OpenLDAP, un outil d'administration réseau.

(Implémentation d'openLDAP à l'INRA de Rennes)

Gilles LASSALLE
Unité Mixte de Recherche d'Amélioration des Plantes et Biotechnologies Végétales
Domaine de la motte BP 35327,
35653 Le Rheu Cedex,
Gilles.lassalle@rennes.inra.fr le 8 octobre 2003.

Résumé

L'avènement de Linux et des logiciels libres a permis le déploiement et la mise en place de réseaux sous ces architectures. L'utilisation de LDAP et en particulier d'OpenLDAP nous a permis de nous doter d'un outil d'administration réseau très performant. Depuis mai 2002, nous exploitons ce logiciel dans un environnement hétérogène. Associés à d'autres logiciels réseaux tels que Samba, Sympa, Apache, il est devenu le centre névralgique de notre architecture.
OpenLDAP nous offre des fonctionnalités variées dont une gestion de l'authentification en réseau dans un environnement hétérogène qui remplace avantageusement Nis. L'exploitation de Samba, avec LDAP et les ACLs sous XFS nous ont permis de constituer ainsi un PDC très performant.
L'annuaire déployé sous OpenLDAP, nous permet de gérer actuellement les comptes utilisateurs pour Linux et pour Windows via le PDC Samba. La principale difficulté a été de déterminer l'organisation de l'arborescence ainsi que de choisir les associations d'objets LDAP permettant de gérer les comptes utilisateurs, machines et groupes. L'association OpenLDAP-Sympa-Postfix, nous permet aussi d'utiliser des mailing-lists dynamiques, se calquant sur la composition de nos équipes et groupes de travail. Enfin, avec Apache et Php, nous gérons la sécurité et les accès à notre intranet et aux divers développements réalisés dans cette architecture.
Il s'agit là d'un formidable outil d'administration à tout point de vue, pour la gestion centralisée des comptes aussi bien que pour le suivi du parc informatique et la gestion de la sécurité.

Mots clefs

OpenLDAP, administration réseau, authentification, Samba, sécurité.

1 Introduction

L'administration des comptes utilisateurs en réseau, sur plusieurs serveurs Linux simultanément peut être relativement compliquée, surtout s'il s'agit aussi de monter un Contrôleur de Domaine NT avec Samba et des contrôleurs secondaires.

Notre objectif était de mettre en place un outil d'administration, centralisé, unique, nous permettant de gérer à la fois les comptes Unix-Linux, l'authentification Windows, les accès à l'intranet et des mailing-lists dynamiques.

LDAP permet de gérer toutes les informations nécessaires à la mise en place d'une telle structure, avec une très grande facilité, pour peu que l'on ait quelques notions sur les annuaires. Son fonctionnement en réseau et son déploiement entre plusieurs serveurs, sont relativement aisés et rapides à mettre en oeuvre. Sa compatibilité avec de nombreux logiciels en font dorénavant un outil incontournable pour un administrateur réseau.

L'objectif de cet article n'est pas de donner un cours théorique sur LDAP mais de présenter une utilisation concrète des fonctionnalités de LDAP au travers du logiciel OpenLDAP, associé à plusieurs autres logiciels et en n'utilisant que des standards LDAP.

Nous avons monté au sein de l'Unité Mixte de Recherche INRA-ENSAR d'Amélioration des Plantes et de Biotechnologies Végétales, une architecture réseau dont OpenLDAP est le centre névralgique. Ce déploiement tourne en production depuis mai 2002 et met aussi en œuvre Samba, Sympa et Apache.

LDAP étant aussi et avant tout un annuaire, nous avons décider de nous baser sur la structure géographique de notre institut pour le déployer.

1.1 Présentation et Organisation de l'INRA.

Créé en 1946, l'Institut National pour la Recherche Agronomique est un établissement public à caractère scientifique et technologique, placé sous la double tutelle des ministères de la Recherche et de l'Agriculture.

L'INRA possède 17 départements de recherche, 21 centres régionaux répartis en près de 200 sites dans toute la France, 257 unités de recherche, 80 unités expérimentales, 131 unités d'appui et de service.

L'Unité Mixte de Recherche (UMR) en Amélioration des Plantes et Biotechnologies Végétales (APBV) de Rennes-Le Rheu (avec environ 100 agents) concentre ses activités de recherches et de sélection sur diverses espèces d'intérêt agronomique : blé, colza, crucifères légumières, féverole et pois protéagineux.

Un parc informatique d'une centaine de PC est à disposition du personnel au sein de l'UMR et des différentes installations.

Nous avons installé un ensemble de trois serveurs Linux Debian afin d'organiser le travail en réseau et d'offrir des solutions logiciels diverses tel que serveur de fichiers, intranet, SGBD, Groupware etc.…

1.2 Le choix d'OpenLDAP.

Sachant que nous voulions mettre en place LDAP et que notre politique informatique est de nous ouvrir à l'Open source, de tous les serveurs LDAP présents sur le marché, seul OpenLDAP était sous licence GPL. De plus, tous les schémas et objets présents nous ont permis de déployer un annuaire fonctionnel répondant complètement à nos besoins, sans développer quoique ce soit.

2 Déploiement d'OpenLDAP

2.1 Logiciels

2.1.1 Logiciels serveurs

Le serveur LDAP que nous utilisons est la version 2.0.25 de OpenLDAP.
La version de Samba est la 2.2.5. Actuellement, nous utilisons encore cette association, car depuis 18 mois elle nous a prouvé sa robustesse et sa fiabilité. Les essais avec des versions antérieures dont la 2.2.4 avaient échoué, car nous avions des bugs réguliers lors de l'authentification.

L'association LDAP-Sympa permet de gérer de manière très efficace des mailing-listes faisant référence à des entrées LDAP. La version utilisée de Sympa est la 3.4.4.1.

2.1.2 Navigateurs LDAP

Les logiciels clients utilisés sont essentiellement GQ sous Linux et LDAP-Browser sous Windows. Les fonctionnalités de GQ permettent de manier très facilement les données dans l'annuaire et de créer de nouvelles entrées à la volée. La version que j'utilise actuellement est la 0.6.0, package standard de la distribution Debian-Woody. En compilant soit-même cet outil, nous pouvons activer des fonctionnalités de drag & drop, parfois pratiques mais très dangereuse en cas d'un « relâché » de souris intempestif.

LDAP-Browser-Editor est très pratique pour visualiser le contenu de l'annuaire à partir d'un poste Windows, par contre des bugs d'affichage peuvent se produire et l'arborescence n'apparaît pas entièrement développée.

2.2 Matériels et architecture

2.2.1 Serveurs

Dans un premier temps nous avons utilisé une batterie de test de 4 serveurs assemblés dont le principal était un bi-Processeur PIII. La mise en place de LDAP est relativement peu coûteuse en ressource machine et tourne aussi sur de simples PC.

Actuellement nous avons 4 serveurs Bi-processeurs Xeon Compaq, nous permettant d'utiliser au maximum la tolérance de pannes et d'assurer, ainsi, la sécurité matériel imposée par la mise en place d'une démarche Assurance Qualité Recherche. La compatibilité matérielle avec ces serveurs est complète à partir du noyau Linux 2.4.20.

2.2.2 Organisation des serveurs

Nous avons actuellement une batterie de 4 serveurs Linux Debian Woody. Un des serveurs possède la réplique-Master alors que les trois autres sont esclaves. Les quatre machines ont des rôles dédies répartis comme suit :

- un PDC Samba,
- un serveur intranet et de messagerie,
- un serveur de base de données
- un serveur de données et d'application bio-informatique.

Tous les serveurs utilisent l'authentification via LDAP et permettent des accès soit sous X11, html+php, et Samba. La figure ci-dessous nous montre l'utilisation de LDAP avant la mise en place des répliques.

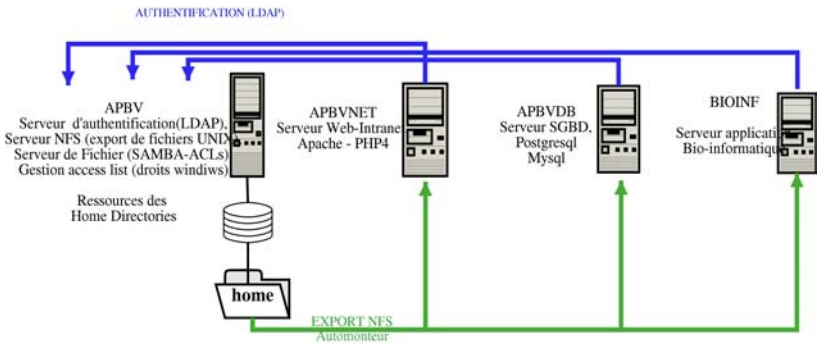

Figure 1 : architecture réseau générale.

2.3 installation

2.3.1 Installation

Nous avons réalisé l'installation à partir des sources OpenLDAP compilé avec des options standard. Pour la compilation, il faut installer auparavant toutes les bibliothèques nécessaires, avec entre-autres la *libldap2*, *libldap2-dev*, la *libdb3* et ses composantes. Cette dernière permet d'utiliser le backend LDBM.

OpenLDAP peut fonctionner avec plusieurs moteurs (backend) de base de données dont le type plus courant est LDBM. L'université de Berkeley développe une base de données de ce type, Berkeley DB, contenu dans les packages *libdb** de Debian. LDBM est le moteur utilisé en standard par OpenLDAP.

Ce dernier est aussi prévu pour fonctionner avec des SGBD utilisant SQL mais les performances de l'annuaire chute de façon très importante du fait, entre-autre de la traduction des requêtes LDAP en SQL. Cette étape est décrite dans les documents et tutoriaux consacrés à OpenLDAP.

2.3.2 Configuration et schémas

Le fichier *slapd.conf* permet de configurer le serveur, de définir les schémas utilisés et le mot de passe de l'administrateur du serveur LDAP.

Les schémas LDAP sont une collection d'objets répondant aux normes de l'OMG (Object Management Group) et dont chaque composant est attribué d'un OID (Object ID). Ces différents schémas permettent ainsi d'implémenter l'utilisation des comptes Samba et Postfix. La déclaration de leur utilisation se fait dans le fichier *slapd.conf*.

Une fois l'installation d'OpenLDAP réalisée et les schémas choisis, il faut déterminer l'architecture de l'arbre et des données à insérer. Ce paramétrage n'est pas définitif et il est possible à posteriori de procéder à une extension des schémas et donc d'en utiliser de nouveaux.

Extrait du fichier *slapd.conf :*

```
        # Schema and objectClass definitions
include         /usr/local/ldap/etc/openldap/schema/core.schema
include         /usr/local/ldap/etc/openldap/schema/cosine.schema
include         /usr/local/ldap/etc/openldap/schema/nis.schema
include         /usr/local/ldap/etc/openldap/schema/misc.schema
include         /usr/local/ldap/etc/openldap/schema/inetorgperson.schema
include         /usr/local/ldap/etc/openldap/schema/openldap.schema
include         /usr/local/ldap/etc/openldap/schema/samba.schema
include         /usr/local/ldap/etc/openldap/schema/postfix.schema
```

Ces schémas sont fournis avec les sources des logiciels des logiciels concernés, ici Samba et Postfix.

2.4 Choix de la structure de l'arbre

2.4.1 Organisation des données dans un annuaire

Les données dans un annuaire LDAP sont organisées dans une structure modélisée par un arbre où nous pouvons distinguer deux catégories d'objets :

- les conteneurs qui peuvent-être considérés comme le départ d'une nouvelle branche,
- les feuilles qui sont les terminaisons des branches.

Un conteneur peut contenir les deux catégories d'objet, soit des conteneurs, soit des feuilles.

2.4.2 Organisation des conteneurs et des feuilles

Concrètement les conteneurs correspondront aux centres et unités de recherches et les objets feuilles correspondront aux utilisateurs et machines.

Il est également possible d'utiliser des objets conteneurs à des fins de rangements. Ainsi les conteurs USER, GROUP et COMPUTER, stockent respectivement les comptes utilisateurs, les groupes Posix et les comptes machines (figure 1). Ils n'ont pas d'autre utilité que de faciliter la lecture des données lors de la navigation.

2.4.4 Architecture générale de l'arbre

Bien que le point de départ de notre annuaire ait été Rennes, nous avons pris soin de créer un arbre qui pourrait s'étendre à l'organisation nationale de l'INRA. Pour cela nous avons organisé notre arbre en fonction de la structure géographique des Centres et Unités. Ce qui correspond à la réalité de la répartition des ressources informatique à l'INRA.

Le format d'entrée des premières données est un fichier LDIF qui nous a permis de créer le « squelette » de l'annuaire avant de commencer à créer les premiers comptes utilisateurs et machines.

Ex :

```
dn: o=inra,c=fr
objectClass: top
objectClass: organization
o: inra
description: Institut National de la Recherche Agronomique
businessCategory: EPST

dn: ou=rennes,o=inra,c=fr
objectClass: top
objectClass: organizationalUnit
ou: rennes
```

Il s'agit ici des deux premières entrée permettant de démarrer l'arbre.

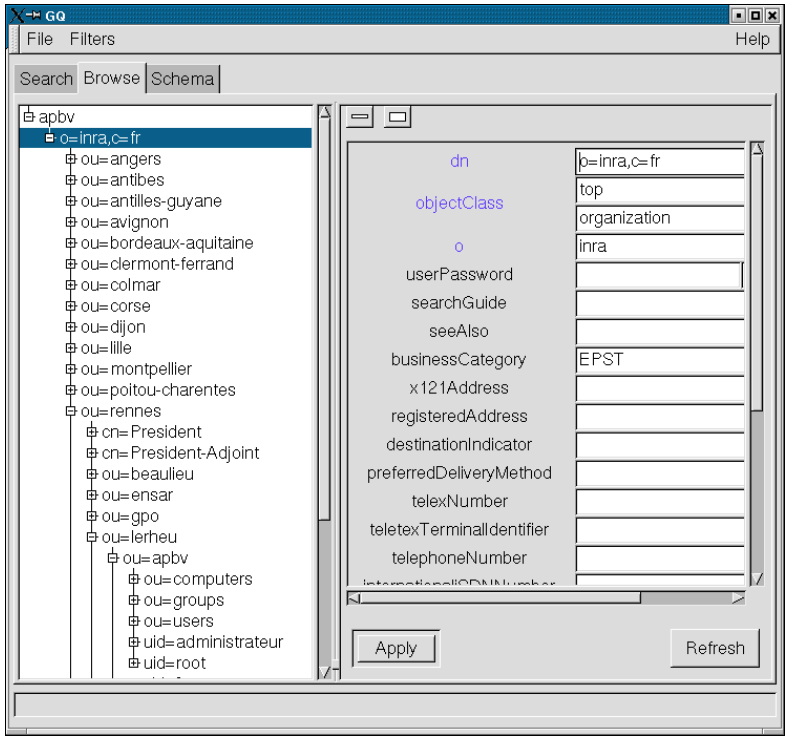

Figure 2 : Vue globale de l'arbre

2.5 Choix des Objets

2.5.1 Centre et Unités de recherches

Il s'agit des objets conteneurs qui forme la base de la structure de notre arbre. L'objet LDAP classique est l'OU ou *OrganizationalUnit.* C'est l'objet conteneur le plus classiquement utilisé dans ce rôle et que l'on retrouve d'ailleurs dans Active Directory. Dans certains cas ces conteneurs ont été associés à des objets *PosixAccount* afin de pouvoir gérer des droits (cf 2.5.3).

2.5.2 *Les comptes utilisateurs*

L'un des principaux objectifs de l'utilisation de LDAP est la gestion des comptes de tous les utilisateurs, aussi bien au sens Linux que Samba (Windows). Un objet utilisateur tel que nous l'entendons n'existe pas dans LDAP et il est nécessaire de le définir en fonction des données administrative (mail, téléphone, photo …) et système dont nous avons besoin.

Ceci nous a conduit à définir un utilisateur comme un cocktail de différents objets LDAP :

- un *PosixAccount*, afin de pouvoir gérer le compte Linux,
- un *SambaAccount* pour la gestion du compte Samba,
- un *InetorgPerson* pour pouvoir disposer d'informations pouvant alimenter une base de données et disposer de l'adresse mail nécessaire pour l'utilisation avec Sympa.

Bien sûr ces objets nécessitent des héritages d'où la présence des objets *Top*, *account* etc… selon les versions d'OpenLDAP.

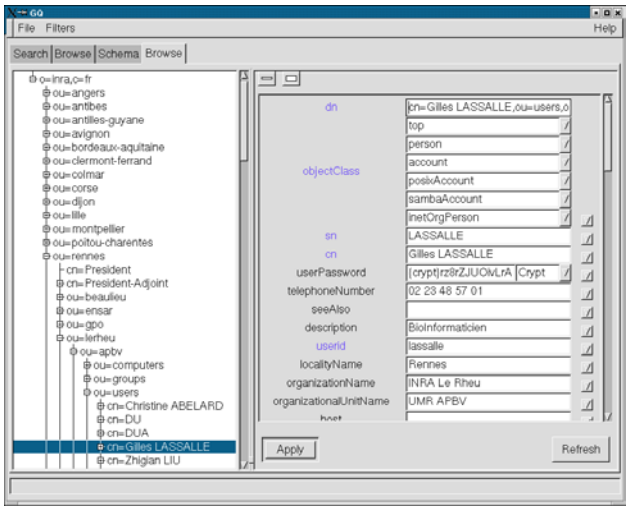

Figure 3 : Vue d'un compte utilisateur

2.5.3 *Les comptes machines*

Afin de pouvoir utiliser Samba comme PDC, nous avons été contraints de créer et de gérer les comptes ordinateurs. Comme pour les comptes utilisateurs, nous avons construit les comptes machines par un ensemble d'objet LDAP dont :

- *PosixAccount* : pour gérer le compte machine sous linux,
- *SambaAccount* : pour gérer le compte Samba,
- *Device* : pour avoir des informations telles que numéro de série,
- *IpHost* : pour l'adresse IP et autres informations réseau.

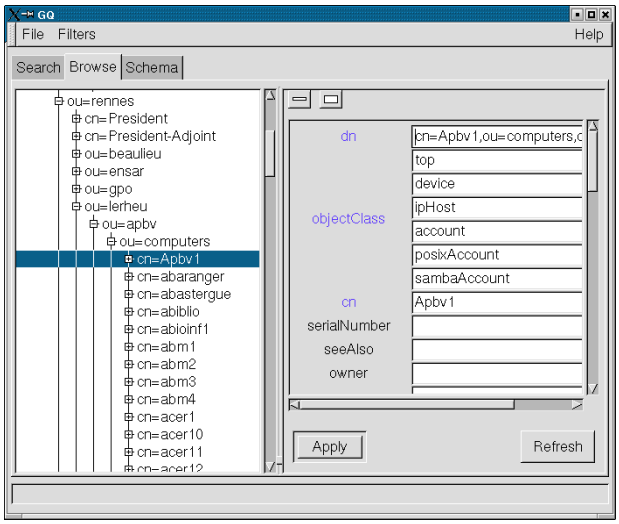

Figure 4 : vue d'un compte machine, détail des objets.

Nous utilisons actuellement toutes ces informations, afin de gérer notre parc informatique ainsi que pour contrôler l'accès à notre intranet via un script PHP.

2.5.4 Les groupes

Il s'agit de gérer les groupes de travail et les équipes de recherche de notre UMR. L'objet LDAP utilisé est *PosixGroup*. Cette organisation, permet à la fois de gérer les droits Unix-Linux ainsi que les Mailing-Lists sous Sympa. Ces objets sont aussi des feuilles car l'information sur l'appartenance des membres est en fait un attribut de l'objet *PosixGroup*.

2.5.5 Des objets Feuilles divers

L'utilisation de l'objet LDAP *OrganizationalRole* nous permet de gérer le poste de directeur d'unité et tous les postes qui correspondent à la philosophie d'un rôle organisationnel (président de centre, etc. …cf. figure 1)
Pour figurer les unités de recherche c'est l'objet OrganisationalUnit qui a été utilisé.

3 Gestion de la sécurité

3.1 Les Access List (Acls)

L'utilisation des ACLs, présentes dans le fichier *slapd.conf*, permet de gérer les droits d'accès aux différentes ressources, en lecture, écriture et navigations des utilisateurs de l'annuaire. La syntaxe des ACLs est relativement simple. Ce qui devient compliqué, c'est la gestion avancée de droits d'accès par utilisateurs et/ou par groupe, qu'il faut gérer de manière quasi-individuelle ou au cas par cas.
ex :

```
access to attribute=userPassword
by dn="cn=root,o=inra,c=fr" write
by anonymous auth
by self write

access to *
by dn="cn=root,o=inra,c=fr" write
by * read
```

Dans cet exemple, la première ligne de chaque ACLs indiquent les ressources auxquelles on veut accéder, ici, soit l'attribut *UserPassword* soit tous les attributs (*). Les lignes suivantes indiquent qui a le droit de faire quoi.
L'utilisateur est décrit soit :

- par son DN,
- par * , pour tout le monde,
- par self, pour ses propres données.

Les droits sont *auth, read, write ou none* permettent respectivement l'authentification, la lecture, l'écriture ou rien du tout. En fait, ici la seconde ACL permet un accès sur tout l'annuaire :

- en lecture à tout le monde,
- en écriture à l'administrateur de l'annuaire (ici : *cn=root,o=inra,c=fr).*

La mise en place d'une stratégie minimale d'ACLs est nécessaire afin de restreindre les accès non souhaités et de protéger les données de l'annuaires tout en respectant les conventions de la CNIL (Commission Nationale de l'Informatique et des Libertés).

3.2 La réplication

La réplication est une nécessité lorsque l'on utilise un annuaire LDAP en production. Il s'agit d'un mécanisme qui permet d'avoir plusieurs copies de l'annuaire actif et de gérer ainsi une tolérance de panne. On paramètre ainsi chaque machine pour qu'elle puisse s'adresser à sa réplique locale.

L'outil qui permet d'implémenter la réplication avec OpenLDAP est slurpd. Lors de la compilation il suffit d'activer l'option –with-slurpd.

Le mécanisme de réplication fonctionne sur une relation maître - esclaves. Chaque protagoniste (maître ou esclave) doit posséder un serveur slapd. Seul le maître possède à la fois slapd et slurpd. La configuration de l'un et de l'autre se fait là encore dans le fichier slapd.conf. Le mécanisme de réplication s'exécute lorsqu'une modification intervient sur le Maître, le fichier slurpd.replog est alors renseigné et le démon slurpd après l'avoir lu envoie les modifications vers les esclaves qui mettent à jour leur annuaire.

Par contre la mise à jour de l'esclave vers le maître même si elle est décrite dans la littérature ne fonctionne pas toujours. De plus il faut impérativement utiliser SASL comme mode d'authentification. Une réplique esclave ne peut pas mettre à jour un maître directement. C'est l'application cliente, qui après avoir tenté une mise à jour sur l'esclave est redirigée et reconnectée vers le maître (rebind) afin que ce dernier prenne en compte les modifications et les répercute sur les différents esclaves. Or le mode d'authentification « simple » ne permet pas ce « rebind », il faut utiliser SASL. Une autre solution est de rendre chaque serveur maître, esclave de l'autre.

Selon les stratégies de réplication (partielle ou complète), on peut laisser des droits sur l'annuaire, à différents utilisateurs, différents selon les conteneurs et selon les répliques(en fait les sites distant). On peut ainsi avoir un annuaire LDAP à jour et complet et qui est en fait renseigné, sur les différents sites, par différentes personnes qui n'ont des droits que sur leur conteneurs. C'est l'un des intérêts de pouvoir mette à jour un LDAP maître à partir d'un esclave.

3.3 La cryptographie et l'authentification

.

LDAP utilise plusieurs modes d'authentification :

- Simple, sans cryptographie ni protocole particulier,
- Kerberos ,
- SASL .

SASL (Simple Authentication and Security Layer) est une méthode qui permet d'ajouter un support d'authentification à un mécanisme de connexion simple. Il permet d'identifier et d'authentifier un compte sur un serveur et le cas échéant de négocier une méthode de protection entre le client et le serveur.

SASL a été mis en place lorsque l'on s'est aperçu que la mise à jour de la réplique master n'était pas possible à partir d'une réplique esclave, le mode « simple » ne permettant pas un « rebind » de l'application cliente vers le serveur maître.

Pour la cryptographie, LDAP peut s'appuyer sur TLS/SSL mais certaines versions d'OpenLDAP deviennent alors très lente dans les mécanismes d'authentification.

4 Utilisation et exploitation de l'annuaire

4.1 Gestion de l'authentification

4.1.1 Sous Linux

L'authentification sous Linux via LDAP s'appuie sur les PAM (Pluggable Authentification Modules).Les PAM permettent de gérer la politique d'authentification sous Unix-Linux sans recompiler quoique ce soit. Les dernières distributions Linux sont livrées en général avec les PAM. Sous linux, lorsque les PAM sont en place, on les trouve soit dans */etc/pam.d* soit */etc/pam.conf*. Ils s'appuient sur 4 types de modules qui sont :

- Auth : gère l'authentifie l'utilisateur,
- Account : gère les restrictions du compte,
- Password : gère les mots de passe.
- session : gère ce qui concerne l'ouverture d'une session, avant et après.

A ces 4 modules, sont attribués des contrôles qui sont entre-autres:

- Required : nécessaire tout en continuant à tester les autres modules,
- Requisite : nécessaire et suffisant,
- Sufficient : suffisant mais pas nécessaire.

Afin d'implémenter cette fonctionnalité, il faut bien sûr des objets LDAP valides (PosixAccount …), installer les bibliothèques permettant de gérer cette authentification, et paramétrer les PAM et les différents fichiers entrant en jeux : *nsswitch.conf, libnss-ldap.conf, pam_ldap.conf*. Le fichier minimal à modifier dans /etc/pam.d est « *login* » auquel il faut rajouter la ligne : "auth sufficient pam_ldap.so"

Cette modification est nécessaire et souvent suffisante pour permettre aux utilisateurs gérer dans l'annuaire d'accéder aux ressources de la machine. J'ai pu constater que modifier d'autres fichiers pouvait provoquer des lourdeurs et des ralentissements dans les mécanismes d'authentification.

Les modifications à apporter dans le fichier *nsswitch.conf* sont très simples :
Ex :

```
passwd:         compat ldap
group:          compat ldap
shadow:         compat ldap
```

Par contre il est primordial de conserver le premier mécanisme d'authentification (soit *file*, soit *compat*) car si l'annuaire LDAP est indisponible, alors même le compte *root* ne pourra plus se loguer.

Les fichiers *Libnss_ldap.conf* et *pam_ldap.conf* sont très similaires. Ces fichiers indiquent quel serveur LDAP rejoindre ainsi que les paramètres de connexions, ce qui permettra d'envoyer les informations d'authentification et de recevoir les informations systèmes.

Exemple les premières lignes d'un de ces fichiers :

```
host apbv.rennes.inra
# The distinguished name of the search base.
base ou=rennes,o=inra,c=fr
ldap_version 3
# stored in /etc/ldap.secret (mode 600)
rootbinddn cn=root,o=inra,c=fr
port 389
# The search scope.
scope sub
#scope one
#scope base
pam_password crypt
```

Scope permet d'indiquer le niveau de recherche dans l'arbre, sous sur le niveau présent (base), sous sur un niveau sous-jacent(one) sous dans toute l'arborescence descendante(sub).

Les bibliothèques *Libpam_ldap* et *Libnss_ldap* interviennent dans les échanges d'informations lors du login. Libpam_ldap permet d'envoyer les informations de login au serveur LDAP et *libnss_ldap* permet de récupérer les informations sur l'utilisateur dont le login est valide.

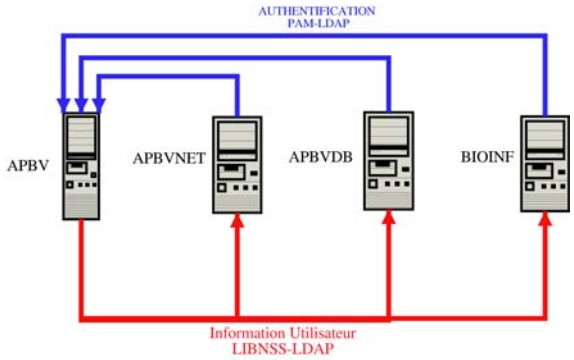

Figure 5 : schémas du mécanisme d'authentification.

4.1.2 Avec Samba

L'association LDAP+Samba permet de faire un PDC très performant autorisant l'authentification sur un domaine NT de client Windows 95 à XP Pro. La première chose à faire pour pouvoir utiliser Samba avec LDAP est de le compiler avec l'option : *--with-ldapsam* ,
La seconde condition est d'avoir un annuaire valide avec le schéma Samba afin d'utiliser les objets LDAP *SambaAccount*. Pour indiquer au serveur Samba qu'il doit se connecter à un serveur LDAP, il faut ajouter le lignes suivantes dans le fichiers smb.conf :

```
[global]

    # parametres de connections LDAP

        ldap server = ldap-server
        ldap port = 389
        ldap suffix = "o=inra,c=fr"
        ldap admin dn = "cn=root,o=inra,c=fr"
        ldap ssl = no
```

Samba est tout à fait capable de fonctionner de façon autonome par rapport au serveur Unix-Linux qui l'héberge. C'est à dire qu'il peut parfaitement authentifier des comptes utilisateurs sur serveur LDAP distant alors que la machine locale utilise une stratégie de comptes locaux complètement différents.

L'activation et la gestion des comptes utilisateurs Samba se fait avec *smbpasswd*. Une fois le compte créé sous LDAP avec l'objet *SambaAccount*, il suffit, pour activer le compte, de lancer la commande « smbpasswd nom_user » et après avoir renseigné le mot de passe, Samba se charge de renseigner tous les champs, en rapport avec Samba, dans LDAP.
Normalement, si un compte Posix, validé sous LDAP fonctionne, alors en utilisant cette commande avec « -a », un objet SambaAccount va être rajouté au compte « nom_user » et de la même manière tous les champs concernant Samba dans l'annuaire seront renseignés.

Afin de créer un PDC avec Samba et LDAP et permettre aux machines Microsoft NT (NT4.0, 2000 et XP) de venir s'authentifier sur le domaine Samba, il faut gérer les comptes machines. LDAP permet cette gestion aussi simplement que ceux des utilisateurs. La seule différence, c'est qu'il faut utiliser « smbpasswd –a –m nom_machine » pour l'activation d'un compte machine. Le *userid*, doit-être du type « nom-machine$ » dans l'annuaire. Enfin pour finir, il faut enregistrer le poste client sur le domaine, avec le compte root renseigné dans LDAP.
Avec cette gestion de comptes machines et utilisateurs dans LDAP, le fichiers smbpasswd est vide. Tous les renseignements dont Samba à besoin, sont dans l'annuaire. Afin de compléter notre architecture, nous avons utilisé le système de fichiers journalisé XFS de SGI en patchant un noyau 2.4.20 afin d'utiliser les ACLs et les Quotas.

4.1.3 avec Apache

Le module auth_ldap installé avec Apache permet de gérer l'authentification directement grâce à l'annuaire. Décrit sur le site www.rudedog.org ce module permet de mutualiser, encore une fois, l'authentification par LDAP.

4.2 Interrogation et manipulation de données.

Le langage d'interrogation et de requêtes est un langage propre à LDAP. Le package ldaptool permet d'installer divers outils permettant de se connecter à un annuaire LDAP, de l'interroger et de manipuler les données.
Les filtres sous les outils qui permettent d'interroger très simplement un annuaire. Il s'agit d'un langage d'interrogation de données complètement différents de SQL et qui est cependant optimisé pour l'extraction de données dans un annuaire LDAP.
ex:

(&(objectclass=PosixAccount)(userid=lassalle))

Ce filtre va permettre de sélectionner l'objet qui est à la fois un PosixAccount et dont le userid est "lassalle".
En retour nous recevrons toutes les informations pour cette entrée. Cette syntaxe est basée sur l'utilisation d'opérateurs logiques qui en font un langage est très pratique et très rapide.

4.3 Gestion de Mailing-Lists avec Sympa

Le gestionnaire de Mailing-Lists SYMPA développé au sein du CRU, utilise de façon très performante les données d'un annuaire LDAP. Il permet d'une part de gérer l'authentification à l'interface Web mais aussi de gérer des Mailing-Lists dynamiques.

Nous avons exploité ce mécanisme à l'aide de la procédure « requêtes en deux passes » qui dans un premier temps scanne le groupe concerné par le premier filtre et ressort tous les userid et en suite va chercher grâce au second filtre, toutes les adresses mails en rapport avec les userid précédemment sélectionnés.

Il faut bien sûr avoir un compte utilisateur possédant un champ « mail » disponible avec l'objet *InetorgPerson*.

La grande force de cette alliance, c'est que le simple fait de rajouter un utilisateur dans un groupe faisant référence à une mailing-lists lui permet d'une part d'accéder aux partages destinés à ce groupe mais aussi de faire partie de la mailing-lists. Le serveur de mails utilisé est Postfix, qui est aussi capable de gérer les boites à lettres directement avec LDAP.

4.4 Utilisation avec Postfix

Postfix est un MTA (Mail Transport Agent) qui remplace très avantageusement Sendmail. De plus, Postfix est livré avec un schéma pour LDAP qui permet de gérer tous les paramètres d'un compte mail-Postfix via l'objet LDAP *PostfixUser*. Dans notre architecture Postfix est le serveur de mail sur lequel Sympa s'appuie. Comme nos boites mails sont gérés par d'autres instances que notre unité de recherche, nous n'exploitons que quelques comptes internes alliant Postfix et LDAP , le reste n'est qu'une redirection des mails vers le serveur principal.

4.5 Utilisation avec les développements internes et l'intranet

4.5.1 Utilisation comme annuaire

La simple présence de l'annuaire nous a fait prendre conscience des fonctionnalités que l'annuaire, en tant que source de données, pouvait nous apporter. Actuellement, nous sommes en train de développer un outils de gestion du personnel, adapté à notre département, en architecture multi-tiers. LDAP est la base des informations concernant l'organisation du département et des unités. Les informations qui ne peuvent pas être gérées dans l'annuaire sont prises en compte dans une base de données Postgresql. L'ensemble est interfacé via Apache avec du Php.

4.5.2 avec PHP

La double gestion des utilisateurs et des machines, nous permet d'une part de gérer l'accès à notre intranet par login et mot de passe, très classiquement, (tous les mots de passe sont synchronisés (Linux, Samba, Apache)) et d'autre part d'identifier les machines par leur adresses IP et de les laisser accéder ou non selon leur présence dans l'annuaire. Le grand intérêt de cette architecture est la centralisation de toutes les données, et la facilité de consultation et de gestion avec Php

5 Conclusions et Perspectives

L'implémentation d'OpenLDAP dans notre structure, nous a permis de mettre en place un réseau, n'utilisant que des produits sous licence GPL et d'avoir une architecture très performante.

L'administration d'un réseau de serveurs Linux devient très souple et très simple. Le point capital est d'appréhender le fonctionnement d'un annuaire LDAP et la façon d'organiser les données. Il s'agit du point peu documenté, qui peut décourager très vite une personne ne connaissant pas cet outil.

Si le point de départ de notre implémentation d'OpenLDAP était de faciliter l'administration du réseau, très vite, nous avons organisé les informations contenues dans l'annuaire afin de l'utiliser comme base de données. C'est un outil qui offre des perspectives et des fonctionnalités très complètes sur le plan de l'administration système et réseau, mais il ne faut pas oublier sa fonction d'annuaire.

Un point faible, cependant, il n'existe pas encore d'outil permettant à la fois de gérer le contenu de l'annuaire et les droits d'accès à ce même annuaire. Il faut reconnaître que même si les ACLs permettent de gérer les droits, elles sont quelques peu rébarbatives à utiliser. L'outil Nwadmin de Novell, remplit ce rôle très bien. Nous pouvons regretter qu'un tel outil n'existe pas encore sous Linux.

Nous avons en cours plusieurs projets, aussi bien système que réseau, concernant la mise en place des auto-montages au travers LDAP et les informations DNS, mais aussi le développement d'un système d'information et de gestion déployé au sein du département de Recherche de Génétique et d'amélioration des plantes.

6 Ressources bibliographiques

Documentation LDAP (complète en anglais) :
http://www.umich.edu/~dirsvcs/ldap/doc/

le site OpenLDAP :
http://www.openldap.org

Samba-ldap :
Le site d'idealx (collection de scripts et d'outils de migration)
http://www.idealx.org/prj/samba/index.fr.html

auth_ldap (apache) :
http://www.rudedog.org/auth_ldap/1.6/auth_ldap.html

PAM :
http://www.docmaster.org/articles/linux109.htm
(en français)

Cyrus IMAP :
http://www.arrayservices.com/projects/Exchange-HOWTO/html/x285.html

Listes de diffusion :

Pour ldap :
www.cru.fr

et Samba :
http://listes.ujf-grenoble.fr/wws