp - nom du périphérique à ouvrir en écriture
sd - le répertoire de spool de l'imprimante (sur la machine locale)
af - le fichier d'accounting
mx - la taille maximum de fichier (zéro pour aucune limite)
if - le nom du filtre en entrée (un script)
```

Pour plus ample information, lisez le Printing HOWTO ou la page de manuel de printcap.

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

Assurez que le répertoire de spool et celui d'accounting existent et son accessibles en écriture, que le chemin correct vers le script smbprint (donné ci-dessous) est indiqué par la ligne « if » et que vous avez sélectionné le bon fichier de périphérique (le fichier spécial dans `/dev`).

Vient ensuite le script smbprint. Il est généralement mis dans le répertoire `/usr/bin` et a été créé par Andre Tridgell, le créateur de Samba pour autant que je sache. Il est fourni avec la distribution sous forme de code source de Samba, mais est absent de certaines distribution binaires. Je l'ai donc recopié ici.

Examinez le avec attention. Certains changement fait à ce script se sont avérées être utiles.

```
#!/bin/sh -x

# Ce script est un filtre d'entrée sur l'impression avec printcap sur une
# machine Linux. Il utilise le programme smbclient pour imprimer le fichier au
```

```
# serveur et service spécifié.
# Par exemple, vous pouvez avoir une entrée printcap comme celle-ci~:
#
# smb:lp=/dev/null:sd=/usr/spool/smb:sh:if=/usr/local/samba/smbprint
#
# qui créerai une imprimante Unix appelée "smb" qui imprimerait par
# l'intermédiaire de ce script. Vous devrez créer le répertoire de spool,
# /usr/spool/smb avec les permissions qui conviennent et le bon propriétaire,
# pour votre système.

# Mettez les valeurs pour le serveur et le service sur lequel vous voulez
# imprimer.  Dans cet exemple, j'utilise un PC sous Windows pour Workgroups
# nommé "laplan" ayant une imprimante appelée "printer" sans mot de passe.

#
# Script modifié par hamiltom@ecnz.co.nz (Michael Hamilton) afin que le
# serveur, le service et le mot de passe puissent être lus depuis un fichier
# /usr/var/spool/lpd/PRINTNAME/.config
#
# Pour que ceci puisse fonctionner, l'entrée du /etc/printcap doit inclure un
# fichier d'accounting (af=...)~:
#
#   cdcolour:\
#       :cm=CD IBM Colorjet au 6eme etage:\
#       :sd=/var/spool/lpd/cdcolour:\
#       :af=/var/spool/lpd/cdcolour/acct:\
#       :if=/usr/local/etc/smbprint:\
#       :mx=0:\
#       :lp=/dev/null:
#
# Le fichier /usr/var/spool/lpd/PRINTNAME/.config devrait contenir~:
#   server=SERVEUR_PC
#   service=NOM_IMPRIMANTE
#   password="mot_de_passe"
#
# Pas exemple~:
#   server=MON_BO_PC
#   service=CJET_371
#   password=""

#
# Fichier de log pour debuggage, changez le à /dev/null si vous le voulez
#
logfile=/tmp/smb-print.log
# logfile=/dev/null


#
# Le dernier paramètre du filtre est le nom du fichier d'accounting
#
spool_dir=/var/spool/lpd/lp
config_file=$spool_dir/.config

# Les variables suivantes devraient être lues depuis le fichier de
# configuration~:
```

```
#   server
#   service
#   password
#   user
eval `cat $config_file`

#
# Des informations de débogage, changez le >> en > si vous voulez économiser
# de la place.
#
echo "server $server, service $service" >> $logfile

(
# NOTE Vous voudrez peut être ajouter la ligne "echo translate" si vous voulez
# une conversion automatiques des CR/LF lors de l'impression
        echo translate
        echo "print -"
        cat
) | /usr/bin/smbclient "\\\\$server\\$service" $password -U $user -N -P >> $logfile
```

La plupart des distributions Linux sont fournies avec nenscript pour convertir des documents ASCII en Postscript. Le script perl qui suit simplifie la vie en fournissant une interface simple à smbprint pour l'impression sous Linux.

```
Usage: print [-a|c|p] <fichier>
       -a imprime <fichier> comme un fichier ASCII
       -c imprime <fichier> formatté en code source
       -p imprime <fichier> en tant que fichier Postscript
       Si aucun paramètre n'est donné, print tente de
       deviner le type de fichier et imprime en conséquence.
```

smbprint a tendance à tronquer les longues lignes lors de l'impression de fichiers ASCII. Ce script coupe les longues lignes sur les espaces (plutôt qu'au milieu d'un mot), si possible.

Le formatage en code source est réalisé par nenscript.Il prend en entrée un fichier ASCII et le formatte sur deux colonnes avec une entête (date, nom du fichier, etc...). Il numérote également les lignes. En prenant ce script comme exemple, on peut faire d'autres types de formatage.

Les documents sont déjà correctement formatés, donc ils passent directement à travers le filtre.

```
#!/usr/bin/perl

# Script:   print
# Auteurs:  Brad Marshall, David Wood
#           Plugged In Communications
# Date:     960808
#
# Script pour imprimer sur oreilly qui est pour l'instant connectée sur
# zimmerman.
# But:      Prendre différentes sortes de fichier en argument et les
#  traiter pour les injecter dans le script d'impression de Samba.
#
# Types de fichier supportés pour l'instant~:
#
```

```
# ASCII      - vérifie que les lignes plus longues que $line_length
#              caractères sont coupés sur un espace.
# Postscript - Aucune action.
# Code       - Formatte en Postscript (à l'aide de nenscript) pour un
#              affichage correct (orientation, fonte, etc...).
#

# Fixe la longueur maximale d'une ligne de texte ASCII
$line_length = 76;

# Le chemin d'accès vers le script d'impression de Samba
$print_prog = "/usr/bin/smbprint";

# Le chemin vers le programme nenscript (le convertisseur
# ASCII->Postscript)
$nenscript = "/usr/bin/nenscript";

unless ( -f $print_prog ) {
        die "Je ne peux pas trouver $print_prog!";
}
unless ( -f $nenscript ) {
        die "Je ne peux pas trouver $nenscript!";
}

&ParseCmdLine(@ARGV);

# Débug
print "filetype is $filetype\n";

if ($filetype eq "ASCII") {
        &wrap($line_length);
} elsif ($filetype eq "code") {
        &codeformat;
} elsif ($filetype eq "ps") {
        &createarray;
} else {
        print "Désolé, ce n'est pas un type de fichier que je connais";
        exit 0;
}
# Envoie le tableau à smbprint
open(PRINTER, "|$print_prog")
         || die "Je ne peux pas ouvrir $print_prog: $!\n";

foreach $line (@newlines) {
        print PRINTER $line;
}
# Envoie un retour à la ligne supplémentaire si jamais le fichier a sa
# dernière ligne incomplète
print PRINTER "\n";
close(PRINTER);
print "Achevé\n";
exit 0;

# -------------------------------------------------- #
#       Tout ce qui suit est un sous programme       #
```

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

sub ParseCmdLine {
        # Traite la ligne de commande, détermine le type de fichier

        # $arg et $file sont respectivement les arguments (s'ils
        # existent) et le nom de fichier
        if ($#_ < 0) {
                &usage;
        }
        # Débug
#       foreach $element (@_) {
#               print "*$element* \n";
#       }

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

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

        # Définition du type de fichier
        unless ($cmd) {
                # Aucun argument

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

                # Traite $file selon le type de fichier et retourne
                # le type de fichier ($filetype)
        } else {
                # Nous utilisons ltype de fichier décrit dans $arg
                if ($cmd =~ /^-p$/) {
                        $filetype = "ps";
                } elsif ($cmd =~ /^-c$/) {
                        $filetype = "code";
                } elsif ($cmd =~ /^-a$/) {
                        $filetype = "ASCII"
                }
        }
}

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

```
        -a imprime <fichier> comme un fichier ASCII
        -c imprime <fichier> formaté en code source
        -p imprime <fichier> en tant que fichier Postscript
        Si aucun paramètre n'est donné, print tente de
        deviner le type de fichier et imprime en conséquence.\n
";
        exit(0);
}

sub wrap {
        # Crée un table contenant les lignes du fichier, avec chaque
        # ligne ayant une longueur < au nombre de caractères
        # spécifiés, et coupée uniquement sur un espace.

        # Récupère la longueur maximum d'une ligne
        $limit = pop(@_);

        # Débug
        #print "Entrée dans la procédure wrap\n";
        #print "La longueur maximum d'une ligne est $limit\n";

        # Lit le fichier, le traite et le stocke dans le tableau
        open(FILE, "<$file") || die "Impossible d'ouvrir $file: $!\n";
        while(<FILE>) {
                $line = $_;

                # Débug
                #print "La ligne est~:\n$line\n";

                # Coupe la ligne si celle-ci dépasse la limite
                while ( length($line) > $limit ) {

                        # Débug
                        #print "Je coupe...";

                        # Prend les premiers $limit +1 caractères.
                        $part = substr($line,0,$limit +1);

                        # Débug
                        #print "La ligne partielle est~:\n$part\n";

                        # Vérifie si le dernier caractère est un
                        # espace
                        $last_char = substr($part,-1, 1);
                        if ( " " eq $last_char ) {
                            # Oui, on imprime le reste

                            # Débug
                            #print "Le dernier caractère était un espace\n";

                            substr($line,0,$limit + 1) = "";
                            substr($part,-1,1) = "";
                            push(@newlines,"$part\n");
                        } else {
                            # Non, on cherche le dernier espace de la
```

```
                                # ligne et on imprime jusqu'à lui

                                # Débug
                                #print "Le dernier caractère n'était pas un espace\n";

                                # Supprime le caractère après $limit
                                substr($part,-1,1) = "";
                                # Inverse la ligne pour trouver plus
                                # facilement l'espace
                                $revpart = reverse($part);
                                $index = index($revpart," ");
                                if ( $index > 0 ) {
                                  substr($line,0,$limit-$index) = "";
                                  push(@newlines,substr($part,0,$limit-$index)
                                      . "\n");
                                } else {
                                   # Aucun espace dans la ligne
                                   # Imprime jusqu'à $limit
                                   substr($line,0,$limit) = "";
                                   push(@newlines,substr($part,0,$limit)
                                       . "\n");
                                 }
                        }
                }
                push(@newlines,$line);
        }
        close(FILE);
}

sub codeformat {
        # Appelle la procédure wrap et filtre par nenscript
        &wrap($line_length);

        # Envoie le résultat à nenscript pour créer un fichier
        # Postscript qui respecte un format décent d'impression pour
        # du code source (orientation paysage, font Courier,
        # numérotation des lignes).
        # Imprime d'abord dans un fichier temporaire.
        $tmpfile = "/tmp/nenscript$$";
        open(FILE, "|$nenscript -2G -i$file -N -p$tmpfile -r") ||
                die "Je ne peux pas ouvrir nenscript~: $!\n";
        foreach $line (@newlines) {
                print FILE $line;
        }
        close(FILE);

        # Relis le fichier temporaire dans un tableau pour pouvoir
        # être passé au script smbprint de Samba.
        @newlines = ("");
        open(FILE, "<$tmpfile") || die "Je ne peux pas ouvrir $file~: $!\n";
        while(<FILE>) {
                push(@newlines,$_);
        }
        close(FILE);
        system("rm $tmpfile");
```

```
}

sub createarray {
        # Crée le tableau pour un fichier postscript
        open(FILE, "<$file") || die "Can't open $file: $!\n";
        while(<FILE>) {
                push(@newlines,$_);
        }
        close(FILE);
}
```

10 Copyright

11 Remerciements

Dès que vous m'enverrez des suggestions, je vous remercierai ici dans la prochaine version du document.