

Chapitre - 8

Le systèmes de fichiers réseaux SaMBa (Server a Message Block a) :

Introduction :

Historique : Andrew Tridgell

En 1991, Andrew Tridgell, thésard à l'Université de Canberra, Australie contraint d'utiliser PathWorks pour pouvoir tester eXcursion et accéder à des données sur des serveurs UNIX. Pour ce faire Tridgell écrit SaMBa pour Ultrix (reverse-engineering). Sa diffusion est non contrôlée et son développement prend fin en 1992. En 1993 il apprend l'existence de Linux et reprend le développement de SaMBa. Le projet SaMBa lui donna ce nom, en choisissant un nom voisin de SMB en interrogeant un dictionnaire UNIX, par la commande `grep "^s.*m.*b" /usr/dict/words`.

Implémentation de SaMBa :

SaMBa est en fait l'implémentation sur UNIX, donc Linux d'une émulation d'un serveur LanManager, développé par Microsoft vers 1987, à partir d'une création d'IBM (1985). Depuis la création, par Microsoft, des concepts de *groupe de travail* (Workgroup) et de contrôleur de domaine le logiciel SaMBa a toujours réussi à implémenter sous Linux les mêmes fonctionnalités. Avec Windows NT est apparu SaMBa version 2. Avec Windows 2000 est apparu SaMBa version 3. SaMBa gère très bien les ressources partagées, mais il a quelques lacunes concernant la base des utilisateurs ou l'utilisation des *Listes des Accès Contrôlés* (ACL, *Access List Control*). Actuellement, il est possible d'intégrer les ACL grâce au système de fichier XFS.

(*) Note : en plus d'améliorer le support pour les systèmes de fichiers réseau du style UNIX, Linux 2.6 apporte également plusieurs améliorations aux systèmes de fichiers réseaux du style Windows. Le système de fichiers partagé standard pour les serveurs Windows (de même que OS/2 les autres systèmes d'exploitation) a été le protocole SMB, *Server Message Block* et le noyau Linux possède un excellent support client du protocole SMB depuis de nombreuses révisions. Cependant Windows 2000 a standardisé un sur-ensemble du protocole SMB, connu sous le nom de CIFS, *Common Internet FileSystem*. L'intention de cette mise à jour majeure était de rationaliser et de redéfinir certains aspects de SMB qui à ce point était devenu un complet fouillis (le protocole lui-même était pauvrement défini et souvent étendu au point qu'il y avait des cas où même les versions Win95/98/ME étaient incompatibles avec les versions WinNT/win2k). CIFS a répondu à cette intention et ajouté le support UNICODE, amélioré le verrouillage des fichiers, les liens (hard linking, éliminé les derniers vestiges des dépendances à NetBIOS, et ajouté quelques autres fonctionnalités pour les utilisateurs de Windows. Comme les utilisateurs de Linux n'aiment pas être laissés dans l'ignorance très longtemps, Linux 2.6 inclut maintenant une réécriture complète du support natif des systèmes de fichiers CIFS. Linux 2.6 comprend maintenant aussi le support pour les extensions SMB-UNIX aux protocoles SMB et CIFS qui autorisent Linux à avoir accès aux types de fichiers non-Windows (tels que les noeuds périphérique et les liens symboliques) sur les serveurs SMB qui le supportent (tel que SaMBa).

Qu'est-ce que SaMBa ?

SaMBa est un ensemble de programmes qui permettent de connecter à un serveur fonctionnant sous un système d'exploitation GNU/Linux, avec des stations fonctionnant sous des systèmes divers : Windows 3.11, Windows 9x, Windows Nt, Windows 2000, OS/2, Mac....

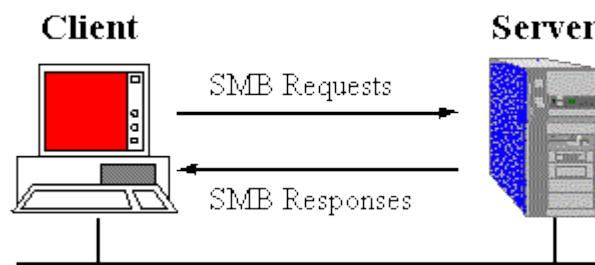
Le serveur GNU/Linux est en mesure de se conduire comme un serveur de fichiers capables d'offrir les services habituels sur un réseau :

- partage de fichiers et de répertoires,
- partage d'imprimantes,
- respect des comptes utilisateurs,
- gestion des permissions d'accès,
- exécution de scripts de connexion personnalisés.

Principe de fonctionnement :

Le protocole de communication sous-jacent qui permet cette communication Linux-Dos/Win s'appuie sur NetBIOS, *Network Basic Input/Output System* et s'appelle SMB, *Server Message Block*. Son fonctionnement est conforme au schéma client-serveur, c'est à dire que le client fait des demandes et le serveur envoie des réponses. La seule exception à ce mode d'échange est lorsque le client a demandé des *verrous opportunistes* (oplocks) et que le serveur doit rompre un `oplock` accordé parce qu'un autre client a demandé l'ouverture d'un fichier dans un mode incompatible avec l'oplock. Dans ce cas, le serveur envoie un message non sollicité au client pour lui signaler la rupture d'oplock. Les rôles de client et de serveur n'étant pas mutuellement exclusifs, une machine peut à la fois être client et serveur. Les clients se relient aux serveurs en utilisant NetBIOS au dessus TCP/IP. Une fois qu'ils ont établi une connexion, les clients peuvent alors envoyer des commandes SMB au serveur qui leur permettent d'accéder aux ressources partagées :

ex : communication SMB entre un client M\$Windoze et un serveur GNU/Linux :) (*)



(*) Note : ici on décrit la notion du client/serveur informatique et le principe de requête/réponse :

Image d'un restaurant : un client entre ; il est "pris en charge" par un serveur, le serveur fournit un service au client ; généralement, le serveur s'occupe de plusieurs clients.

Internet : plusieurs millions de serveurs serveurs DNS, de courrier, de nouvelles, de FTP, de Web...

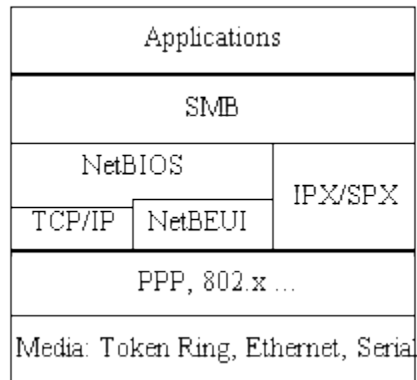
donc ; **machine cliente** ---> **machine serveur**

+ logiciel client <--- **+ logiciel serveur**

Rappels TCP/IP, NetBIOS :

Techniquement, SMB (mis au point par Microsoft, Intel et IBM) est une implémentation de LAN Manager sur TCP/IP permettant de partager différentes ressources telles que de l'espace disques, des imprimantes, des ports séries, etc...

NetBIOS, *Network Basic Input/Output System*, est une API réseau. Les programmes Windows utilisent cette API pour communiquer avec une autre machine. NetBIOS correspond aux couches transport et session de OSI :



SMB, un protocole complexe :

- le serveur offre ses ressources (système de fichiers, imprimantes ...) aux clients Windows qui s'y connecteront sous un compte créé par `root`, après une authentification par mot de passe.
- le travail est partagé par deux démons (daemons) : `smbd` pour le service serveur et `nmbd` pour le service de résolution des noms Netbios.
- du côté client, le protocole SMB fonctionne au-dessus de plusieurs protocoles.
- il nécessite NetBIOS au dessus de TCP/IP (par contre NetBEUI n'est pas utile)

Comparaison avec NFS (Network File System) :

Tableau comparatif :

SMB	NFS
de nombreuses commandes avec des sous-commandes	une douzaine de commandes au total
orienté connexion	sans état (stateless)
mécanismes sophistiqués (verrouillage, chaînage de commandes)	mécanismes de base
mal documenté, documentation suivant l'implémentation	protocole modélisé avant d'être implémenté

Les composants de SaMBa :

smbd processus principal gérant les requêtes de fichiers et d'imprimantes
nmbd processus gérant le service de nommage et d'authentification
testparm teste la syntaxe du fichier de configuration
swat outil d'administration graphique de SaMBa
smbclient client simple permettant de tester SaMBa
nmblookup utilitaire permettant de rechercher des noms de machines

Authentification :

Trois niveaux de sécurité possibles :

share pas de vérification de mot de passe
user vérification du mot de passe en local
server validation du mot de passe sur un autre serveur SMB, puis en local
domain authentification par un *contrôleur de domaine* (domain controller)

Le choix se fait en fonction des fonctionnalités désirées et de l'installation courante.

Contrôle d'accès :

Par utilisateur : les droits d'accès des utilisateurs sous UNIX s'appliquent. En particulier, possibilité de visualiser le contenu des répertoires des autres utilisateurs.

Par machine : contrôle au niveau de SaMBa, par deux directives (`host deny`, `host allow`).

Administration et configuration de SaMBa :

Quand est-il nécessaire ?

On installe un serveur SaMBa (une machine sur laquelle on installe un service de partage de fichiers et d'impressions) pour que chacun des ordinateurs sur le réseau puisse communiquer entre eux utilisant des systèmes d'exploitations différents de GNU/Linux. Le serveur SaMBa à exécuter doit donc en plus de servir des fichiers partagés et des connexions utilisateurs, s'intégrer dans un contexte de systèmes hétérogènes sans perturber le fonctionnement de l'ensemble du réseau :

ex : si vous décidez d'installer des serveurs et de stations GNU/Linux et gardez seulement quelques ordinateurs avec d'autres OS, sur votre *réseau local* (lan), vous pouvez migrer en douceur ces systèmes et applications pour ensuite les remplacer par des équivalents en Logiciel Libre.

Installation de SaMBa : smbd et nmbd

SaMBa fournit des services grâce à deux démons (daemons). Les partages de fichiers et d'imprimantes effectués par le démon `/usr/sbin/smbd` et un service de nommage et d'authentification avec le démon `/usr/sbin/nmbd`. L'installation du paquetage contenant SaMBa s'effectue sur la machine serveur (serveur SaMBa : serveur de fichiers et d'impressions) soit en mode graphique avec les CD-ROMs de la distribution, soit en ligne de commande.

Sous Mandrakelinux, vous n'avez rien à télécharger tapez simplement :

```
ex:urpmi samba samba-common samba-client samba-swat
```

Sous Redhat/Fedora, vous téléchargerez le paquet correspondant et l'installerez par :

```
ex:rpm -Uvh samba-xxxxx.rpm
```

Sous Debian, vous n'avez rien à télécharger tapez simplement :

```
ex:apt-get install samba samba-common smbclient smbfs swat samba-doc
```

Pour toutes les autres distributions, Vous passerez par les sources et téléchargerez la dernière version :

```
ex:$ cd /chemin_absolu_ou_est_le_fichier_xxx.tar.gz
$ tar -xzvf samba-xxx.tar.gz
$ cd samba-xxx/source/
$ ./configure --with-fhs --prefix=/usr --sysconfdir=/etc --localstatedir=/var
--enable-static --enable-shared --with-manpages-langs=fr --with-syslog --
with-smbwrapper --with-automount --with-libsmbclient --with-smbmount
$ make
$ su
<password>
# make install
# touch /etc/samba/smb.conf
```

A ce stade SaMBa est installé, vous devez maintenant ajuster les droits sur certains programmes et lancer les daemons nécessaires à son fonctionnement, par :

```
ex:# cd /usr/bin/
# chmod 4755 smbmount smbmount
# /usr/sbin/smbd -D
# /usr/sbin/nmbd -D
```

Fichier important de SaMBa :

/etc/samba/smb.conf ce fichier de configuration unique dans lequel se trouvent des blocs :

```
ex:vi /etc/samba/smb.conf ce fichier est le centre de la configuration de SaMBa.
```

```
testparm valide le fichier smb.conf dans le répertoire /etc/samba
```

```
testparm /etc/samba/smb.conf teste fichier smb.conf minimal ci-dessous
```

```
[global]
```

```
printing = bsd
```

```
printcap name = /etc/printcap
```

```
load printers = yes
```

```
guest account = pcguest
```

```
log file = /var/log/samba/log.%m
```

```
[tmp]
  comment = Temporary file space
  path = /tmp
  read only = yes
  public = yes
```

Si SaMBa est lancé avec ce fichier de configuration, les postes restant avec Windows du réseau local pourront voir dans le voisinage réseau qu'un ordinateur nommé (le nom de votre machine GNU/Linux) partage un répertoire /temp dans lequel vous avez le droit d'écrire.

(*) Note : chaque demande de connexion par SaMBa, d'une station au serveur GNU/Linux, laisse une trace stockée dans un fichier `log.%m`, situé dans le répertoire `var/log/samba/log.%m` (`%m` désigne le nom de la station). Toute connexion pourra donc être identifiée de manière précise puis examinée sur une ligne de ce fichier : nom d'utilisateur, nom de la machine, date, heure de début, heure de fin, services utilisés...

Le fichier de configuration (paramètres "avancés") :

`[global]` : la section globale introduit les paramètres qui s'appliquent au serveur dans sa totalité ou qui seront les paramètres par défaut pour certaines sections.

`netbios name` : vous pouvez spécifier le nom netbios de votre serveur SaMBa. Le nom netbios est visible dans le voisinage réseau de vos ordinateurs sous windows. Si vous ne spécifiez pas de nom netbios, le serveur SaMBa-Linux prendra comme nom netbios son nom réseau.

`invalid users` : liste des utilisateurs interdits d'accès à SaMBa. Par exemple "root" devrait être interdit.

`interfaces` : si votre serveur linux dispose de plusieurs cartes réseaux et que vous voulez restreindre son activité à un seul réseau.

`security` : choix du mode de sécurité que vous voulez utiliser. Si on utilise `security=user` ça oblige que chaque utilisateur ait un compte sur le serveur GNU/Linux. Si vous voulez lancer un SaMBa qui ne gère pas les utilisateurs et qui ne partage que des ressources identiques pour tout le monde, vous pourrez utiliser `security=share`.

`workgroup` : nom du groupe de travail dont votre serveur SaMBa-Linux doit faire partie.

`server string` : description de votre serveur SaMBa-Linux.

`socket options` : une liste d'options qui *ajuste* (tune) SaMBa et faire en sorte qu'il réagisse plus vite (par exemple).

`encrypt passwords` : faut il utiliser les mots de passes encrypté ? Il est important de savoir que tous les OS Windows (ou presque) utilisent un système différent !

`wins support` : votre serveur SaMBa-Linux fait-il aussi office de serveur wins ?

`os level` : "Niveau" de votre OS pour savoir qui sera élu maître du domaine, local, etc.

`domain master` : active SaMBa comme serveur maître du domaine

`local master` : active SaMBa comme serveur maître local

preferred master : SaMBA doit il être "préféré" à tout autre serveur si il y en a ?
domain logons : SaMBA doit-il gérer les contrôles de connexions pour le domaine ?
logon script : quel sera le script à lancer lors de l'ouverture de la session utilisateur ?
logon path : où se trouvent les fichier scripts de démarrage ?
logon home : où faut-il stocker les profils itinérants des utilisateurs ?
name resolve order : dans quel ordre on fait appel aux ressources pour trouver le nom d'une machine du réseau ?
dns proxy : le serveur SaMBA doit il faire aussi proxy DNS ?
preserve case : permet de garder la casse des noms de fichiers.
short preserve case : permet de garder la casse des noms de fichiers.
unix password sync : faut-il synchroniser les mots de passe UNIX et Windows ?
passwd program : quel programme lancer pour changer le mot de passe.
passwd chat : quel est le "protocole" de discussion pour changer le mot de passe.
max log size : taille maximum du fichier de log.

[netlogon] : section où l'on spécifie le répertoire /var/lib/samba/netlogon contenant les scripts.

[profiles] : section où l'on spécifie le répertoire /var/lib/samba/profiles contenant les profils utilisateurs des systèmes.

[homes] : section qui permet à chaque utilisateur d'accéder à son répertoire /home/xxx de la machine SaMBA-Linux depuis un PC sous Windows. Il s'agit d'un type de partage spécial.

Les variables de SaMBA :

Variables	Définitions
Client : %a %l %m %M	Architecture du client (Windows, Mac ...) Adresse IP du client Nom NetBios du client Nom DNS du client
Utilisateur : %g %H %u	Groupe primaire de l'utilisateur %u Répertoire home de l'utilisateur %u Nom de l'utilisateur UNIX courant
Partage : %P %S	Racine du partage actuel Nom du partage actuel
Serveur : %h %L %v	Nom DNS du serveur SaMBA Nom NetBios du serveur SaMBA Version de SaMBA
Diverse : %T	La date et l'heure courantes

Ajout d'utilisateurs : smbpasswd

Commencez par créer des utilisateurs spécifiques au réseau :

```
ex : groupadd smbusers
```

```
    useradd -g smbusers -s /bin/false smbadmin
```

```
    groupadd smbguests
```

```
    useradd -g smbguests -s /bin/false smbguest
```

A ce stade vous avez deux utilisateurs spécifiques à SaMBa : smbadmin (group smbusers) et smbguest (group smbguests). Inutile de leur affecter des mots de passe locaux, ils ne pourront jamais se connecter localement à la machine SaMBa. Par contre il leur faut un mot de passe réseau.

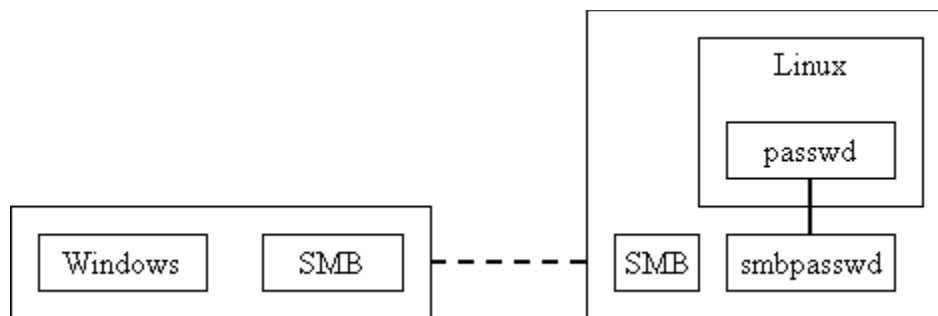
Il ne vous reste plus qu'à signaler au serveur SaMBa-Linux les utilisateurs, qui auront accès à SaMBa. Il peut s'agir des utilisateurs "normaux" de votre serveur GNU/Linux (ceux qui ont des /home) mais aussi ceux du groupe smbusers. Pour chaque utilisateur tapez dans un terminal :

```
ex : smbpasswd -a nom_utilisateur
```

 suivi de son mot de passe réseau (celui que vous taperez pour vous connecter au serveur SaMBa, depuis un client GNU/Linux ou Windows).

Configuration de mots de passe Linux pour SaMBa :

Un utilisateur Windows qui veut accéder à une ressource sur SaMBa-Linux va utiliser le protocole SMB pour y accéder. Cet utilisateur doit auparavant être authentifié par le serveur. Le serveur SaMBa-Linux, pour l'authentifier, vérifie sa présence dans le fichier /etc/samba/smbpasswd des mots de passe SaMBa. Il faut donc créer ce fichier des mots de passe smbpasswd pour le serveur SaMBa à partir du fichier passwd des mots de passe Linux :



Nous allons donc créer les utilisateurs SaMBa et leur affecter leur mot de passe. Pour cela, on fait dans un premier temps une copie du fichier /etc/shadow dans le répertoire /root sous le nom de passwd.tmp en utilisant la commande en console root ci-dessous :

```
ex : #cp /etc/shadow /root/shadow.tmp
```

On supprime alors tous les utilisateurs du fichier /root/passwd.tmp sauf nobody, toto, titi et vous. Puis, on crée le fichier des mots de passe SaMBa /etc/samba/smbpasswd :

```
ex : #awk mksmbpasswd /root/shadow.tmp > /etc/samba/smbpasswd
```

