Authentification par certificats : l'importance du gestionnaire de profils

Roland Dirlewanger
CNRS – Délégation Aquitaine et Poitou-Charentes
Esplanade des Arts et Métiers, BP105
33402 TALENCE Cédex
rd@dr15.cnrs.fr

Résumé

Les mécanismes standards des serveurs Apache/mod_ssl ou IIS permettent facilement d'autoriser ou d'interdire l'accès à une application WWW en fonction de certificats. Malheureusement, ces mécanismes se révèlent insuffisants dès lors que les informations fournies par l'application doivent être limitées aux seules données auxquels le titulaire du certificat est en droit d'accéder.

Lors de la mise en place de l'accès authentifié par certificat au Système d'information régional, une réflexion a été menée à la Délégation Aquitaine et Poitou-Charentes autour de la notion des profils d'utilisateurs, c'est à dire de l'ensemble des permissions dont ils disposent pour accéder à une application. Certains profils peuvent être déduits du certificat lui-même, d'autres pas. Dans ces cas là, il est nécessaire de recourir à un mécanisme externe, appelé le gestionnaire de profils.

A travers quelques exemples significatifs ayant des besoins d'authentification distincts, nous déduisons les fonctionnalités indispensables d'un gestionnaire de profils. Ensuite, nous présentons comment ces fonctionnalités ont été implémentées dans Ldapauth, le gestionnaire de profils développé à la Délégation.

Cet outil a incontestablement facilité l'écriture et le déploiement des applications de l'Extranet de la Délégation. Il a évolué au cours du temps, passant d'une base Postgres à un annuaire LDAP. Récemment, l'IETF a normalisé les certificats d'attributs pour répondre aux besoins d'authentification des applications. Sont-ils une alternative à un gestionnaire de profils ?

Mots clefs

PKI, IGC, authentification, certificats d'authentification, certificats d'attributs, gestion de profils, gestion de rôles, LDAP, X509, SSL, TLS, HTTPS, Apache, mod_ssl.

1 Introduction

L'apparition des premiers serveurs WWW s'est immédiatement accompagnée de la question de permettre ou non l'accès à certaines pages WWW selon l'utilisateur qui s'y connecte. Pendant des années, il n'y a eu que deux types de réponses à ce besoin : l'utilisation du mécanisme d'authentification basique inclus dans le protocole HTTP [1] ou bien l'utilisation de listes de contrôles d'accès sur l'adresse IP des clients, ces listes pouvant être configurées au niveau des implémentations des serveurs ou des équipements de protection du réseau.

Les limites de ce modèle sont évidentes. Pour ses tâches quotidiennes, l'utilisateur doit mémoriser un nombre de plus en plus impressionnant de combinaisons d'identificateurs et de mots de passe. Pour les administrateurs du site WWW, la gestion des accès par adresse IP est un problème difficile en soi. Il est devenu quasiment insolvable, notamment dans le cas des postes de travail mobiles et ceux dont l'adresse a été allouée dynamiquement.

La problématique est rendue encore plus complexe dans des organismes fortement déconcentrés comme le CNRS. L'utilisateur dans un laboratoire est confronté à une multitude de services WWW : ceux de son laboratoire, de sa délégation régionale, de son université, de son Département scientifique, du Système d'information national, etc. Chacun de ces services offre ses propres mécanismes d'authentification, indépendants les uns des autres.

Dès la fin des années 1990, des études ont été lancées par l'Unité réseau du CNRS (UREC) pour mettre en place une Infrastructure de gestion de clés (IGC). L'un des buts de cette IGC est d'utiliser les certificats X509 [2] comme mécanisme d'authentification générique pour l'accès aux divers systèmes d'informations de l'établissement.

A partir de 2001, les délégations de Bordeaux et de Toulouse ont mené des expériences pilotes pour le déploiement des certificats issus de l'IGC du CNRS et pour l'accès aux informations financières des laboratoires via ces certificats. Durant la même période, des expériences similaires ont été conduites dans certains laboratoires et dans le Département scientifique des Sciences et techniques de l'information et de la communication (STIC).

L'objectif d'utiliser le même certificat pour s'authentifier à des applications locales aux laboratoires, régionales ou nationales indépendantes les unes des autres a bien été atteint. Toutefois, les certificats ne contiennent aucune information sur les permissions accordées à leur titulaire par ces diverses applications . Il revient donc à l'application de gérer elle-même l'association entre le certificat de l'utilisateur et les permissions qu'on lui accorde.

A la Délégation Aquitaine et Poitou-Charentes, nous avons souhaité mettre en oeuvre un mécanisme, appelé gestionnaire de profils, qui résout ce problème d'une façon générique pour toutes les applications authentifiées par certificat. Dans la suite, nous montrerons à travers les besoins de certaines applications comment se dessinent les spécifications d'un tel mécanisme et comment il a été implémenté dans le système d'information régional de la délégation.

2 Besoins des applications en terme d'authentification et contrôle d'accès

2.1 Quelques rappels

Dans toute communication, électronique ou pas, l'**authentification** consiste pour l'un des partenaires de cette communication à vérifier que l'autre est bien celui qu'il prétend être. Le mécanisme le plus courant est le couple identificateur et mot de passe. Les protocoles *Secure Socket Layer* (SSL [3]) et *Transport Layer Security* (TLS [4]) utilisent des certificats X.509 pour l'authentification : dans la phase d'initialisation du protocole, le serveur (toujours) et le client (sur demande du serveur) échangent, entre autres, leurs certificats respectifs, des données aléatoires et la signature électronique du dialogue d'initialisation. L'authentification consiste donc pour chacun à vérifier plusieurs points : le certificat présenté par l'autre a été émis par une autorité de confiance, est en cours de validité, n'a pas été révoqué et la signature du dialogue est conforme. Si au moins une de ces conditions n'est pas remplie, la connexion est interrompue.

Le **contrôle d'accès** consiste à vérifier si une entité authentifiée précédemment est autorisée à accéder à une ressource ou pas. Par exemple, un utilisateur connecté à un système Unix ou Windows NT4, donc dûment identifié, n'est pas nécessairement autorisé à accéder à tous les fichiers stockés sur le système.

Les exemples qui suivent sont significatifs de besoins d'authentification et de contrôles d'accès nécessaires aux diverses applications ou services qu'on rencontre dans la plupart des laboratoires ou délégations régionales. Pour chacun d'eux, nous donneront des indications sur la façon de les mettre en œuvre via les mécanismes classiques fournis par les serveurs WWW comme Apache ou IIS. Nous verrons que ces mécanismes ne permettent pas toujours de répondre à tous les besoins.

2.2 Intranet d'établissement ou de laboratoire

Bien souvent, le serveur WWW des laboratoires s'accompagne d'un espace qui est restreint aux membres du laboratoire. Il contient des documents et des informations internes au laboratoire qu'on ne souhaite pas divulguer au grand public. On appelle cet espace « l'Intranet du laboratoire ».

Il y a diverses façons de réaliser un Intranet. On peut configurer le serveur WWW lui-même pour interdire l'accès à ces pages depuis des machines dont le nom ou l'adresse IP ne fait pas partie du plan d'adressage du laboratoire. On peut aussi configurer les systèmes de protections du réseau du laboratoire (pare-feux ou filtres sur le routeur d'entrée) pour interdire tout accès vers l'adresse IP du serveur WWW depuis des réseaux externes au laboratoire.

Pour un établissement comme le CNRS, ce type de solution est difficilement réalisable. En effet, les réseaux IP des laboratoires étant dans la plupart des cas gérés par les partenaires du CNRS (universités, écoles, autres établissements), il est très difficile de collecter les adresses et les noms de domaines qui correspondent effectivement à des unités CNRS.

Dans tous les cas, ce type de solution est peu conciliable avec les besoins en terme de mobilité de nos utilisateurs. Un chercheur en mission peut souhaiter accéder à l'Intranet de l'établissement ou de son laboratoire depuis le portable qu'il connecte sur le lieu de sa mission ou dans sa chambre d'hôtel.

Une solution classique est d'associer un mot de passe à l'espace Intranet du laboratoire ou du CNRS dès lors qu'on ne se connecte pas depuis une machine authentifiée comme faisant partie du laboratoire. C'est malheureusement la principale cause de multiplication des mots de passe dans notre système d'information.

Pour les personnels des unités du CNRS, les certificats contiennent dans l'attribut *organisationalUnit* (OU) le laboratoire auquel appartient le titulaire, dans l'attribut *Organisation* (O) la chaîne « CNRS » et dans l'attribut *Country* (C) la chaîne « FR ». On peut donc utiliser les caractéristiques des certificats CNRS pour mettre en œuvre des autorisations du style « service ouvert à tous les membres du laboratoire XYZ » ou bien « service ouvert à tous les personnels des unités du CNRS ». A titre d'exemple, la configuration pour Apache qui est utilisée sur notre serveur pour la diffusion du logiciel Xlab. Elle permet de restreindre l'accès à cette distribution uniquement aux titulaires d'un certificat CNRS.

```
<Location /Extranet/CNRS/Xlab>
SSLRequireSSL
SSLVerifyClient require
SSLRequire      %{SSL_CLIENT_S_DN_O} eq "CNRS" and %{SSL_CLIENT_S_DN_C} eq "FR"
</Location>
```

La configuration ci-dessus est statique et ne nécessite aucune intervention ultérieure de l'administrateur du site qui rend se service. Dès lors qu'un utilisateur détient un certificat CNRS, il peut accéder au service. C'est le cas de figure le plus simple où l'application ne prend en compte ni l'authentification, ni les contrôles d'accès. Ces deux aspects sont pris en charge directement par le serveur WWW qui accorde ou pas l'accès à l'application selon une caractéristique commune contenue dans tous les certificats. C'est une fonctionnalité qui paraît essentielle pour ce type d'applications :

Le contrôle d'accès doit pouvoir s'effectuer selon une caractéristique commune aux certificats.

2.3 Consultation des remboursements de frais de mission

Les laboratoires sollicitent souvent les services financiers des délégations en ce qui concerne le décompte du remboursement des frais de mission. En effet, des limitations dans l'actuel système d'informations financières font que les laboratoires n'ont accès qu'au montant total remboursé à l'agent à son retour de mission. La façon dont se décompose cette somme n'est accessible que depuis la Délégation.

Afin d'améliorer le service rendu aux laboratoires, notre délégation a mis en place un service en ligne authentifié par certificat qui permet :

- à chaque personne titulaire d'un certificat d'afficher ses propres décomptes
- au gestionnaire du laboratoire d'afficher les décomptes de toutes les missions des membres du laboratoire

Le deuxième point concernant l'accès pour le gestionnaire fait partie de la famille d'applications regroupées sous le terme d'Extranet financier de la Délégation et décrit plus loin. Nous nous limiterons dans ce paragraphe au point concernant l'utilisateur souhaitant afficher ses propres décomptes.

A priori, la configuration au niveau du serveur WWW est la même que précédemment. Si l'utilisateur dispose d'un certificat CNRS, on peut extraire de ce dernier l'attribut « OU » du certificat de l'utilisateur pour déterminer son laboratoire et l'attribut CN pour déterminer son prénom et son nom. L'application peut interroger le système d'informations financières pour déterminer les missions qui correspondent à ces trois critères.

Deux problèmes se posent avec cet algorithme. D'une part, il ne permet pas de distinguer les missions de deux personnes homonymes à l'intérieur d'un même laboratoire. D'autre part, la sélection des informations est rendue difficile dans le cas où l'utilisateur a obtenu son certificat avec un attribut CN qui correspond à ses noms et prénoms usuels alors que le système d'informations financières utilise les noms et prénoms de l'état civil.

Habituellement, ces deux problèmes sont résolus de façon triviale par l'utilisation du numéro de l'agent dans le système d'information de l'organisme. Ce numéro peut être inclus aisément dans un certificat sous la forme d'une extension.

Toutefois, dans le cas de l'IGC du CNRS, toute personne travaillant dans une unité CNRS peut recevoir un certificat. En pratique, moins d'un tiers d'entre elles sont des agents CNRS. Seul le numéro INSEE est unique à travers la centaine de systèmes d'informations des établissements partenaires du CNRS. En pratique, les contraintes réglementaires et le caractère confidentiel de ce numéro ne permettent pas son utilisation dans un certificat qui est, par essence, public.

Actuellement, ce problème n'a donc pas de solution. L'application devra donc gérer en interne les équivalences entre les divers noms et prénoms sous lesquels est connu l'utilisateur :

Un utilisateur peut avoir plusieurs identités.

2.4 Extranet financier de la Délégation

Le système d'informations financières du CNRS est composé essentiellement de deux applications. La première, appelée Xlab, est utilisée pour la gestion des laboratoires. La seconde, appelée GCF, est utilisée par les services financiers des délégations régionales. Ces deux applications communiquent par échange de fichiers.

Le processus d'échange ne porte pas sur toutes les données, si bien que certaines informations intéressant directement les laboratoires ne sont présentes que dans la GCF, et réciproquement. Dans certaines délégations, le couple Xlab / GCF a donc été complété par une application WWW qui permet aux directeurs et aux gestionnaires d'accéder aux informations financières qui concernent leur laboratoire. Cette application est appelé l'Extranet financier.

L'authentification dans une telle application peut se faire soit via un identificateur et un mot de passe, soit via un certificat. Si on choisit d'utiliser des certificats, l'application récupère le code du laboratoire contenu dans l'attribut « OU » du certificat de l'utilisateur afin de limiter l'affichage aux données concernant le laboratoire de ce dernier. On souhaite néanmoins réserver l'accès aux informations financières du laboratoire uniquement aux directeurs et gestionnaires d'unité. Il faut donc que la configuration des contrôles d'accès prenne en compte la liste exhaustive des utilisateurs autorisés à accéder à l'application. Heureusement, à la fois Apache et IIS offrent des moyens simples de configurer cette liste.

Il y a malgré tout deux problèmes qui se posent :

- certains directeurs d'unités de recherche sont aussi directeurs de groupements de recherche. De même, certains gestionnaires le sont pour plusieurs laboratoires simultanément Ces utilisateurs doivent donc avec un même certificat pouvoir accéder aux informations financières de tous les laboratoires qu'ils dirigent ou qu'ils gèrent.

- la configuration des contrôles d'accès telle que décrite ci-dessus est bien souvent du ressort de l'administrateur ou de l'administratrice du site WWW qui n'a pas nécessairement la connaissance de qui est gestionnaire dans chaque laboratoire. En outre, il ou elle n'aura la connaissance de l'intitulé exact du certificat d'un utilisateur que lorsque ce certificat sera émis.

Cet exemple permet d'introduire une nouvelle notion qui est celle de **profil d'utilisateur**. Il s'agit de la liste des identificateurs avec lesquels l'utilisateur est autorisé à accéder à une application donnée. Par exemple, pour un gestionnaire, son profil vis à vis de l'Extranet financier est la liste des laboratoires qu'il gère. Le **gestionnaire de profils** est l'application qui va permettre de créer, modifier, supprimer l'association entre le certificat d'un utilisateur et son profil.

On retiendra également de cet exemple que :

Un utilisateur peut disposer de plusieurs identificateurs pour une même application.

2.5 Portail vers les applications authentifiées par mots de passe

Comme dans beaucoup d'organismes de taille comparable, le système d'informations du CNRS est constitué d'une multitude d'applications répondant chacune à un besoin précis d'une catégorie précise d'utilisateurs. Parmi ces applications, nombreuses sont celles qui utilisent des interfaces WWW avec des accès restreints authentifiés par mot de passe.

L'utilisateur doit donc maintenir d'une manière ou d'une autre un ensemble de triplets (adresse du service, identificateur, mot de passe). Le navigateur lui apporte une aide appréciable dans cette tâche : l'accès aux sites se fait à travers les signets (ou favoris), les données d'authentification sont mémorisées directement par le navigateur. Malgré tout, pour pouvoir utiliser ces mécanismes, il est nécessaire que l'utilisateur ait eu auparavant connaissance de l'adresse exacte de ces sites et qu'un identificateur et le mot de passe correspondant lui aient été fournis pour chacun d'eux. La pratique montre que ces informations ne sont jamais disponibles au moment où l'utilisateur en a besoin.

Nous avons mis en place une page WWW permettant d'accéder directement à ces différentes applications. Pour chacune d'elles, la page affiche son intitulé et un bouton de connexion. Un click sur ce bouton émet la requête HTTP permettant de

transmettre les données d'authentification telles qu'attendues par l'application. Cette opération est totalement transparente pour l'utilisateur.

Cette page, que nous pouvons qualifier de portail vers les applications authentifiées par mot de passe, a un accès contrôlé sur la base du certificat de l'utilisateur. Seules s'affichent les applications auxquelles ce dernier est autorisé à accéder. Il peut toutefois demander l'accès aux autres applications du portail par le biais d'un formulaire. Sa demande sera acceptée ou refusée par l'une des personnes de la Délégation habilitées à délivrer les accès aux diverses composantes du système d'information.

On retrouve dans la description de ce portail la notion de profil de l'utilisateur. Dans l'Extranet financier, le profil de l'utilisateur était la liste des laboratoires qu'il dirige ou qu'il gère. Ici, le profil est la liste des identificateurs des applications auxquelles il a accès.

Un nouveau besoin est mis en évidence à travers cet exemple : l'utilisateur doit être en mesure d'effectuer des requêtes pour modifier son profil vis à vis de telle ou telle application. Ces dernières sont transmises à un responsable de l'application qui accepte, modifie ou rejette ces demandes :

Un utilisateur peut demander à modifier son profil. Sa demande est acceptée ou non par une personne habilitée.

3 Fonctionnalités d'un gestionnaire de profils

3.1 Les solutions techniques : annuaire ou base de données ?

Comme nous l'avons vu au paragraphe 2.4, le rôle principal du gestionnaire de profils est, pour une application donnée, d'associer un certificat d'utilisateur et le profil de ce dernier. Le profil est lui-même défini comme la liste des identificateurs de l'utilisateur pour l'application en question.

Le cœur du gestionnaire de profils est donc une fonction ayant deux paramètres, le nom d'une application et un certificat. Elle retourne le profil correspondant à l'utilisateur titulaire de ce certificat. En pratique, une telle fonction peut s'appuyer soit sur une base de données relationnelle, soit sur un annuaire de type LDAP [5].

Dans les deux cas, il faut trouver une donnée unique extraite du certificat de l'utilisateur pour servir soit d'index dans des tables de la base de données, soit d'entrée dans l'annuaire LDAP. Nous avons choisi l'adresse de messagerie. Elle différencie de façon univoque les utilisateurs indépendamment de l'IGC qui a émis leur certificat.

Dans le cas où le gestionnaire de profils est implémenté via une base de données relationnelle, il suffit d'une table à trois colonnes comprenant le nom de l'application, l'adresse électronique de l'utilisateur et son identificateur. Si un utilisateur a plusieurs identificateurs pour une application, on crée un enregistrement par identificateur. Un exemple d'une telle table est donné ci-dessous :

Application	Courriel	Identificateur
application1	user1@domaine1.fr	id1
application1	user1@domaine1.fr	id2
application2	user1@domaine1.fr	id3
application2	user2@domaine2.fr	id4

Figure 1 -*Gestion de profils avec une base de données*

Lors d'une connexion authentifiée par un certificat, le serveur HTTPS enrichit l'environnement des applications avec des informations issues de ce certificat. Parmi ces informations, il y a notamment l'adresse de messagerie de l'utilisateur. Pour déterminer le profil correspondant, il suffit de sélectionner tous les enregistrements de la table qui contiennent cette adresse de messagerie.

Dans le cas où le gestionnaire de profils est implémenté via un annuaire LDAP, on crée une branche pour chaque application. On rattache à cette dernière les profils pour les différents utilisateurs. L'exemple ci-dessus devient :

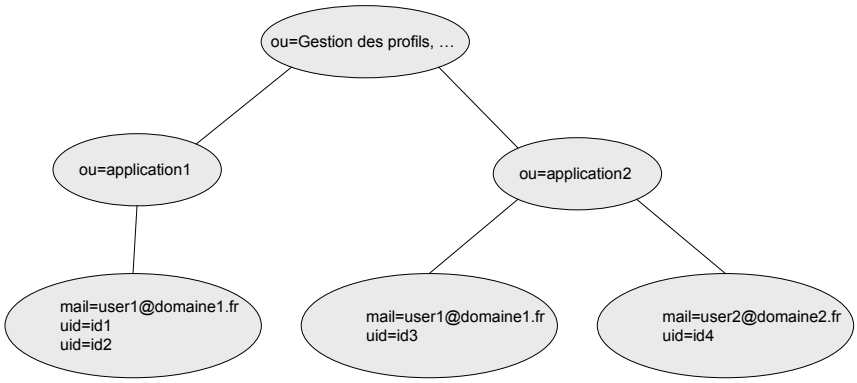

Figure 2 - *Gestion de profils avec un annuaire LDAP.*

Pour récupérer le profil d'un utilisateur vis à vis d'une application on effectue une requête LDAP à partir du nœud « `ou=$application, ou=Gestion des profils, …` », avec une profondeur de 1 et en filtrant sur l'adresse de messagerie issue du certitificat.

Il n'y a a priori pas d'argument décisif en faveur d'une solution ou d'une autre. Nous avons implémenté successivement l'une puis l'autre de ces solutions. LDAP a été préféré pour deux raisons : d'une part, la gestion des identificateurs multiples ainsi que des diverses identités d'un utilisateur est plus naturelle en LDAP. D'autre part, s'il faut partager des profils entre des applications situées sur différents sites (des laboratoires et leur délégation ou bien plusieurs délégations) il sera bien plus aisé de le faire en utilisant les mécanismes inhérents à ce protocole.

3.2 L'interface avec les utilisateurs

Il y a trois catégories d'utilisateurs :

- les utilisateurs finaux : ils effectuent des demandes vers le gestionnaire de profils afin de modifier les permissions qu'ils détiennent vis à vis de chaque application
- les administrateurs d'applications : ils peuvent créer, modifier, supprimer des profils pour les utilisateurs finaux. Ils peuvent aussi valider ou rejeter les demandes des utilisateurs finaux
- les administrateurs du gestionnaire de profils : ils peuvent créer, modifier, supprimer des applications. Ils configurent pour chacune d'elles la liste de ses administrateurs

L'interface la plus appropriée pour réaliser ces actions est le formulaire WWW authentifié par certificat. Il offre une ergonomie intuitive pour la majorité des utilisateurs. En outre, il est accessible indépendamment de la localisation physique de ces derniers et avec des garanties optimales en terme de sécurité.

Il est important que le gestionnaire de profils soit interfacé d'une manière ou d'une autre avec l'annuaire de l'IGC. En effet, on a vu que la clé qui désigne un utilisateur est son adresse de messagerie. Il est peu probable que l'administrateur connaisse toutes les adresses des utilisateurs et soit en mesure de les saisir sans erreur. La correspondance entre le nom de l'utilisateur et son adresse de messagerie est donc effectuée par l'intermédiaire d'une requête à l'annuaire de l'IGC.

Les paragraphes 2.4 et 2.5 montrent qu'il y a deux types d'applications. Dans l'Extranet financier, l'ensemble des identificateurs est potentiellement grand (il dépasse la centaine) et n'est pas nécessairement connu à l'avance. Au contraire, dans le portail d'accès aux applications authentifiées par mots de passe, l'ensemble des identificateurs est petit et peu souvent modifié.

Dans le premier cas, l'utilisateur sera invité à saisir les identificateurs pour l'application par le biais de zones de saisies comme dans la figure ci-dessous.

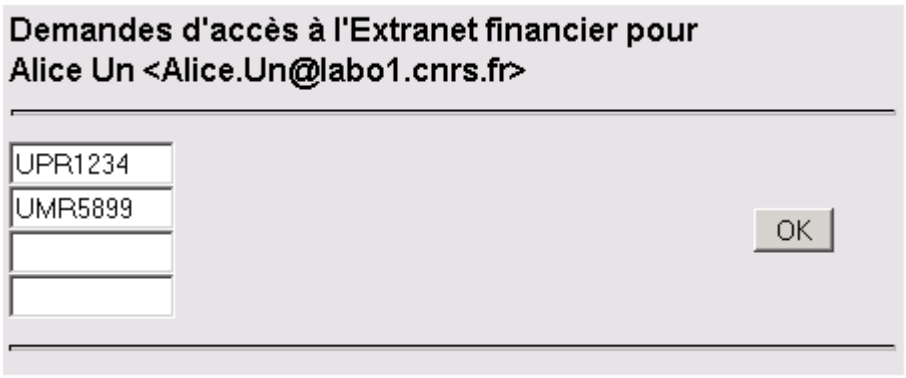

Figure 3 - *Demande d'accès avec zones de saisie*

Dans le second cas, l'utilisateur sera invité à saisir les identificateurs pour l'application par le biais de cases à cocher comme dans la figure ci-dessous.

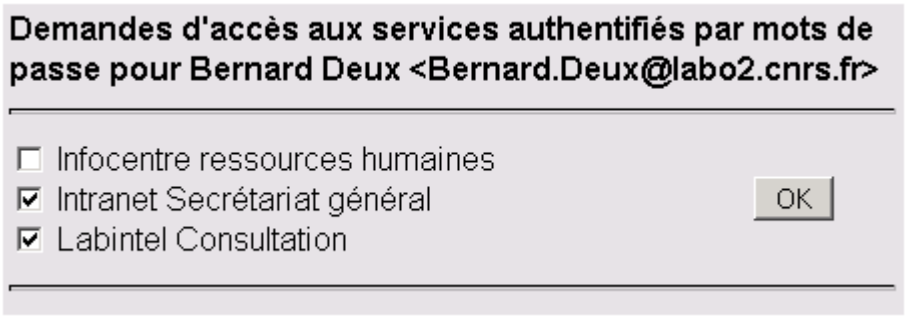

Figure 4 - *Demande d'accès avec cases à cocher*

Lorsque l'utilisateur soumet le formulaire, sa demande est rajoutée dans une table de la base de données ou dans une branche de l'annuaire contenant les demandes en attente. Un message électronique est envoyé à l'administrateur de l'application. Ce dernier utilise le formulaire permettant d'afficher les demandes en attente, de les valider, avec ou sans modification, ou bien de les rejeter. Lorsque l'utilisateur est lui même administrateur de l'application, le profil est immédiatement mis à jour.

L'interface pour l'administrateur d'une application contient deux formulaires : le premier permettant d'accéder aux requêtes en attente de validation, le second permettant de saisir directement le profil d'un utilisateur. En pratique, il s'agit du même formulaire que celui employé par l'utilisateur final.

L'interface pour l'administrateur du gestionnaire de profils contient en outre les formulaires pour créer, supprimer ou configurer des applications.

3.3 L'interface applicative

L'interface applicative consiste à implémenter la fonction « profil (*application*, *certificat*) → {*liste d'identificateurs*} ». Pour plus de commodité vis à vis des applications utilisant le gestionnaire de profils, cette fonction peut renvoyer davantage d'informations, notamment :

- un drapeau indiquant si l'utilisateur est administrateur de l'application
- les valeurs des divers attributs issus du certificat, comme le prénom, le nom, l'adresse électronique, etc.
- les noms et prénoms de l'état civil lorsque ces derniers diffèrent des noms et prénoms usuels contenus dans le certificat.

Il est souhaitable d'implémenter également la notion de cascades de profils. Pour reprendre l'exemple du système d'information régional, une personne qui est identifiée comme Directeur de laboratoire doit avoir automatiquement accès à l'Extranet financier, et d'une manière générale, à tous les services en ligne qui concernent son laboratoire.

Finalement le cœur du gestionnaire de profils est une fonction de n descripteurs d'applications et un descripteur de certificat d'utilisateur qui retourne le profil :

profil ($application_1$, …, $application_n$, *certificat*) → { *liste d'identificateurs*, *informations sur l'utilisateur*, *drapeaux* }

3.4 Ldapauth : une implémentation basée sur un annuaire LDAP

Ldapauth est une implémentation d'un gestionnaire de profils réalisée à la Délégation Aquitaine et Poitou-Charentes. Il est composé d'un ensemble de scripts et de fichiers d'inclusion PHP. Il s'appuie sur un annuaire OpenLdap. Il n'utilise que des objets LDAP standards.

La hiérarchie LDAP est similaire à celle décrite dans la Figure 2 ci-dessus. Les applications sont décrites par des objets LDAP de la classe `applicationProcess`. A ces objets sont rattachés :

1. objets de la classe `inetOrgPerson` : ils décrivent les profils de chaque utilisateur. Les attributs ont leur signification usuelle, notamment l'attribut `uid` dont les valeurs sont les identificateurs de l'utilisateur pour l'application et l'attribut `commonName` dont les valeurs sont ses différentes identités.

2. objets de la classe `groupOfNames` : ils décrivent les autorisations d'accès à l'application. Ils ont des noms distingués relatifs (RDN, *Relative Distinguished Name*) pré-définis qui permettent de paramétrer la liste des autorités de certifications acceptables, la liste des administrateurs, etc.

3. un objet de la classe `top` : il contient la liste des profils en attente de validation par l'un des administrateurs de l'application. En cas d'acceptation de la requête, le profil en attente vient remplacer le profil actuel. En cas de rejet de la requête, elle est simplement supprimée.

La figure ci-dessous décrit le sous-arbre correspondant à l'application « Extranet financier ».

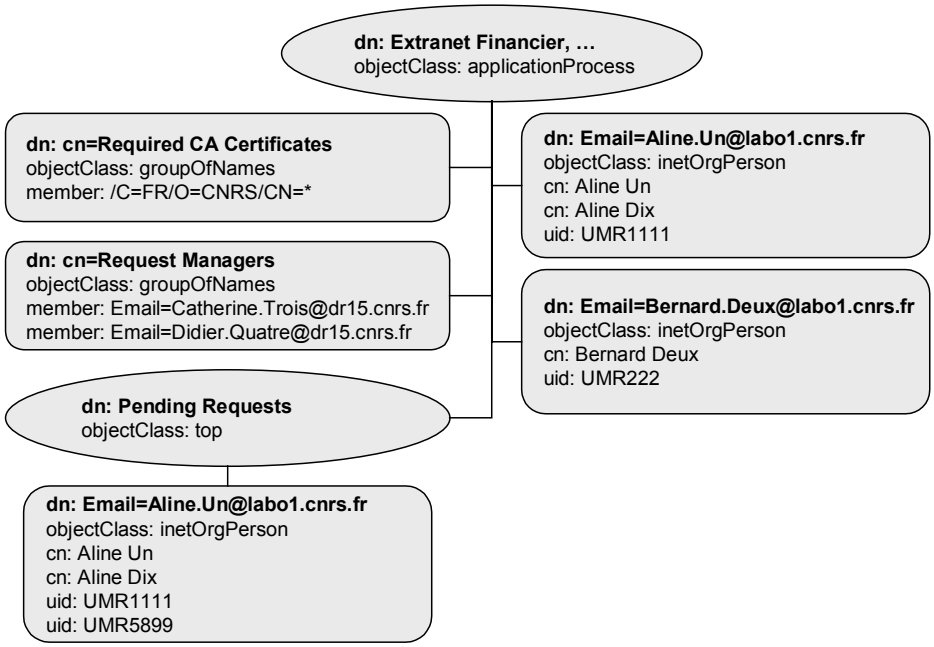

Figure 5 - *Exemple de l'arbre LDAP correspondant à l'Extranet financier*

Dans cet exemple, on trouve le profil de deux utilisateurs : d'une part, Aline Un, qui porte également comme nom Aline Dix et qui a comme identificateur UMR1111 et, d'autre part, Bernard Deux qui a comme identificateur UMR222. Une requête de modification du profil de Aline Un est en attente de validation. On trouve également la liste des adresses de messagerie des deux administrateurs de l'application (C. Trois et D. Quatre) ainsi qu'une contrainte sur les autorités de certifications obligatoires (*Required CA Certificates*) : l'application n'accepte que les certificats d'utilisateurs émis par une autorité dont le nom satisfait l'expression régulière indiquée. On trouvera en annexes la liste de toutes objets de la classe *groupOfNames* et leur signification.

Un ensemble de classes et de méthodes PHP existent pour manipuler les objets contenus dans ce type d'arborescence. Une interaction avec la messagerie électronique permet d'alerter les administrateurs de l'application dès lors qu'une demande de création ou de modification de profil a été émise par un utilisateur.

3.5 Avantages de Ldapauth

L'interface applicative très simple de Ldapauth a permis de faciliter considérablement l'écriture des applications utilisant l'authentification par certificat et a effectivement atteint l'objectif initial de fournir un mécanisme générique de vérification des permissions accordées à un utilisateur sur des applications de toute nature. Actuellement, les quatre applications citées en exemple au chapitre 2 utilisent Ldapauth.

Les interfaces WWW mises à disposition des administrateurs d'applications sont suffisamment intuitives pour être confiées à des utilisateurs proches du domaine fonctionnel de ces applications. Mis à part la création et le paramétrage initial d'une application, la gestion des profils s'effectue donc sans aucune intervention de l'administration système du site.

L'organisation du sous-arbre LDAP qui décrit une application est telle qu'une seule requête[1] permet de déterminer complètement les éléments qui constituent le profil d'un utilisateur : vérifier si les autorités qui ont émis son certificat sont acceptables, vérifier si l'utilisateur est administrateur de l'application, récupérer la liste de ses identificateurs. L'utilisation du gestionnaire de profil induit donc très peu d'*overhead* dans les applications.

Le choix de l'adresse de messagerie comme identificateur permet de prendre d'ores et déjà en compte l'existence d'IGC chez les partenaires institutionnels du CNRS. En effet, on peut imaginer que dans un futur très proche, les universités, écoles , établissements, etc. avec lesquels nous partageons des unités de recherche mettront elles aussi en place des IGC avec une politique de certification comparable à celle du CNRS. Les membres de ces unités disposeront alors, soit d'un certificat CNRS, soit d'un certificat de la co-tutelle, soit des deux. La configuration de leur profil dans Ldapauth est indépendante du certificat avec lequel ils accèdent à notre serveur.

3.6 Faiblesses de Ldapauth

L'utilisation de Ldapauth dans la configuration d'un serveur WWW comme Apache nécessiterait un développement spécifique. Certes, Apache 2.0 permet grâce à des modules comme *mod_auth_ldap* [6] de subordonner l'accès à des pages en fonction d'une requête LDAP : si la requête renvoie une ou plusieurs entrées, l'accès est autorisé, sinon il est interdit. Dans le cas de Ldapauth, le traitement postérieur appliqué au résultat de la requête LDAP n'est pas implémentable dans Apache. Ldapauth s'avère donc une solution mal adaptée au cas des pages statiques.

Par ailleurs, le module *mod_auth_ldap* ne prévoit pas l'enrichissement de l'environnement des scripts appelés. Il n'est donc pas possible de transmettre les éléments constitutifs du profil de l'utilisateur vers l'application. Cette dernière est donc contrainte de réémettre la requête LDAP, doublant ainsi les accès vers l'annuaire.

L'insertion de Ldapauth directement dans une application existante est triviale et, d'après ce qui précède, elle est aussi plus performante. Actuellement, seule l'interface PHP existe. Aucun développement n'est prévu pour le support d'applications écrites dans des langages populaires comme Python ou Perl.

4 Conclusion

Ldapauth a été conçu et développé en juillet 2002 à partir d'une première expérimentation fondée sur une base de données relationnelle Postgres. La configuration de cette base était effectuée « à la main » via des outils comme PgAccess ou

[1] `(|(objectClass=groupOfNames)(&(objectClass=inetOrgPerson)(mail=`*adresse_extraite_du_certificat*`)))`

PhpPgAdmin qui permettent de manipuler des tables avec une interface graphique. Les formulaires WWW de demande de création ou de modification de profil par les utilisateurs, la vérification puis la validation de ces demandes de façon instantanée ont permis de ramener à une valeur quasi-nulle la charge de gestion des profils pour des applications critiques comme l'Extranet Financier.

Toutefois, Ldapauth est une solution d'attente avant que se confirment des technologies plus standards comme l'identification unique (SSO, *Single Sign On*) ou les certificats d'attributs.

L'idée sous-jacente à SSO est séduisante : on récupère via un certificat électronique auprès d'un service d'identification un jeton qui peut être réutilisé pour s'authentifier auprès de n'importe quel service en ligne composant le système d'information.

Les certificats d'attributs [7] offrent eux aussi des perspectives intéressantes. Ils contiennent une référence à un certificat d'utilisateur et, sous la forme d'extensions spécifiques à chaque application, une description des permissions accordées à l'utilisateur. Ils doivent pouvoir être émis directement ou indirectement par les fournisseurs de services en fonction de leurs besoins.

Les grands chantiers décrits dans le Schéma directeur des système d'informations 2003-2006 du CNRS [8] prévoient implicitement la mise en œuvre de telles solutions. Elles permettraient d'homogénéiser à l'échelle de l'organisme les problèmes liés à la gestion des permissions, indépendamment de l'entité qui rend le service.

Le gestionnaire de profils comme celui décrit ci-dessus a des ambitions bien plus modestes. Mais il permet dès aujourd'hui d'appréhender les problèmes complexes liés à l'authentification et au contrôle d'accès dans un environnement fortement déconcentré et multi-tutelles.

Annexes

La table ci-dessous donne la liste des RDN (*Relative Distinguished Name*) prédéfinis pour l'administration de Ldapauth

Relative Distinguished Name **Classe d'objet**	**Description**
`ou=Ldapauth Managers` `objectClass: applicationProcess`	Application particulière permettant de définir les accès au gestionnaire de profil lui-même. Les utilisateurs qui ont le droit d'accéder à cette application peuvent effectuer toutes les opérations de création, modification et suppression d'applications. Ils sont considérés comme administrateurs du gestionnaire de profils. Le contrôle d'accès à cette application s'effectue de la même façon que pour toutes les autres. Ldapauth est donc un gestionnaire de profils qui peut être administré par lui-même.

La table ci-dessous donne la liste des RDN qui permet de paramétrer le contrôle d'accès pour une application.

Relative Distinguished Name **Classe d'objet**	**Description**

`Email=`*adresse_de_l'utilisateur*
`objectClass: inetOrgPerson`
	Profil de l'utilisateur désigné par son adresse de messagerie. Sauf indication contraire, l'utilisateur est autorisé à accéder à l'application avec l'un des identificateurs indiqués par les attributs `userid`.
	
	`dn: Email=Aline.Un@labo1.cnrs.fr, …`
	`objectClass: inetOrgPerson`
	`c: FR`
	`o: CNRS`
	`ou: UMR1111`
	`sn: Un`
	`givenName: Aline`
	`mail: Aline.Un@labo1.cnrs.fr`
	`cn: Aline Un`
	`cn: Aline Dix`
	`userid: UMR1111`
	`userid: UMR5899`

`cn=Required CA Certificates`
`objectClass: groupOfNames`
	Conditions nécessaires concernant les autorités de certifications ayant émis le certificat de l'utilisateur.
	
	Les membres de ce groupe sont des expressions régulières décrivant des DN d'autorités de certifications. Seuls les utilisateurs disposant de certificats émis par les autorités de certification vérifiant l'une ou l'autre de ces expressions régulières peuvent accéder à l'application.
	
	Exemple pour accepter uniquement les certificats émis par une autorité du CNRS :
	
	`dn: cn=Required CA Certificates, …`
	`objectClass: groupOfNames`
	`member: /C=FR/O=CNRS/CN=*`

`cn=Requires User Certificates`
`objectClass: groupOfNames`
	Conditions nécessaires concernant les certificats des utilisateurs.
	
	Les membres de ce groupe sont des expressions régulières décrivant des DN de certificats d'utilisateurs. Seuls les utilisateurs disposant de certificats vérifiant l'une ou l'autre de ces expressions régulières peuvent accéder à l'application.
	
	Exemple pour accepter uniquement les certificats des utilisateurs du laboratoire UMR1111 et UMR222 :
	
	`dn: cn=Required User Certificates`
	`objectClass: groupOfNames`
	`member: /C=FR/O=CNRS/OU=UMR1111/CN=*/Email=*`
	`member: /C=FR/O=CNRS/OU=UMR222/CN=*/Email=*`

`cn=Allowed CA Certificates` `objectClass: groupOfNames`	Conditions suffisantes concernant les autorités de certifications ayant émis le certificat de l'utilisateur. Les membres de ce groupe sont des expressions régulières décrivant des DN d'autorités de certifications. Les utilisateurs disposant de certificats émis par les autorités de certification vérifiant l'une ou l'autre de ces expressions régulières peuvent accéder à l'application. Ils accèdent toutefois à l'application avec une liste d'identificateurs vide. Cette fonctionnalité est utilisée pour créer des Intranets/Extranets sur la base d'autorités de certification. La syntaxe est la même que pour « `cn=Required CA Certificates` »
`cn=Allowed User Certificates` `objectClass: groupOfNames`	Conditions suffisantes concernant le certificat de l'utilisateur. Les membres de ce groupe sont des expressions régulières décrivant des DN de certificats d'utilisateurs. Les utilisateurs disposant de certificats émis par les autorités de certification vérifiant l'une ou l'autre de ces expressions régulières peuvent accéder à l'application. Ils accèdent toutefois à l'application avec une liste d'identificateurs vide. Cette fonctionnalité est utilisée pour créer des Intranets/Extranets sur la base de caractéristiques contenues dans les certificats d'utilisateurs. La syntaxe est la même que pour « `cn=Required User Certificates` »

`cn=Request Managers`
`objectClass: groupOfNames`
	Liste des administrateurs de l'application.
	
	Les membres de ce groupe sont les adresses électroniques des administrateurs de l'application. Par exemple :
	
	`dn: cn=Request Managers`
	`objectClass: groupOfNames`
	`member: Email=user1@labo.cnrs.fr`
	`member: Email=user2@labo.cnrs.fr`

`cn=List of Userids`
`objectClass: groupOfNames`
	Liste exhaustive des identificateurs possibles pour cette application.
	
	Les membres de ce groupe sont de la forme « `label`=*description de l'identificateur*, `value`=*identificateur* ». L'attribut `label` permet de décrire sommairement la signification de l'identificateur. L'attribut `value` donne l'identificateur lui-même.
	
	L'exemple ci-dessous indique comment est déclarée la liste des identificateurs dans la Figure 4 ci-dessus.
	
	`dn: cn=List of Userids`
	`objectClass: groupOfNames`
	`member: label=Infocentre ressources humaines, value=inforh`
	`member: label=Intranet Secrétarial général, value=intrasg`
	`member: label=Labintel Consultation, value=labcons`

`ou=Pending Requests` `objectClass: top`	Liste des demandes de modification de profils en attente de validation. Les objets rattachés à cette entrée de l'arbre sont des profils qui ont été constitué à partir du formulaire de demande de création ou de modification du profil de l'utilisateur.

Références

[1] T. Berners-Lee, R. Fielding et H. Frystyk, Hypertext Transfer Protocol – HTTP/1.0, RFC 1945, Mai 1996.

[2] ISO/IEC 9594-8 / ITU-T Recommandation X.509 v3 : Information technology : Open systems interconnection – The Directory : Authentication framework, Juillet 1996.

[3] A. Frier, P. Karlton and P. Kocher, The SSL 3.0 Protocol. *http://wp.netscape.com/eng/ssl3/ssl-toc.html*, Mars 1996.

[4] T. Dierks et C. Allen, The TLS Protocol Version 1.0, RFC 2246, Janvier 1999.

[5] W. Yeong, T. Howes et S. Kille, Lightweight Directory Access Protocol, Mars 1995.

[6] Apache HTTP Server Documentation Project, *http://httpd.apache..org/docs-2.0/mode/mod_auth_ldap.html*

[7] S. Farell et R. Housley, An Internet Attribute Certificate Profile for Authorization, RFC 3281, Avril 2002.

[8] CNRS, Direction des systèmes d'information, Schéma directeur des systèmes d'information 2003-2006, *http://www.dsi.cnrs.fr/SchemaDirecteur/SchemaDirecteurDSAI-04-03.pdf*, Avril 2003.