Mise en oeuvre d'un intranet à partir de logiciels Open Source avec intégration des certificats numériques et login unique

Xavier Jeannin
Unité Réseaux du CNRS
Université P. & M. Curie
4, place Jussieu - 75252 PARIS CEDEX 05
mailto:xavier.jeannin@urec.cnrs.fr
Date: 8 Octobre 2003

Résumé

Le projet d'intranet pour le département scientifique des Sciences et Technologies de l'Information et de la Communication (STIC) visait à mettre en place un intranet sécurisé de personnes réparties dans toute la communauté scientifique (CNRS, Faculté, etc.) et accessible depuis l'Internet. Une première réponse à ce problème a été l'utilisation des certificats numériques de l'infrastructure de Gestions de Clés du CNRS, mais lors de la mise en place des applications collaboratives, nous avons été confrontés à la multiplication des mots de passe pour les utilisateurs.
De plus, le besoin du département STIC s'est orienté vers la possibilité de gérer plusieurs groupes de travail. Nous avons donc développé un logiciel A2C2 (Accès aux Applications Contrôlés par Certificat) permettant d'administrer des groupes de travail indépendants et fournissant à chaque groupe de travail la possibilité de publier des pages HTML avec accès contrôlé et partage évolué de fichiers. De plus, A2C2 permet d'intégrer d'autres applications en utilisant uniquement les certificats comme moyen de connexion.

A2C2 a été conçu pour les laboratoires du CNRS, pour cela il a été écrit avec des logiciels libres populaires dans cette communauté à savoir Apache, PHP, MySQL.

Mots clefs

Web, HTTP, certificat numérique, contrôle d'accès, Single Sign-On, outils coopératifs, groupe de travail

1 Objectif du projet

L'objectif de ce projet est de créer un intranet pour la direction du département scientifique des Sciences et Technologies de l'Information et de la Communication (STIC). La direction du département souhaite, grâce à cet intranet, développer les échanges avec ses laboratoires et les échanges entre laboratoires (par exemple les Réseaux Thématiques Pluridisciplinaires RTP). L'intranet doit être capable d'accueillir des groupes de travail dont les membres sont disséminés dans différents laboratoires. Discriminer les utilisateurs par leurs adresses IP paraît, dès lors, impossible, il faut bâtir un intranet sécurisé de personnes. Les certificats permettent de gérer finement l'accès à l'intranet, de réaliser une authentification forte et la mise en place d'un chiffrement des données échangées.

Sur un plan fonctionnel, l'intranet doit pouvoir supporter plusieurs groupes de travail indépendants les uns des autres et permettre l'intégration de logiciels notamment les logiciels coopératifs (Groupware). La gestion des groupes de travail peut être lourde et difficile pour un administrateur de site car il ne connaît pas forcément ni les membres des groupes de travail ni leurs droits ; nous avons voulu déléguer cette partie à l'animateur du groupe de travail. Pour ce faire, nous avons défini le rôle de super-administrateur pour l'administrateur de site, d'administrateur de groupe pour l'animateur du groupe et de simple utilisateur pour les membres du groupe. Le super-administrateur est ainsi soulagé des tâches de gestion des utilisateurs du groupe et de la gestion de leurs droits d'accès aux données et aux applications du groupe. Le super-administrateur et l'administrateur de groupe ont besoin d'un outil pratique pour gérer les droits d'accès. Chaque groupe pourra accéder à un logiciel de publication de page HTML, un logiciel de partage de données, une liste de diffusion. A ce cahier des charges nous avons ajouté la possibilité d'intégrer d'autres logiciels et ceci en utilisant exclusivement les certificats comme unique moyen de validation de connexion. Sur un plan technique, l'objectif est de réaliser un accès transparent, sans recourir à plusieurs mots de passe pour chacune des applications déployées dans l'intranet. Le logiciel A2C2 (Accès aux Applications Contrôlés par Certificat) est donc une plate-forme fournissant des fonctionnalités de base pour un groupe de travail qui peuvent être enrichies par l'ajout d'autres applications.

Afin de pouvoir réutiliser cette application qui nous apparaissait relativement généraliste, nous avons choisi des constituants logiciel répandus dans les laboratoire et du domaine des logiciels Open Source : Linux, OpenSSL, Apache, ModSSL, Moteur Zend PHP, ModPerl, CGI-BIN, les bases de données MySQL, enfin les applications Open Source dites « web ».

2 Intranet tentative de définition

Les notions d'origine commerciale d'intranet et d'extranet sont des concepts mal définis. Bien qu'il existe plusieurs définitions, on admet communément les définitions suivantes :

- Les communications dans l'entreprise : intranet
- Les communications avec les partenaires et clients : extranet
- Les communications avec tout le monde : Site Web externe

La caractéristique de l'intranet et de l'extranet réside dans le fait que l'on veut pouvoir limiter l'accès à ce site Web à des individus déterminés. Techniquement, la discrimination des personnes peut se faire « géographiquement » en s'appuyant sur les numéros IP utilisés au sein de l'entreprise. Le périmètre du réseau de l'entreprise peut être difficile à définir. Au CNRS, par exemple, la limite commence-t-elle à la sortie du réseau du laboratoire ou à la sortie du campus ? Cette sélection par numéro IP comporte des limitations et entraîne des difficultés de gestion ; les utilisateurs cherchent à construire des « intranets de personnes » qui discriminent l'accès en fonction d'une personne ou de la fonction occupée par une personne. Pour se faire, on peut gérer des comptes d'utilisateur et des mots de passe ou utiliser des certificats numériques.

La différence entre extranet et intranet est très ténue, par exemple pour un laboratoire du CNRS, devons-nous considérer les autres laboratoires comme faisant partie de l'extranet ou de l'intranet ? En pratique, les sites ont besoin de pouvoir définir des niveaux d'accès aux ressources de leur site Web en fonction des entités et des personnes avec lesquelles ils travaillent. Ce niveau d'accès est déterminé en fonction du type de coopération et du niveau confiance.

L'accès au site recouvre à la fois l'accès à des informations (pages HTML), l'accès aux données brutes (fichiers) mais aussi l'accès aux applications. Quelques applications sont accessibles à tous (forum) mais la grande majorité est du domaine de l'intranet par nature. Les applications Web peuvent être des applications de travail partagé (Groupware : par exemple un calendrier partagé, partage de fichiers, etc.), des accès à des services réseaux (par exemple Webmail), des applications métiers (propre à la société). Ces dernières peuvent être portées sur l'intranet et l'accès au système d'informations se fait via le Web. L'intranet devient donc un secteur sensible d'autant plus s'il est accessible depuis l'extérieur du réseau de l'entreprise.

Les intranets sont centrés sur des applications de travail partagé (Groupware), le but de ces applications est d'augmenter la coopération au sein de l'entreprise et de pérenniser des compétences (Knowledge Magnagement). Les intranets se différencient d'entreprise en entreprise lorsque l'on introduit des applications métier qui ont été interfacées pour fonctionner avec un serveur Web. Dans ce cas, le contenu des intranets est très différent selon la nature de l'entreprise (banque, constructeur automobile, laboratoire du CNRS ou de la faculté). Les caractéristiques exigées (niveaux de sécurité, taille, criticité, etc.) peuvent aussi être très variables. Les performances (rapidité, fiabilité) et l'ergonomie particulière des applications Web restent un frein au passage de ces applications sur les intranets.

L'intranet peut être réparti sur plusieurs serveurs et sur plusieurs sites en fonction des besoins, de la charge ou de l'organisation de l'entreprise, ce qui accentue le besoin d'une authentification forte voire unique.

3 Contrôle dynamique d'accès par certificat aux ressources de l'intranet

Notre projet vise à créer un intranet de personnes s'appuyant sur une authentification forte basée sur les certificats numériques. L'usage des certificats permet d'authentifier les personnes connectées et de leur assurer une connexion sûre quel que soit le lieu géographique depuis lequel elles font leur accès.

Rappel sur les certificats [1][2] : un certificat est l'équivalent d'une carte d'identité ou d'un passeport, il contient l'équivalent des mêmes informations. Le format reconnu actuellement est le format X509V3 [3]. C'est un petit fichier, généré par une autorité de certification, qui contient au moins les informations suivantes :

- Le nom de l'autorité de certification qui a créé le certificat
- Le nom et le prénom de la personne
- Son organisation (CNRS par exemple)
- Son service (au CNRS, le nom du laboratoire)
- Son adresse électronique
- Sa clé publique
- Les dates limites de validité du certificat
- Signature électronique

Le certificat contient notamment un champ « Subject » qui est le Distinguished Name (DN) du propriétaire du certificat. Nous utilisons ce champ pour authentifier la personne qui se connecte à l'intranet.

Nous distinguons trois types de ressources : les pages affichables dans un navigateur (pages HTML), les applications Web et les fichiers de données brutes (fichiers texte, fichiers PDF, etc.). L'accès aux pages HTML et l'accès aux applications peuvent se traiter par le contrôle qu'Apache fait sur les répertoires du site Web. Pour les fichiers de données et les répertoires contenant ces fichiers de données, nous gérons les droits au niveau de l'application de partage de données et non au niveau d'Apache ce qui nous permet de définir des propriétaires, des groupes, des droits en lecture et écriture.

3.1 Contrôle d'accès et gestion des pages HTML

Il existe plusieurs méthodes, maintenant bien connues, pour paramétrer l'accès à un répertoire sur un serveur Apache [4]. Comme nous voulions que ce paramétrage soit dynamique, nous nous sommes appuyés sur des fichiers « .htaccess » placés dans les répertoires du serveur Web. Ces fichiers contiennent une référence à des fichiers de pseudo mot de passe qui contiennent le « Distinguished Name » du certificat de la personne [4]. La création et la modification de ces deux types de fichiers sont prises en compte tout de suite par Apache sans avoir besoin de le relancer. Ceci nous permet de modifier les droits d'accès dynamiquement. C'est ainsi qu'est géré le droit d'accès au page HTML et aux applications.

Configuration d'Apache :

```
<Directory "/usr/local/apache/htdocs/MonIntranet">
    AllowOverride AuthConfig
    SSLVerifyClient require
    SSLVerifyDepth 5
    SSLRequireSSL
    SSLRequire %{SSL_CLIENT_I_DN_CN} eq "CNRS-Standard"
    SSLOptions +FakeBasicAuth
</Directory>
```

Exemple de fichier ".htaccess"généré par A2C2 :

```
AuthUserFile /home/apache/MonIntranet/password.400.6e201560e8c7c7a83e966fdda2f3b1fd
AuthName "Authentification needed ..."
AuthType Basic
require valid-user
```

Exemple de pseudo fichier de mot de passe password.400.6e201560e8c7c7a83e966fdda2f3b1fd généré par A2C2 :

```
/C=FR/O=CNRS/OU=UPS836/CN=Anne Atol/Email=aa@urec.cnrs.fr:xxj31ZMTZzkVA
/C=FR/O=CNRS/OU=UPS836/CN=Juliette Romeo/Email=jr@urec.cnrs.fr:xxj31ZMTZzkVA
/C=FR/O=CNRS/OU=UPS836/CN=Pierre  Meuliere/Email=pm@urec.cnrs.fr:xxj31ZMTZzkVA
```

Le logiciel A2C2 offre la possibilité à l'administrateur de groupe de créer et de gèrer les fichiers ".htaccess" et les fichiers de pseudo mot de passe en fonction des commandes réalisées dans l'interface. Les administrateurs de site ont pointé la lourdeur de la gestion à la main de ces fichiers, l'automatisation de cette tâche permet une gestion aisée des droits d'accès à un site Web. De plus, le logiciel A2C2 permet à un administrateur de groupe de gérer (créer, déposer, détruire) les pages HTML. Ces fonctionnalités sont réalisées par le module d'Application de Gestion de Fichier (AGF) de A2C2.

3.2 Le contrôle d'accès aux applications

L'accès aux applications est contrôlé de deux manières. Tout d'abord, le répertoire où se situe l'application est protégé par un fichier « .htaccess ». L'administrateur de groupe peut régler par ce moyen les droits d'accès des utilisateurs de son groupe à une application. Enfin si l'application gère un compte et un mot de passe, il faut faire la correspondance entre l'utilisateur et le compte de l'utilisateur dans l'application. L'administrateur de groupe doit créer les comptes au sein de cette application pour chaque utilisateur. S'il n'existe pas d'API pour cette application, cette tâche n'est pas automatisable sans modifier l'application. La création à la main des comptes dans l'application peut être fastidieuse mais l'intégration plus avant de l'application au sein de A2C2 pose le problème de la maintenance future de cette application modifiée.
Dans l'application, l'installation d'une application pour un groupe de travail est réalisée par le super-administrateur.

3.3 Contrôle d'accès et gestion des fichiers de données.

A partir du logiciel de partage de données phpCommander, nous avons créé le module Application de Gestion de Fichier (AGF). Ce module est à même de gérer le propriétaire, les groupes d'un répertoire ou d'un fichier ainsi que les droits

d'écriture sur un fichier et les droits de dépôt dans un répertoire. Les données sont conservées en dehors de l'arborescence du serveur Web pour des raisons de sécurité. Le module AGF est utilisé dans deux cas pour le dépôt et la gestion des pages HTML et pour le dépôt, la gestion et l'accès aux fichiers partagés.

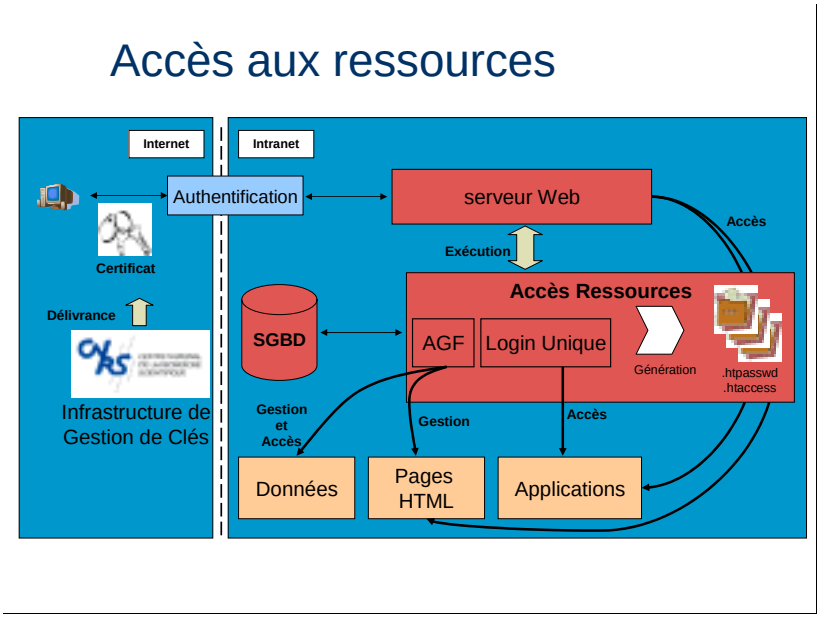

Figure 1 schéma du contrôle d'accès aux ressources de l'intranet

4 Certificats numériques et login unique

L'authentification forte par certificat permet de sécuriser et de discriminer finement les accès ; de plus l'usage du certificat offre un avantage plus palpable pour les utilisateurs : l'accès direct à leurs ressources sur le réseau notamment l'accès aux applications se fait sans utiliser un nom de compte utilisateur et un mot de passe.

4.1 Le Web Single Sign-On (SSO)

Cette technique d'usage unique d'un nom de compte utilisateur et de mot de passe unique est connue sous le nom de Single Sign-On. Le principe est de s'authentifier une fois et de pouvoir accéder à toutes ses applications (courriel, agenda partagé, compte Unix). Le type d'authentification est libre : certificat numérique, nom de compte et mot de passe ou autres. Le domaine du SSO se divise en un SSO poste et un SSO Web. Le SSO poste touche toutes les applications traditionnelles, le SSO Web concernent les applications Web. Les mêmes techniques peuvent être employées dans les deux types de SSO.

Voici un exemple des technologies employées dans le Web SSO :

- Serveur d'authentification et agent Web : l'agent Web intercepte la requête et interroge le serveur d'authentification. Les agents Web sont en pratique des modules du serveur Web
- Redirection d'URL : par exemple le navigateur client peut être redirigé vers un serveur d'authentification.
- Mandataire (proxy) : pour accéder au site, il est nécessaire de passer par la passerelle proxy qui se charge de la connexion au site ou à l'application à la place du client
- Le « postage » d'URL : la connexion à une application se traduit par le remplissage d'un formulaire par un utilisateur. Ces formulaires sont souvent de type POST (ils peuvent être occasionnellement de type GET), il est alors possible d'envoyer une requête HTTP de ce type contenant les paramètres de connexion (mot de passe etc.) avec d'éventuels cookies afin de réaliser la connexion à l'application Web
- Security Assertion Markup Langage (SAML) [5]: SAML est un protocole normalisé par l'OASIS, basé sur XML et qui vise à permettre l'échange entre serveurs d'informations d'authentification et d'accréditation

4.2 Principe de l'intégration des applications par certificat dans A2C2

Notre but est d'offrir à l'utilisateur la possibilité d'utiliser son certificat comme moyen unique de connexion à l'intranet et aux applications, notre projet est une application « modeste » de la problématique générale du SSO Web. L'authentification est faite par les certificats numériques. Les autorisations d'accès sont contrôlées grâce à la configuration d'Apache et à notre

base de données locale. Un principe fondamental de développement de l'application A2C2 est de n'avoir pas à modifier les applications Web qui seront intégrées à l'intranet. En effet, nous estimions que le travail de maintenance qui en découlerait, serait trop lourd ; c'est pourquoi nous nous sommes imposés à trouver une solution qui ne nécessite pas de modifier les applications. L'intégration des applications se fait par la technique simple du postage des paramètres de connexion. L'intranet permet également de se connecter à des applications installées sur un autre serveur, les limitations dans ce cas sont importantes. Notre projet correspond donc plus à un SSO local à l'intranet.

L'absence de modification des applications permet d'intégrer un grand nombre d'applications et le travail de maintenance de l'application cible est simplifié. Pour intégrer une application, il est nécessaire d'écrire un programme qui sert d'interface pour la connexion à chaque application. Ce programme de connexion que nous appelons « logcertif.php », est simple. Dans l'application A2C2, les arguments à passer à l'application et utilisés par logcertif.php, sont sauvegardés dans la base de données lors le l'intégration de cette application au groupe de travail. Le script logcertif.php identifie la personne grâce à son certificat, puis il récupère les paramètres de la personne (mot de passe) et de l'application dans la base de données et enfin il réalise la connexion pour la personne à votre application. Pour intégrer votre application, vous n'avez qu'à recopier le logcertif.php fourni avec A2C2 dans le répertoire de l'application. Certaines applications nécessitent des modifications de logcertif.php, par exemple dans le cas de redirection HTTP, l'adaptation est alors d'autant plus simple que les principes qui régissent les applications Web (les cookies, les sessions, formulaire) sont connus.

Le principe général est de « poster » (envoyer une requête POST) qui remplit le formulaire de connexion à l'application. Cette méthode est largement utilisée dans les logiciels SSO du marché. Ces paramètres sont liés au compte de l'utilisateur mais aussi à l'application même (par exemple le type de base de données utilisée par l'application). Cette méthode fonctionne facilement sans chiffrement (sans utilisation HTTPS) , mais si la liaison entre le client (le navigateur) et le serveur Web est chiffrée (HTTPS) comme dans notre cas, il est impossible de s'immiscer dans la session SSL. Dès lors, la solution consiste à initier une nouvelle connexion HTTPS entre le serveur et l'application sur ce même serveur. Il faut utiliser un nouveau certificat pour établir la connexion, nous nommons ce certificat « un certificat de service ». Ce certificat de service est local au serveur, sous le contrôle de l'administrateur et doit permettre l'accès à toutes les applications. Cette seconde connexion sert à remplir le formulaire HTML de connexion à l'application. Une fois connecté à l'application, cette dernière renvoie les données de session (cookie) et le résultat de la connexion à logcertif.php. Enfin, le programme logcertif.php va retransmettre toutes ces données au navigateur du client. Dès lors, le client est connecté à l'application sans avoir fourni son nom de compte et son login.

Les étapes de connexion :

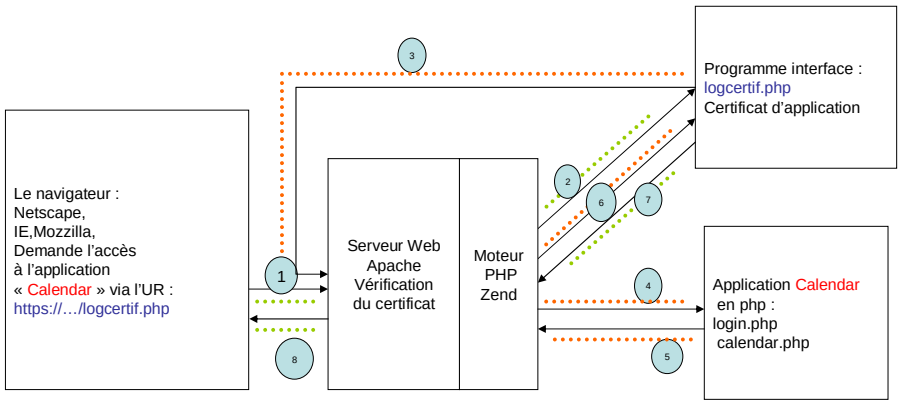

Figure 2 Schéma de la connexion à une application Web en utilisant le certificat comme login

1. Demande de connexion à l'application (l'url https://monsite.edu/monappli/logcertif.php)
2. Connexion au programme d'interface (1ére session SSL)
   Vérification des droits d'accès
   Obtention des paramètres de connexion : login/password, autres
   Création d'une requête vers l'URL de connexion à l'application : https://monsite.edu/monappli/login.php
3. Envoi de la requête sur login.php au serveur avec certificat de service pour passer le contrôle du serveur (début de la seconde session SSL)
4. login.php s'exécute comme si les paramètres du formulaire étaient déjà remplis
5. Retour de login : page HTML + COOKIE (fin de la seconde session SSL)
6. Le programme interface logcertif.php obtient la page HTML + cookie, il peut modifier cette page HTML etc…
7. Renvoi de la page HTML + cookie après traitement
8. La page HTML + cookie arrive au navigateur, l'application continue normalement

Dans l'intranet, il n'y a pas un nom de compte et un mot de passe unique pour toutes applications auxquelles un utilisateur a accès. Au contraire, l'application A2C2 génère un nom de compte et un mot de passe par utilisateur pour chaque application.

4.3 Un exemple d'intégration d'une application à l'intranet

Voici un exemple plus concret pour expliquer la méthode d'intégration de l'application WebCalendar dans l'intranet. Pour commencer il faut identifier les paramètres de connexion suivants :

- URL de connexion (emplacement de l'application)
- Paramètres pour remplir le formulaire.

Tous ces paramètres sont lisibles dans le code source de la page HTML de connexion à l'application. (cf la balise FORM sans oublier les paramètres cachés).

Nous devons rechercher dans le code de la page de connexion (login.php) à l'application, les paramètres que ce formulaire envoie lorsque l'on « clique » sur le bouton « connexion ». Un exemple de formulaire de connexion à l'application WebCalendar :

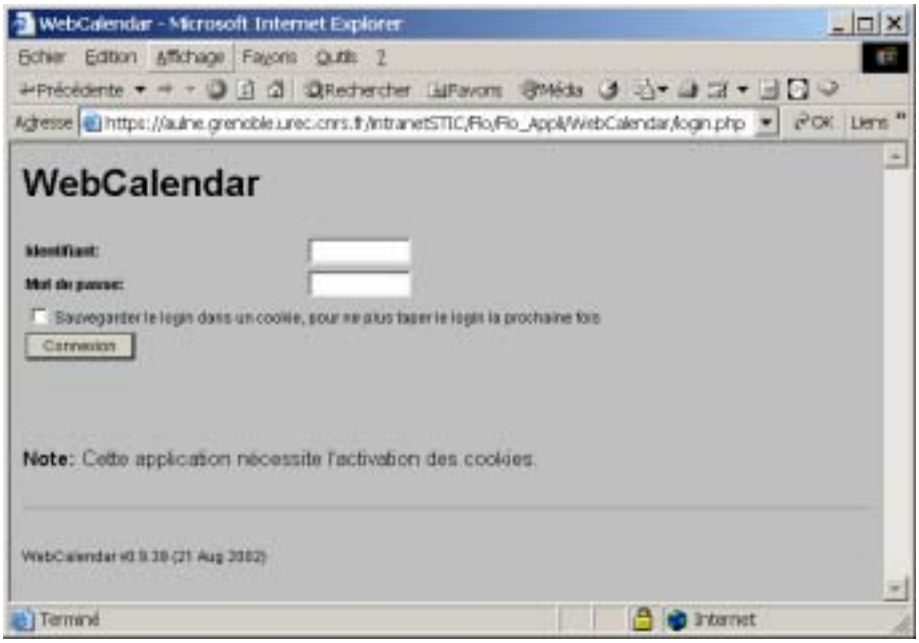

Figure 3 Formulaire de connexion à l'application WebCalendar

Il faut repérer dans le code de cette page HTML la partie entre les balises <FORM> et </FORM> :

```
<FORM NAME="login_form" ACTION="login.php" METHOD="POST" ONSUBMIT="return
valid_form(this)">

<TABLE BORDER=0>
<TR><TD><B>Identifiant:</B></TD>
  <TD><INPUT NAME="login" SIZE=10 VALUE="" TABINDEX="1"></TD></TR>
<TR><TD><B>Mot de passe:</B></TD>
  <TD><INPUT NAME="password" TYPE="password" SIZE=10 TABINDEX="2"></TD></TR>
<TR><TD COLSPAN=2><INPUT TYPE="checkbox" NAME="remember" VALUE="yes" > Sauvegarder le
login dans un cookie, pour ne plus taper le login la prochaine fois</TD></TR>
<TR><TD COLSPAN=2><INPUT TYPE="submit" VALUE="Connexion" TABINDEX="3"></TD></TR>
</TABLE>

</FORM>
```

Les paramètres à récupérer afin modifier le script logcertif.php sont marqués en bleu ou en grisé. On recherche l'URL du champs ACTION de la balise <FORM>, c'est l'URL de connexion (dans notre exemple : `"login.php"`). Puis, on s'intéresse à toutes les balises <INPUT> à l'exception de celle du type Submit.

Nous possédons donc tous les éléments pour modifier logcertif.php. Dans A2C2, tous ces paramètres sont sauvegardés dans la base de données de l'intranet. Le programme logcertif.php récupère l'URL de connexion et les paramètres de l'utilisateur (nom de compte et mot de passe). Cette opération automatise la connexion de l'application ; dans la pratique, il est parfois nécessaire d'adapter le logcertif.php à votre application par exemple si la connexion à l'application réalise des redirections HTTP.

```
<?php
// Début du programme est le même pour toutes les applications
// identifier l'utilisateur par rapport à son certificat

//on récupère les valeurs par défaut des paramètres et on construit
// la chaine de parametres pour la requete POST
$finChaineParamPost  = "";
$resParam = $loginUnique->LOGINUNIQUEgetDefaultParam($mysql, $_REQUEST['idApplication']);
while ($upletParam = mysql_fetch_array($resParam))
{
      $nomParam = $upletParam['Nom'];
      $valParam = $upletParam['Valeur'];
      $finChaineParamPost  .="&".$nomParam."=".$valParam;
}

//url de connexion à l'application
$pageLogin     = $applicationGT->APPLICATIONGTgetURLLogin($mysql, $_REQUEST['idAppliGT']);
//informations sur les certificats pour la 2ime session SSL
$certificatPEM = $intranet->INTRANETgetConfigIntranet($_SESSION['idIntranet'], '', '',
'CERTIFICAT_SSO', $mysql);
$certificatPW  = $intranet->INTRANETgetConfigIntranet($_SESSION['idIntranet'], '', '',
'CERTIFICAT_PASSWD', $mysql);
$certificatCA  = $intranet->INTRANETgetConfigIntranet($_SESSION['idIntranet'], '', '',
'CERTIFICAT_CAINFO', $mysql);

# preparation de l'envoi du formulaire, le cookie recu est stocke sur le serveur dans le
fichier "/tmp/cookie".$_REQUEST['idCompte']
# et la page web est recu en resultat de curl_exec
$ch = curl_init ($pageLogin);
curl_setopt($ch, CURLOPT_SSLCERT, $certificatPEM);
curl_setopt($ch, CURLOPT_SSLCERTPASSWD, $certificatPW);
curl_setopt($ch, CURLOPT_CAINFO , $certificatCA);

//recupération des login, password à partir de l'identifiant du compte
$infoCompte  = $gestionCompteApplication->GESTIONCOMPTEAPPLICATIONsearchOneCompte($mysql,
$_REQUEST['idCompte']);
$upletCompte = mysql_fetch_array($infoCompte);
$valLogin    = $upletCompte['Login'];
```

```
$valPassword = $upletCompte['Password'];

$chaineParamPost = "login=".$valLogin."&password=".$valPassword.$finChaineParamPost;
curl_setopt($ch, CURLOPT_POSTFIELDS, $ChaineParamPost);
$cookie = "/tmp/cookie".$_REQUEST['idCompte'];
curl_setopt($ch, CURLOPT_COOKIEJAR, $cookie);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);

# envoi du formulaire et recuperation du cookie et de la page HTML
ob_start();
$postResult = curl_exec ($ch);
ob_end_clean();
curl_close ($ch);

// Fin du programme
// Envoyer la page HTML et les cookies vers le navigateur client
?>
```

Les caractères en bleu ou en grisé représente les paramètres passés à l'application et extrait de la base de données de l'intranet.

4.4 Type d'applications intégrables et limitations

Les applications Web intégrables sont les applications CGI-BIN, Fast-CGI, Mod_Perl et PHP. Théoriquement, l'intégration d'application java ne doit pas poser de problème technique particulier mais cette intégration n'est pas pour l'instant prévue dans l'application.

Certaines applications associent à la session le numéro IP du client, or du point de vue de l'application c'est le serveur de l'intranet et non le client qui réalise la connexion ce qui entraîne que le client ne peut récupérer la session. Dans ce cas, vous pouvez modifier le code de l'application et passer un paramètre supplémentaire grâce à logcertif.php qui positionne correctement le numéro IP de la machine cliente. Enfin, les applications, qui utilisent des cookies comprenant l'adresse IP du client de manière chiffrée (cookies « non rejouables »), ne sont pas intégrables par A2C2.

Vous pouvez interconnecter une application qui se trouve hors de votre intranet ; ici nous avons deux cas : soit l'application cible reconnaît les certificats (par exemple Sympa), soit l'application ne reconnaît pas les certificats. Si l'application reconnaît les certificats, il suffit de faire un lien HTML vers cette autre application. Si cette application ne les reconnaît pas, vous utilisez la même méthode qu'en local en changeant seulement l'URL. Vous pouvez ainsi réaliser une connexion à une autre application Web située sur un autre site et donc permettre à vos utilisateurs de se connecter à leurs applications Web favorites sans mot de passe via leur certificat et votre intranet. Il existe une limitation importante à ce dernier point, si le site de l'application se trouve dans un domaine différent du vôtre et que l'application utilise des cookies, il vous sera impossible de retransmettre ces cookies au navigateur de votre utilisateur. En effet pour des raisons de sécurité, on ne peut renvoyer un cookie d'un domaine extérieur au vôtre. Par exemple, si votre intranet se trouve dans le domaine urec.cnrs.fr, vous pouvez renvoyer un cookie pour les domaines urec.cnrs.fr ou cnrs.fr mais pas pour cru.fr, ou gouv.fr [6]. Si la connexion avec des applications distantes est possible, le but premier d'A2C2 est de permettre un accès aux applications locales à l'intranet.

5 L'application « A2C2 »

L'objectif de l'intranet est de pouvoir fournir un outil de communication aux différents groupes de travail qui opèrent au sein du département STIC. L'application que nous avons développée vise à faciliter la création et la gestion de groupes de travail. Les personnes qui participent à l'intranet s'authentifient par certificat.

Chaque groupe de travail comprend automatiquement un espace de publication HTML, un espace de partage de fichiers et un logiciel pour gérer ces deux espaces. On peut incorporer des applications Web supplémentaires pour un groupe de travail. Les droits sur l'accès aux répertoires de publication de pages HTML d'un groupe de travail peuvent être attribués pour le groupe de travail ou pour des sous-groupes. Les droits sur les fichiers de données de l'espace d'échange du groupe de travail peuvent être positionnés en fonction du groupe d'affectation en lecture écriture pour les fichiers et en droit de d'entrée et droit de dépôt pour les répertoires.

Dans le cadre de l'intranet du département STIC, des listes de diffusion ont été mises à la disposition des groupes de travail sur un serveur annexe de listes de diffusion SYMPA [7], l'administration de ces listes se faisant grâce aux mêmes certificats

utilisés pour accéder à l'intranet. Enfin, nous avons intégré un calendrier partagé et un accès direct à l'application externe Labintel (une application de gestion du CNRS) pour le groupe de travail « Direction du STIC ».

L'administration de chaque groupe est déléguée à un membre du groupe. Ce dernier peut ainsi gérer les membres de son groupe, créer des sous-groupes, régler les droits sur les répertoires de publication Web, régler les droits sur les fichiers mis en partage.

5.1 Les acteurs

Il existe trois types d'acteurs concernés dans l'application :

- « Utilisateur Courant » : c'est une personne autorisée seulement à accéder à l'Intranet, elles accède aux groupes de travail auxquels elle participe. Pour chaque groupe, elle peut utiliser l'ensemble des applications proposées
- « Administrateur de Groupe » : c'est une personne autorisée à accéder à l'Intranet, qui possède les droits pour administrer un groupe de travail
- « Super-administrateur » : c'est la personne qui a la responsabilité de gérer et de veiller au bon fonctionnement matériel et logiciel de l'intranet

5.2 Le groupe de travail

Comme le montre le schéma ci-dessous (Figure 4), un groupe de travail de l'intranet est composé :

- d'utilisateurs, de sous-groupes et d'un administrateur du groupe de travail
- d'une zone de l'intranet avec un répertoire d'application, un répertoire de pages HTML, une zone de dépôt de documents non HTML
- d'un ensemble de droits associés à chaque utilisateur pour l'accès aux applications, aux pages HTML et aux fichiers de données de la zone de dépôt du groupe de travail
- d'un outil de publication de page HTML accessible à l'administrateur de groupe
- d'un outil de partage de fichiers (Application de Gestion de Fichiers AGF) accessible à tous

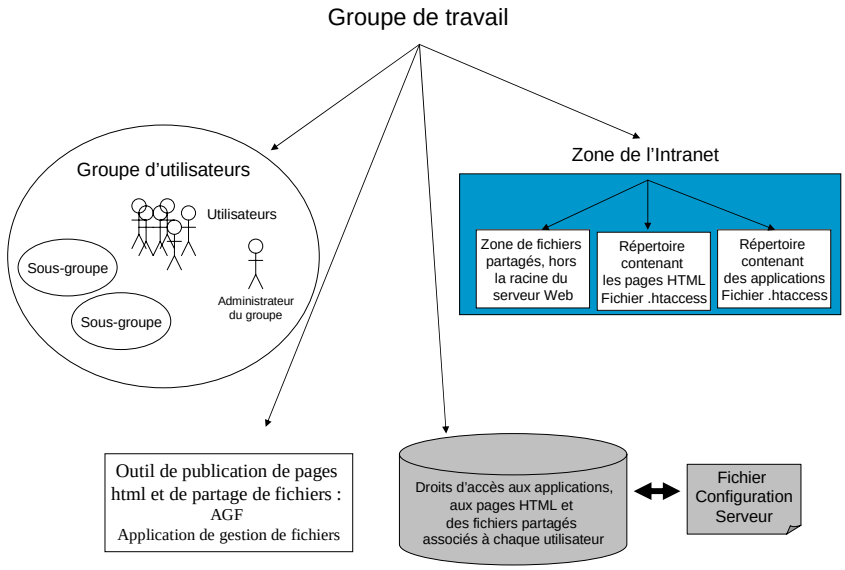

Figure 4 Groupe de travail

5.3 Architecture de l'application

L'application se divise en deux modules : le module module « accès au ressources » et module « administration ». Le module « accès au ressources » permet à un utilisateur d'accéder à aux pages HTML, à ses applications et aux fichiers partagés en fonction des protections qui ont été mises en place.

Architecture Globale

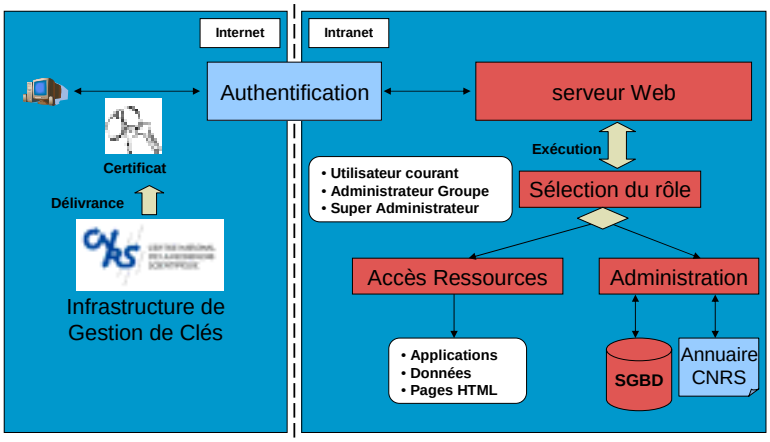

Figure 5 Architecture de l'application

Le second module « administration » permet aux administrateurs de groupe et au super-administrateur de :

- Gérer des utilisateurs de l'Intranet
- Gérer des groupes de travail et des sous-groupes
- Gérer des ressources
  - Gérer des applications de l'Intranet
  - Gérer des pages HTML d'un groupe de travail
  - Gérer des fichiers partagés
- Surveiller de l'Intranet

Administration

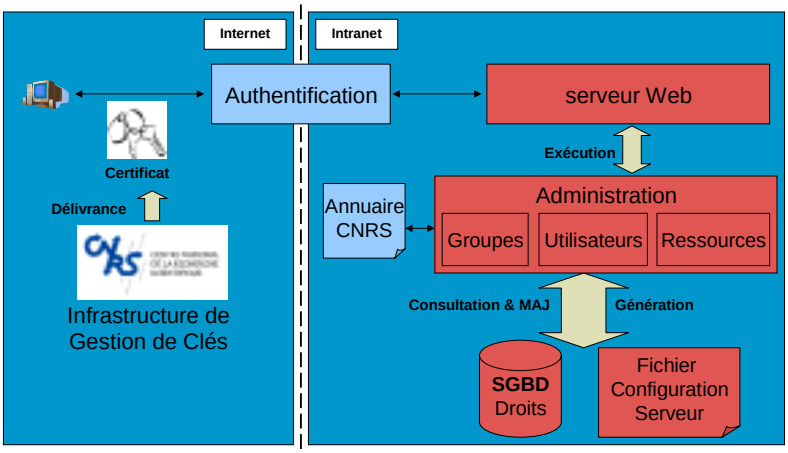

Figure 6 Le module d'administration

5.4 Fonctionnalités

Les fonctionnalités principales sont les suivantes :

- Gestion des utilisateurs de l'intranet (Figure 7)
  Les utilisateurs sont identifiés par le DN de leur certificat. Les utilisateurs peuvent être intégrés soit manuellement ou soit à partir d'un annuaire LDAP. En effet, A2C2 récupére les informations de l'utilisateur directement dans un annuaire LDAP et propose au surper-administrateur de sélectionner les personnes désirées. Pour l'intranet du département STIC, nous utilisons une Infrastructure de Gestion de Clés (IGC) du CNRS.

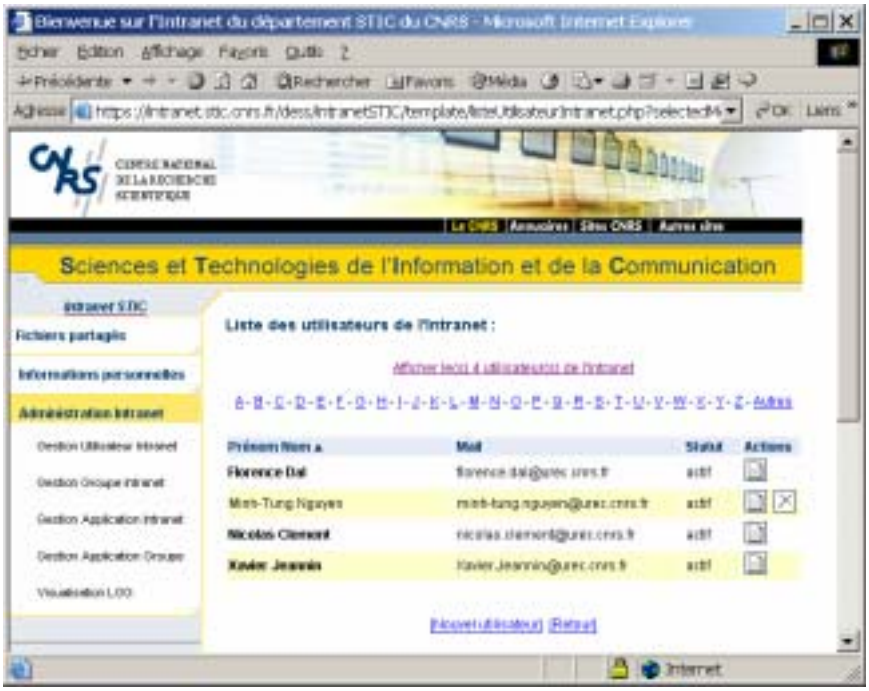

Figure 7 Exemple de gestion des utilisateurs de l'intranet

- Gestion des groupes de travail et des sous-groupes (Figure 8)

Figure 8 Exemple de sélection du groupe de travail

- Gestion des données d'un groupe de travail : le module AGF (Figure 9)
  Les fichiers partagés peuvent être protégés en lecture ou écriture et affectés à tout le groupe ou à des sous-groupes.

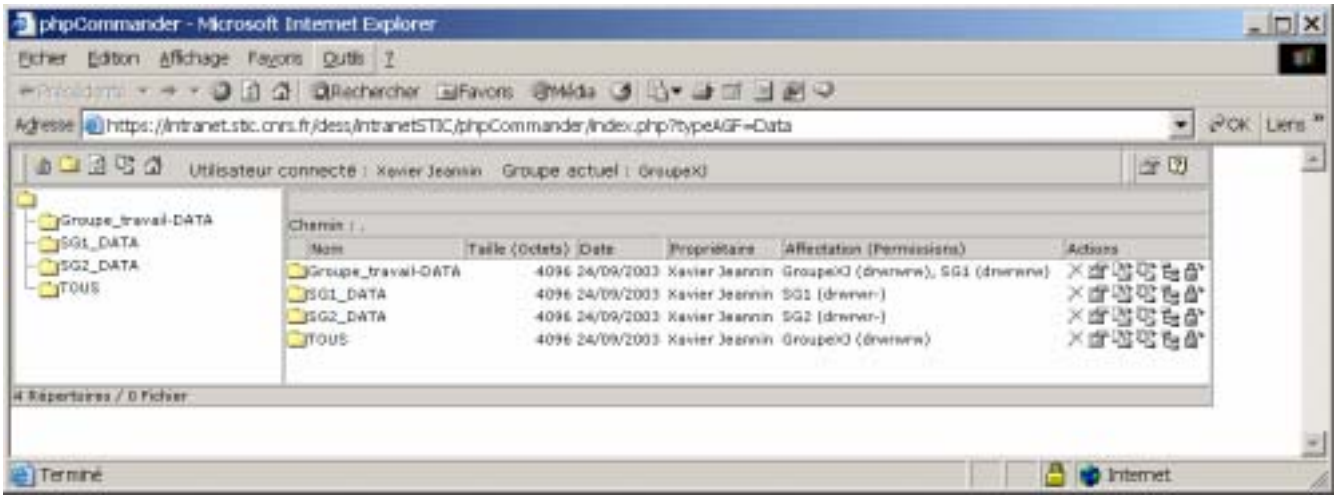

Figure 9 Module AGF de partage de données

- Gestion des pages HTML d'un groupe de travail : module AGF (Figure 10)
  L'accès aux répertoires du site Web peut être facilement contrôlé en fonction des groupes et sous-groupes.

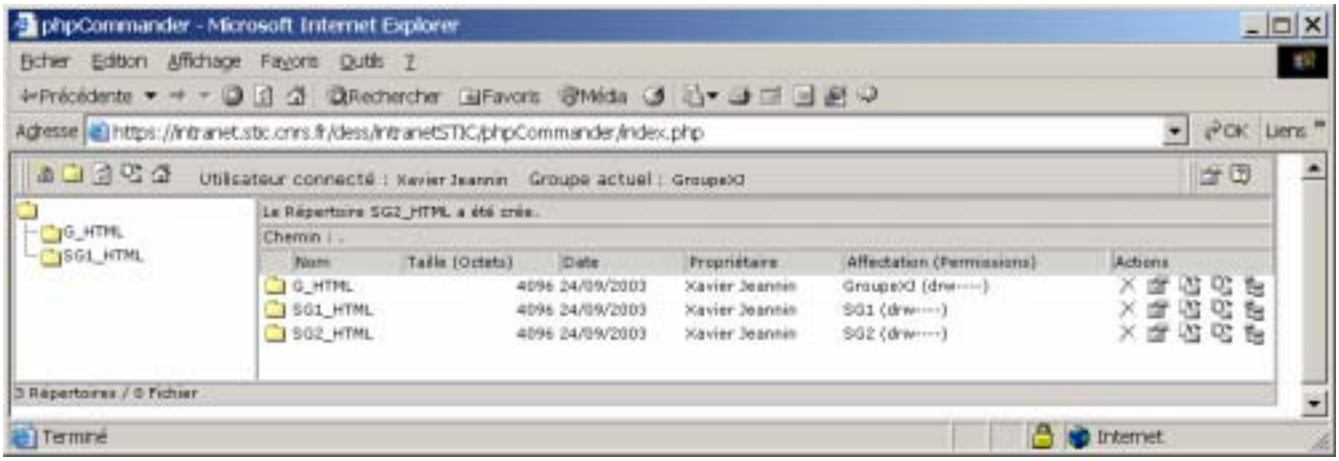

Figure 10 Dépôt et contrôle des accès au site Web du groupe de travail par le module AGF

- Gestion des applications de l'Intranet (Figure 11)
  Le super-administrateur incorpore des applications dans l'intranet. Les administrateurs de groupe choisissent parmi ces applications lesquelles ils veulent mettre à disposition de leur groupe. Le super-administrateur doit alors installer l'application pour le groupe et mettre en place « le login unique », c'est-à-dire le programme d'interface entre l'application et les certificats logcertif.php. Une fois cette étape réalisée, l'administrateur de groupe règle les droits d'accès à cette application puis crée au sein de l'application les comptes correspondant aux droits d'accès.

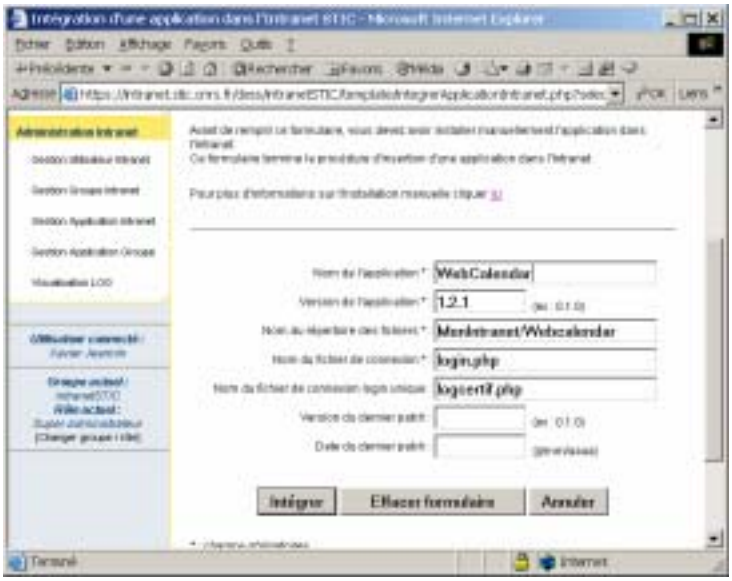

Figure 11 Formulaire permettant l'intégration d'une application

- Surveillance de l'Intranet (Figure 12)

Figure 12 Exemple de Log de l'intranet

6 Conclusion

L'augmentation de la demande de logiciels collaboratifs (Groupware) se traduit par la mise en place d'applications gérant des bases de compte indépendantes, entraînant ainsi la multiplicité des mots de passes pour les utilisateurs. Parallèlement, ces utilisateurs répartis dans différents réseaux, veulent des accès différenciés et sécurisés. Cette demande peut se traduire par la mise en place d'intranets sécurisés de personnes.

Notre projet est né de la demande du département STIC qui désirait créer un intranet lui permettant d'animer des groupes de travail répartis dans tous ses laboratoires. Le logiciel A2C2 a été créé pour permettre la construction d'intranet de ce type, avec possibilité également, de mise en œuvre dans les laboratoires du CNRS.

Une des particularités est le choix d'offrir à l'utilisateur la possibilité d'utiliser son certificat numérique comme moyen unique de connexion à l'intranet ; ces certificats n'étant généralement pas reconnus comme identifiant par les applications, A2C2 propose une solution qui permet d'interconnecter une application sans la modifier. Cependant comme cela a été précisé dans l'article c'est une application « modeste » de la problématique générale du SSO Web, étant parti du principe que ce SSO demeure un SSO local à l'intranet.

En résumé le logiciel A2C2 propose d'une part une solution qui permet d'interconnecter une application sans la modifier ; d'autre part offre la possibilité de gérer facilement des groupes de travail, en fournissant à chaque groupe de travail une fonction de publication de page HTML et de partage de fichiers incluant gestion des droits d'accès sur les pages HTML et gestion des droits de lecture et d'écriture pour les fichiers partagés. Pour la fonction de liste de diffusion nous nous sommes appuyés sur le logiciel Sympa qui supporte les certificats. Enfin la gestion des droits d'accès des membres de chaque groupe de travail est déléguée aux administrateurs de groupes qui sont plus à même, en tant qu'animateur de remplir cette fonction.

En ce qui concerne la mise en œuvre, pour le STIC les groupes de travail envisageables sont au nombre de 40, avec un nombre de personnes allant jusqu'à 250 par groupe. Les limites du programme ne sont pas connues, le projet étant encore très jeune et les applications fournies, au nombre six actuellement, n'étant pas forcément gourmandes : Tutos, Web Calendar (php), un autre WebCalendar en (CGI-bin en mod Perl), PHP2BB (avec modification), Squirremail, Labintel. La solution choisit permet d'utiliser rapidement les certificats pour les applications Web en attendant des solutions plus pérennes : adaptation des applications aux certificats, mise en place de véritables solutions de SSO ou mise en place d'infrastructure de gestion de privilèges.

Livré comme un logiciel Open-Source, A2C2 a été développé par deux stagiaires DESS (Florence Dal et Minh Tung Nguyen) et un ingénieur à mi-temps depuis janvier 2003. Conçu pour pouvoir être également utilisable dans les laboratoires du CNRS, il a été écrit avec des logiciels libres, à savoir : Apache, PHP, MySQL et a été testé sous Linux.

Remerciements

Nous remercions Nicolas Clément pour sa contribution au développement et son travail pour la fiabilisation du programme.

Références

[1] Nicole Dausque, Infrastructures de gestion de clefs. 09/05/2000.
[2] Jean-Luc Archimbaud, Certificats (électroniques) : Pourquoi ? Comment ? http://www.urec.cnrs.fr/securite/articles/, Décembre 2000.
[3] RFC2459 : Internet X.509 Public Key Infrastructure Certificate and CRL Profile http://www.pasteur.fr/cgi-bin/mfs/01/24xx/2459
[4] Serge Aumont, Claude Gross et Philippe Leca, Certificat X509 et Infrastructure de Gestion de Clés. Tutoriels JRES 2001.
[5] Security Assertion Markup Language (SAML) http://www.oasis-open.org/
[6] RFC2109 : HTTP State Management Mechanism http://www.ietf.org/rfc/rfc2109.txt
[7] http://www.sympa.org/