

**Projet : Note de synthèse**

**Authentification unifiée sur  
un réseau hétérogène**



## **Introduction**

Le but de ce projet est de centraliser la gestion des utilisateurs d'un domaine sur un réseau composé de différents types de systèmes d'exploitations (Windows et Linux). Cette centralisation permet ainsi d'éviter une redondance des informations et de faciliter la mise à jour des informations concernant les utilisateurs.

J'ai mis en pratique ce projet au cours de mon stage de seconde année que j'ai effectué au pôle Universitaire Pierre-Jakez Hélias à Quimper

## Sommaire

1. Le pôle Universitaire, son enseignement, sa structure informatique. ....	4
2. Les problèmes et les besoins. ....	5
Figure 1 : Réseau avec plusieurs bases de données utilisateurs .....	5
3. La nature du projet .....	5
4. Les solutions .....	6
4.1. Un réseau homogène .....	6
4.2. La Solution Windows.....	7
Figure 2 : Réseau Windows avec une seule base de données utilisateurs.....	7
4.3. La Solution Unix .....	8
Figure 3 : Réseau Unix avec une seule base de données utilisateurs .....	8
5. Le choix final. ....	9
5.1. Comment choisir ? .....	9
5.1.1. Le coût .....	9
5.1.2. Les performances .....	9
Figure 4 : Temps de réponse SAMBA / Windows 2000.....	9
5.1.3. La fiabilité .....	10
5.1.4. L'évolutivité.....	10
5.1.5. La faisabilité.....	10
5.2. Le choix.....	11
5.3. La réalisation.....	11
5.3.1. Démarche de réalisation.....	11
5.3.2. Authentification des postes Linux par l'annuaire LDAP .....	12
5.3.2.1. Configuration Serveur .....	12
Figure 5 : Schéma LDAP .....	12
5.3.2.2. Configuration des clients.....	13
5.3.3. Configuration automatisée des postes Linux .....	14
5.3.4. Authentification des postes Windows par l'annuaire LDAP .....	16
5.3.4.1. Configuration Serveur .....	16
5.3.4.2. Configuration des Clients Windows .....	17
Figure 6 : Paramétrage d'un client Windows.....	17
5.3.5. Gestion automatisée des utilisateurs .....	18
6. En conclusion.....	20

## 1. Le pôle Universitaire, son enseignement, sa structure informatique.

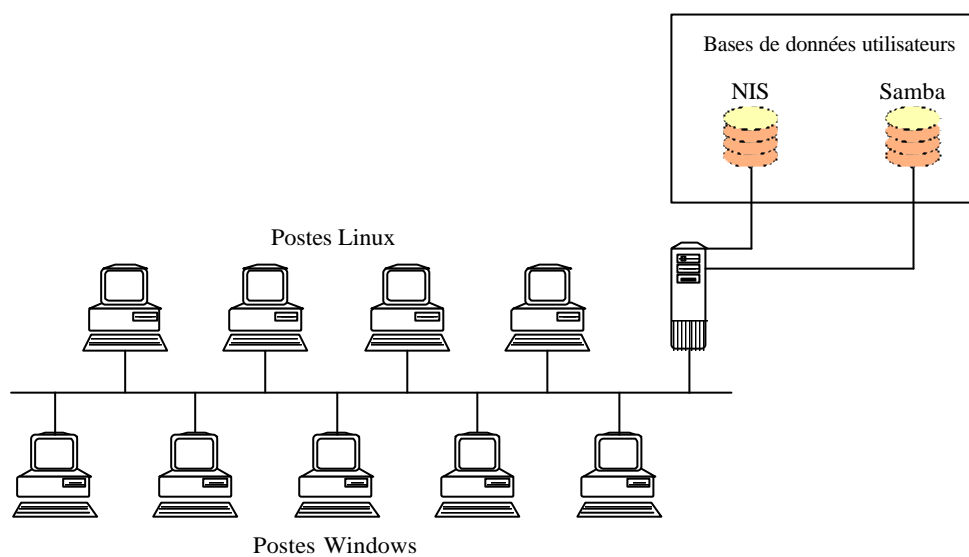
Le pôle Universitaire Pierre-Jakez Hélias est une structure commune de l'Université de Bretagne Occidentale regroupant sous un même toit toutes les composantes présentes à Quimper, IUT excepté. Cette mise en commun permet d'offrir un meilleur potentiel de fonctionnement et a accéléré l'éclosion d'une véritable vie universitaire.

L'enseignement qui y est dispensé couvre les études de Lettres, Langues, Histoire, Géographie, Histoire de l'Art, Droit, Administration et de Sciences Economiques.

## 2. Les problèmes et les besoins.

Le pôle dispose d'une centaine d'ordinateurs fonctionnant sous Windows (NT/XP) et Linux. Les utilisateurs, pour pouvoir accéder au poste de travail Linux ou Windows doivent s'authentifier sur le réseau et disposer de leur profil pour chaque interface graphique (NT/2000/XP/KDE/GNOME/WMAKER/...) et d'un accès à leur répertoire personnel sur le serveur de fichier.

La solution en place consiste à authentifier tous les utilisateurs par un serveur Samba pour les postes sous Windows et par un serveur NIS (Network Information System) pour les postes sous Linux. Mais cette solution pose un problème en ce qui concerne la gestion des utilisateurs. En effet Samba comme NIS doit disposer de sa propre base de données d'utilisateurs.



**Figure 1 : Réseau avec plusieurs bases de données utilisateurs**

Ce système pose alors un problème de redondance des informations concernant les utilisateurs ainsi qu'un risque que les informations ne soient plus synchronisées. La maintenance est doublée tout comme les causes possibles d'un dysfonctionnement.

## 3. La nature du projet

Afin de réduire les temps d'administration, de réduire le risque de pannes, de se donner des possibilités d'élargissement de services, il faut centraliser la gestion des utilisateurs sur le domaine du pôle.

## 4. Les solutions

### 4.1. Un réseau homogène

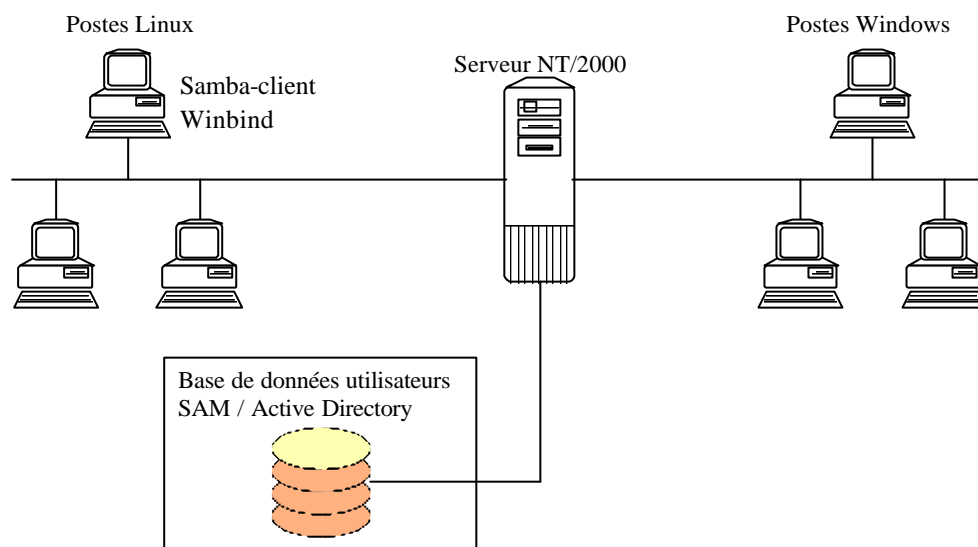
La première solution possible serait d'avoir un réseau homogène composé soit uniquement de postes Linux, soit uniquement de postes Windows, ce qui aurait pour conséquence de n'avoir plus qu'un seul annuaire d'utilisateurs. Mais cette solution n'est pas envisageable pour de multiples raisons :

- Des cours d'Unix sont donnés et l'émulation d'un terminal Unix ne permet pas une bonne sensibilisation à ce système.
- La multitude de logiciels sous l'environnement Unix incitent de nombreux professeurs à faire travailler leurs étudiants sous cette plate-forme
- Ms-Windows reste encore un « standard », on trouve encore de nombreux fichiers sur Internet au format Word, Excel ou PowerPoint. De plus, peu d'étudiants sont prêts à abandonner l'univers Windows au profit de l'Unix bien que des interfaces graphiques de type KDE sous Linux émulent parfaitement l'interface windows.

## 4.2. La Solution Windows

La seconde solution consiste à authentifier les postes Linux et Windows par un serveur sous Ms-Windows NT ou 2000. L'ensemble des informations concernant les utilisateurs serait centralisés sur SAM (Security Accounts Manager) pour Ms-Windows NT ou Active Directory (basé sur LDAP) pour Ms-Windows 2000.

En ce qui concerne les postes sous Windows, cette solution ne pose aucune difficulté puisqu'il s'agit de l'authentification standard de Microsoft. Pour ce qui est des postes Linux, chaque poste doit disposer de SAMBA-client qui permet la reconnaissance de la machine par le serveur Windows et de Winbind pour s'authentifier sur le serveur Windows. Ces deux logiciels sont gratuits et le code source est disponible (open-source).

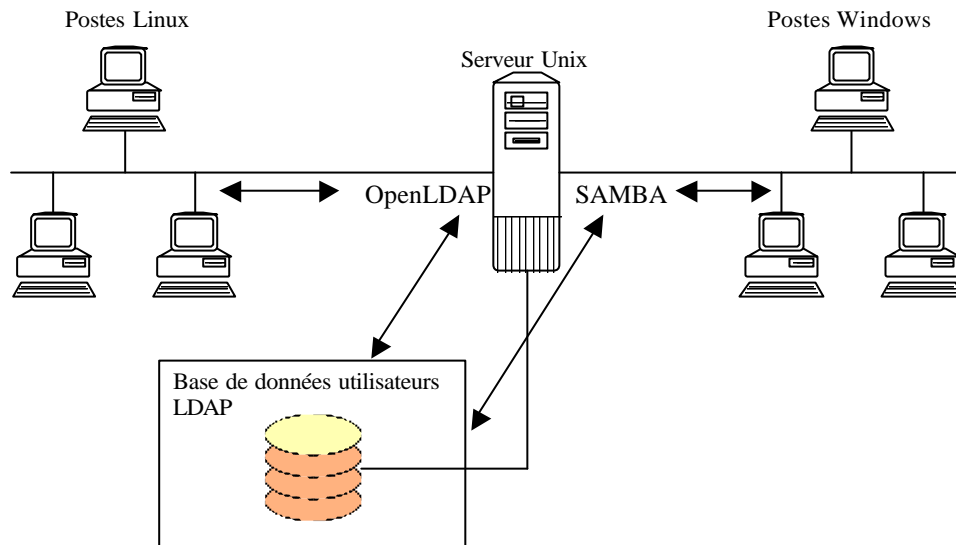


**Figure 2 : Réseau Windows avec une seule base de données utilisateurs**

### 4.3. La Solution Unix

La troisième solution consiste à authentifier les postes Linux et Windows par un serveur Unix. L'ensemble des informations concernant les utilisateurs serait centralisés sur LDAP (Lightweight Directory Access Protocol).

Les postes Linux seraient authentifiés par OpenLDAP qui ferait appel à l'annuaire LDAP. Quand aux postes Windows, ceux-ci seraient authentifiés par SAMBA qui ferait lui aussi appel à l'annuaire LDAP.



**Figure 3 : Réseau Unix avec une seule base de données utilisateurs**

## 5. Le choix final.

### 5.1. Comment choisir ?

Pour choisir une solution parmi les possibilités envisageables, il faut se baser sur différents critères tels que :

#### 5.1.1. Le coût

Le coût du matériel ne diffère pas en ce qui concerne les deux solutions. Ce qui diffère c'est l'achat des licences :

Pour un serveur Windows :

MS- Windows 2000 Advanced Server Licence Education : 700 €(Licence Standard : 5000 €)

Gestion des quotas disque (NT4) Licence Education : 1000 €(Licence Standard : 2500 €)

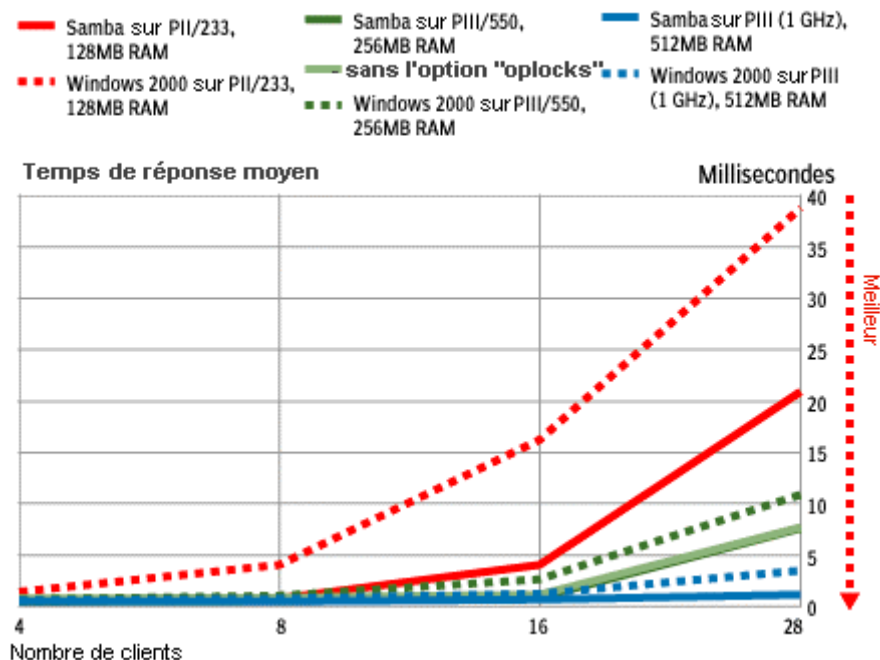
Gestion des quotas d'impression Licence Education : 1000 €(Licence Standard : 3000 €)

Des mises à jour payantes seront à effectuées tout les 3 à 4 ans.

Pour un serveur Linux :

Du fait de la multitudes d'outils gratuits de qualité professionnelle, il n'y a aucune licence à acheter pour cette solution.

#### 5.1.2. Les performances



**Figure 4 : Temps de réponse SAMBA / Windows 2000**

( Source : PC Magazine USA 11/01 )

Authentification unifiée sur un réseau hétérogène

Ce test illustre parfaitement la performance d'un serveur Samba face à un serveur Windows 2000. On constate que lors d'une montée en charge Windows 2000 perd de ses performances.

Linux est peu gourmand en ressources. On considère que pour délivrer une puissance égale à un serveur NT ou 2000, Linux nécessite entre 50 et 70% de ressources en moins. En d'autres termes, un serveur Linux Pentium III 1 Ghz avec 512 Mo de RAM fera l'affaire là où un serveur NT ou 2000 nécessiterait une machine deux, voire trois fois plus puissante avec deux, voire trois fois plus de RAM.

Dans le cas inverse, il suffit de remplacer NT ou 2000 par Linux et à configuration équivalente, on note un gain de vitesse de 50% lié à une diminution notable de l'occupation de la RAM et du temps CPU.

( Source : Comparaison Windows – Unix par John Kirch, spécialiste en réseaux certifié par Microsoft )

### 5.1.3. La fiabilité

Linux est stable. La durée de fonctionnement entre deux redémarrages ( « l'uptime » ) d'un serveur de fichiers et d'impressions NT est au mieux d'un mois. Si le serveur héberge en plus des applications telles qu'un serveur Web ou un serveur Exchange, il est prudent de l'arrêter et le redémarrer au minimum tous les 15 jours.

L'uptime d'un Linux correctement configuré se mesure en mois, parfois en années, sauf défaillance matérielle...

( Source : Quinn P. Coldiron - Linux Gazette n°29 )

### 5.1.4. L'évolutivité

En terme d'évolutivité, il est beaucoup plus intéressant de se pencher sur la solution Linux. En effet, des milliers de développeurs autour du monde travaillent tous les jours pour améliorer les fonctionnalités et corriger les éventuels bugs des programmes en open source tel que Samba.

### 5.1.5. La faisabilité

Même si l'administration d'un serveur 2000 n'est pas chose aisée, elle est tout de même plus facile que celle d'un serveur Linux. Administrer un serveur Linux ne s'improvise pas. Cela dit, une fois l'installation et la configuration initiale réalisées, il n'y a pas grand-chose à faire, et la supervision, comme pour 2000, peut facilement être réalisée à distance par Internet ou par modem.

## 5.2. Le choix

Ces éléments en mains, il faut maintenant choisir une des deux solutions : l'authentification par un serveur Windows ou par un serveur Linux.

Mon choix c'est porté vers la solution Linux car le seul «réel» défaut de cette solution est une plus grande difficulté de mise en place. Il faut dire que face à un coût minime (seulement en moyens humains), à des performances remarquables, à une fiabilité à toute épreuve et à une possibilité d'évolution sur du long terme, la solution Windows est beaucoup moins intéressante.

## 5.3. La réalisation

### 5.3.1. Démarche de réalisation

Pour simplifier la réalisation de ce projet, les travaux à effectuer ont été divisés en quatre étapes :

- L'authentification des postes Linux par l'annuaire LDAP :
  - Mise en place de l'annuaire LDAP
  - Configuration du processus d'authentification
- La configuration automatisée des postes Linux :
  - Réalisation de scripts bash et perl
- L'authentification des postes Windows par l'annuaire LDAP :
  - Installation de Samba sur le contrôleur de domaine
  - Configuration des clients Windows
- La gestion automatisée des utilisateurs :
  - Réalisation d'un script perl

### 5.3.2. Authentification des postes Linux par l'annuaire LDAP

Dans cette première partie, l'objectif est de centraliser l'authentification des utilisateurs de postes Linux sur un annuaire LDAP.

#### 5.3.2.1. Configuration Serveur

Du côté du serveur, il s'agit d'installer le logiciel openLDAP et de mettre en place le schéma de l'annuaire LDAP.

L'installation sur Debian se fait très aisément grâce à la commande `apt-get`.

L'arbre des utilisateurs choisi pour les utilisateurs LDAP est le suivant :

```
dc=fr,dc=univ-brest,dc=polepjh,ou=people
```

En ce qui concerne les groupes, ceux-ci seront mis dans :

```
dc=fr,dc=univ-brest,dc=polepjh,ou=groups
```

Ce schéma a été choisi par rapport au standard «PAYS – ENTREPRISE - STRUCTURE » tel que cela est conseillé pour l'implantation d'un schéma LDAP. On aura donc :

- `dc=fr` pour le pays : France
- `dc=univ-brest` pour « l'entreprise » : Université de Bretagne Occidentale
- `dc=polepjh` pour la structure : Pôle Pierre-Jakez Hélias

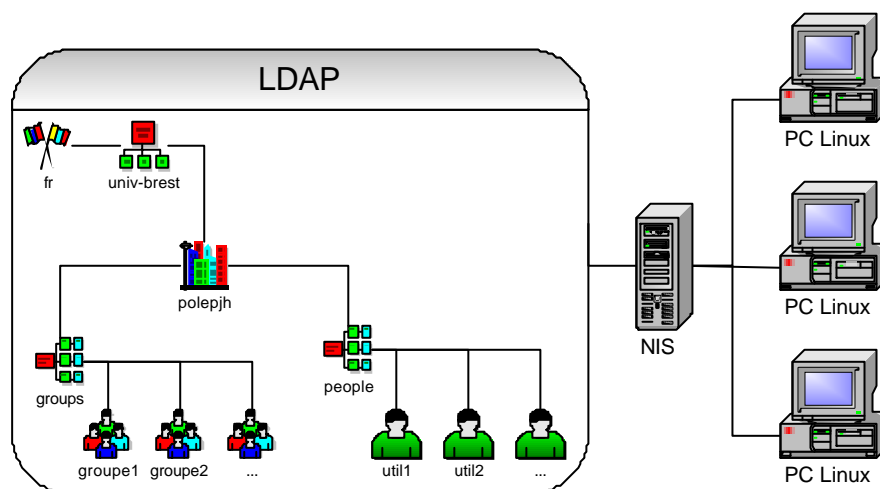


Figure 5 : Schéma LDAP

### 5.3.2.2. Configuration des clients

Dans la configuration des clients, il y a deux étapes fondamentales.

La première consiste à installer les packages indispensables pour faire le lien entre le système et le serveur openLDAP : ldap-utils, libnss-ldap et libpam-ldap.

Dans la seconde étape, il s'agit de configurer chaque client pour que la procédure d'authentification se fasse par rapport à openLDAP. Les fichiers /etc/nsswitch.conf, /etc/libnss-ldap.conf, /etc/pam\_ldap.conf, /etc/pam.d/login et /etc/pam.d/passwd servent à s'authentifier sur le système de commande. Pour chaque service supplémentaire nécessitant une authentification, un fichier est également à modifier. Par exemple : /etc/pam.d/kde pour KDE ou /etc/pam.d/ssh pour SSH.

Dans notre cas, seules les connections en ligne de commande et sur les environnements graphiques utilisent LDAP, les étudiants n'ayant pas besoins d'autres types de connections (SSH, Telnet ou FTP par exemple).

Exemples de fichiers de configuration :

```
<fichier /etc/nsswitch.conf>
# Ce fichier indique à linux les emplacement ou
# il doit chercher différentes informations
passwd: files ldap
group: files ldap
shadow: files ldap
hosts: dns ldap
networks: files
protocols: files
services: files
ethers: files
rpc: files
netgroup: files nis
</fichier>
```

```
<fichier /etc/pam_ldap.conf>
# Informations sur le serveur LDAP
host 192.168.1.1
base dc=univ-brest,dc=fr
ldap_version 3
pam_password crypt
</fichier>
```

```
<fichier /etc/pam.d/login>
# Ce fichier indique à linux quels modules il doit
# lancer lors d'une procédure d'authentification
auth sufficient /lib/security/pam_ldap.so
account sufficient /lib/security/pam_ldap.so
password sufficient /lib/security/pam_ldap.so
</fichier>
```

### 5.3.3. Configuration automatisée des postes Linux

Dans cette seconde partie, l'objectif est d'automatiser la configuration des postes en vue d'une authentification par un serveur LDAP. On utilisera ssh, un script bash et un script perl.

Un script perl exécuté sur le serveur va passer sur chaque poste par le biais de connexions ssh. Il permettra de lancer l'exécution du script bash sur chacun des clients. Le script bash va venir chercher des fichiers de configuration et les répertoires des utilisateurs dans des répertoires partagés sur le serveur.

Après un redémarrage (automatisé dans le script), les postes clients sont prêts à l'emploi.

Le script bash exécute les commandes suivantes :

- `apt-get install` : pour installer les différents modules que chaque poste requiert
- `mkdir` : pour créer les répertoires qui serviront à monter les partages NFS (Network File System)
- `mount` : pour monter le répertoire partagé où sont placés les différents fichiers de configuration des clients LDAP
- `cp` : pour faire des copies de sauvegarde des précédents fichiers de configuration et copier les nouveaux fichiers faisant appels au serveur LDAP.

Extrait du script perl de configuration automatisée :

Il s'agit de la procédure appelée pour chaque client exécutant les commandes permettant de paramétrer chacun des ordinateurs.

```
sub config {  
  
#  
# Procédure de configuration d'un poste client LDAP  
# Appel : config("nom du serveur","nom du client")  
#  
  
# Récupération des noms du serveur et du client :  
$user = "root";  
$host = $_[0];  
$serv = $_[1];  
  
print "\n\nConfiguration de $host...\n";  
  
# Test de la présence sur le réseau du poste client :  
$p = Net::Ping->new("icmp");  
$test = $p->ping($host);  
$p->close();  
  
if ($test == 1){  
  
# Copie de la clé ssh permettant la connexion :  
system("scp -r /configldap/temp/.ssh $user@$host:/root/");  
  
}
```

```

# Commandes exécutées sur le poste client :
$cmd = 'export PATH=/usr/bin:/bin:/sbin:/usr/sbin:/usr/local/sbin:
; DEBIAN_FRONTEND=noninteractive apt-get install nfs-common ldap-utils
libnss-ldap libpam-ldap --assume-yes && if [ ! -d "/configldap" ] ; then
mkdir /configldap ; fi && mount '.$serv.':/configldap/ /configldap && bash
/configldap/cfg_cli_ldap.bash && umount /configldap && echo Configuration
LDAP terminée';

sshopen2("$user@$host", *READER, *WRITER, "$cmd");

while (<READER>) {
    # Retour des informations du client sur le serveur
    # exécutant le script :
    chomp();
    print "$_\n";
};

close(READER);
close(WRITER);
}else{
    print "Problème de connexion réseau avec $host\n";
};
};

```

### 5.3.4. Authentification des postes Windows par l'annuaire LDAP

L'objectif de cette troisième partie est de centraliser les informations sur les utilisateurs de machines Windows avec un contrôleur de domaine principal Linux. Pour se faire, nous allons mettre en place un serveur Samba qui authentifiera les clients via l'annuaire LDAP.

#### 5.3.4.1. Configuration Serveur

Il faut tout d'abord compiler Samba avec le support LDAP car Samba n'est pas compilé avec LDAP par défaut puis inclure les références à Samba dans la structure de schémas de LDAP.

Compilation de Samba :

Téléchargez les sources de samba-2.2.7a dans /tmp sur samba.org et faites :

```
tar zxvf samba-2.2.7a.tar.gz
cp -a samba-2.2.7a/packaging/Debian/debian samba-2.2.7a/
```

Editez le fichier samba-2.2.7a/debian/rules et le modifier comme suit (en gras) :

```
<fichier rules>
61      [ -f source/Makefile ] || (cd source && ./configure \
62          --host=$(DEB_HOST_GNU_TYPE) \
63          --build=$(DEB_BUILD_GNU_TYPE) \
64          --with-fhs \
65          --prefix=/usr \
66          --sysconfdir=/etc \
67          --with-privatedir=/etc/samba \
68          --localstatedir=/var \
69          --with-netatalk \
70          --with-smbmount \
71          --with-syslog \
72          --with-sambabook \
73          --with-utmp \
74          --with-readline \
75          --with-libsmbclient \
76          --with-winbind \
77          --with-msdfs \
78          --quotas \
79          --with-ldapsam)

131      #install -m 0644 source/nsswitch/pam_winbind.so \
132      #$(DESTDIR)/lib/security/

142      #mv $(DESTDIR)/usr/bin/pam_smbpass.so $(DESTDIR)/lib/security/

182      #cp debian/samba.pamd $(DESTDIR)/etc/pam.d/samba
</fichier>
```

Dans /tmp/samba-2.2.7a/debian/ :

- Supprimez la ligne "lib/security/pam\_smbpass.so" du fichier libpam-smbpass.files (le fichier devrait être vide).

- Supprimez la ligne "/etc/pam.d/samba" du fichier samba-common.conf (le fichier devrait être vide).
- Supprimez la ligne "lib/security/pam\_winbind.so" du fichier winbind.files.

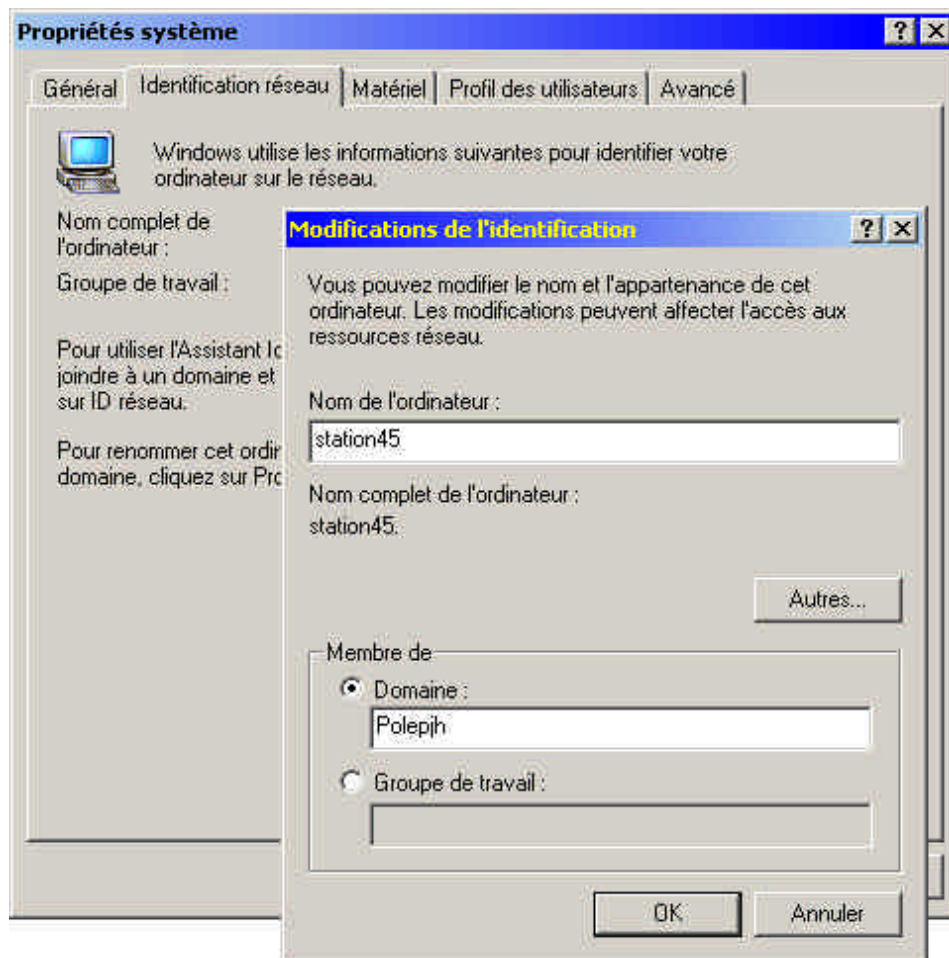
Puis faites : `dpkg-buildpackage`

La compilation terminée, allez dans /tmp/ et faites : `dpkg -i *.deb`

### 5.3.4.2. Configuration des Clients Windows

Pour WinNT, W2k et WinXP :

Dans l'onglet "Identification Réseau" des "propriétés système", cliquez sur le bouton "Propriétés". Choisissez l'option "Domaine" et indiquez le nom de votre domaine (celui indiqué dans /etc/samba/smb.conf ). Validez et entrez le compte ainsi que le mot de passe de l'administrateur (ajouté automatiquement avec le script de gestion automatique des utilisateurs -> howto suivant). Windows vous répond "Bienvenue dans le domaine", vous devez ensuite redémarrer.



**Figure 6 : Paramétrage d'un client Windows**

Authentification unifiée sur un réseau hétérogène

### 5.3.5. Gestion automatisée des utilisateurs

Arrivé à ce stade, tout est opérationnel mais la gestion des utilisateurs risque encore de rester fastidieuse. Afin de la simplifier, il est préférable de réaliser un programme automatisant ces tâches.

Cette quatrième et dernière étape consiste donc à réaliser un script Perl pour automatiser la gestion des utilisateurs.

Ce script permet de créer ou de supprimer des utilisateurs, des groupes et d'afficher toutes les informations nécessaires à l'administration des utilisateurs.

Extrait du script perl :

```
sub add_user {
#####
# Pour ajouter un utilisateur
#####
# add_user ( $_[0], $_[1], $_[2], $_[3], $_[4], $_[5], $_[6], $_[7], $_[8],
# $_[9], $_[10])
# login, num_etudiant, nom, prenom, num_groupe, repertoire_groupe, shell,
uid, nom_groupe, iup, mot_de_passe

    if ($_[10] eq ""){
        # Génération du mot de passe si pas de mot de passe
        # (4 derniers chiffres du numéro d'étudiant)
        $password = substr($_[1],-4,4);
    }else{
        $password = $_[10];
    };

    # Cryptage du mot de passe (Linux)
    $pass_u = crypt($password,substr($_[1],-4,2));

    # Cryptage du mot de passe (Windows)
    ($lmpassword,$ntpassword) = ntlmgen $password;

    # Ajout à la base LDAP
    $result = $ldap->add ( 'uid=.'.$_[0].',ou=People,.'.$base,
        attr => ['objectclass' => ['top',
            'person',
            'posixAccount',
            'sambaAccount',
            'organizationalPerson',
            'inetOrgPerson'],
            'cn' => $_[2].".".$_[3],
            'sn' => $_[0],
            'uid' => $_[0],
            'uidNumber' => $_[7],
            'gidNumber' => $_[4],
            'homeDirectory' => $_[5].$_[0]."/",
            'loginShell' => $_[6],
```

```

        'employeeNumber' => $_[1],
        'userPassword'   => "{CRYPT}" . $pass_u,
        'pwdLastSet'     => 0,
        'logonTime'      => 0,
        'logoffTime'     => 2147483647,
        'pwdCanChange'   => 0,
        'pwdMustChange' => 2147483647,
        'acctFlags'      => '[U          ]',
        'rid'             => $_[7],
        'primaryGroupID' => 221,
        'lmPassword'     => $lmpassword,
        'ntPassword'     => $ntpassword
    ]

    );

$erreur = 0 ;

# $result->code = 1 si l'ajout n'a pas réussi
if ($result->code){
    $erreur = 1 ;
};

# En cas d'erreur, on ne crée pas son répertoire
if ($erreur == 0){
    print "Création de ".$_[0]." (id:".$_[7].")\n";
    system ("mkdir ".$repcomptes.$_[8]."/".$_[0]);
    system ("mkdir ".$repcomptes.$_[8]."/".$_[0]."/.profile");
    system ("chown -R ".$_[0].":".$_[4]." ".$repcomptes.$_[8]."/".$_[0]);

    # Etablissement des quotas
    if ($_[9] == 1){
        system ("edquota -p quotaiup ".$_[0]);
    }else{
        system ("edquota -p quotapole ".$_[0]);
    };
    print{passwd}
    $_[0].":".$pass_u.":".$_[7].":".$_[4].":,:":.$_[5].$_[0].":".$_[6]."\n";
    print{liste} $_[8].":".$_[2].":".$_[3].":".$_[1].":".$_[0]."\n";
    $debuid += 1;
}else{
    print "Erreur : Impossible de créer ".$_[0]."\n";
    print "Message: ".$result->error."\n";
};
};

```

## 6. En conclusion

La mise en place du service LDAP a déjà permis de n'avoir qu'une seule source d'informations pour les utilisateurs. Mais cette source unique de stockage des informations permet de larges perspectives d'évolution. On peut ainsi, grâce à cette nouvelle façon de centraliser la source d'informations sur les utilisateurs, imaginer de mettre en place très facilement une multitude de services y faisant appel, comme du FTP, du MAIL...