Fédération d'identités et propagation d'attributs avec Shibboleth

Olivier Salaün
Comité Réseau des Universités
olivier.salaun@cru.fr

Florent Guilleux
Comité Réseau des Universités
florent.guilleux@cru.fr

Pascal Aubry
IFSIC – Université de Rennes 1
pascal.aubry@univ-rennes1.fr

Résumé

Nous décrivons tout d'abord la problématique et les besoins en matière de fédération d'identités et de propagation d'attributs, en donnant des scenarii d'utilisation.

Nous donnons ensuite les solutions techniques existantes à même de répondre aux besoins énoncés, puis focalisons sur le système Shibboleth, projet internet2. Nous en décrivons le fonctionnement, et montrons comment ce système peut être mis en œuvre.

Nous décrivons enfin la fédération pilote du CRU, basée sur Shibboleth, et nous appuyons sur cet exemple pour mettre en exergue les problèmes organisationnels rencontrés lors de la mise en place d'une telle fédération.

Mots clefs

Fédération d'identités, propagation d'attributs, coopération inter-établissements, SAML, Shibbotleth.

Avertissement au lecteur

Cet article est le support du tutoriel du même nom présenté lors du congrès JRES2005, à Marseille en décembre 2005. Si la fédération d'identités et la propagation d'attributs sont des problématiques anciennes, les solutions présentées dans cet article sont pour la plupart récentes, et les implémentations et les formats d'échange montrés ici sont susceptibles d'évoluer sensiblement dans un futur proche.

Par ailleurs, plusieurs prérequis sont nécessaires pour une bonne compréhension de l'article. Le lecteur pourra se reporter aux références bibliographiques correspondantes pour plus d'informations. En particulier, aucun rappel n'est fait dans cet article sur les notions suivantes :

– Techniques web (paramètres CGI, redirections, *cookies*) ;
– Authentification unifiée (*Single Sign-On*) ;
– Certificats serveur.

Le lecteur pourra enfin se reporter à la liste située en fin de document pour une explication des abréviations employées au fil du texte.

La version de référence de ce document peut être trouvée sur le site de la fédération pilote du CRU (http://federation.cru.fr).

Table des matières

1 Problématique, besoins et scenarii d'utilisation

1.1 Problématique et besoins

La multiplication des référentiels d'authentification est une vieille habitude dans le monde des applications web. Lorsqu'une application ressent le besoin de restreindre l'accès à tout ou partie de ses services, elle crée un référentiel d'utilisateurs, avec des identifiants, des rôles, des mots de passe, un système de rappel/réallocation de ces mots de passe. Ce foisonnement de référentiels s'est fait au détriment de la sécurité, de l'intégration des services et surtout au détriment de l'ergonomie pour les utilisateurs.

La tendance s'est inversée dans notre contexte de l'enseignement supérieur avec le déploiement des ENT [1] dont deux briques majeures sont l'annuaire (LDAP [2]) et l'authentification unifiée (*Single Sign-On*, ou SSO [3]). On ne peut en effet se passer du service d'authentification unifiée lorsqu'on propose un portail des services aux utilisateurs. Mais ce service d'authentification reste cantonné à des besoins internes à l'établissement ; il n'est pas utilisable par l'étudiant lorsqu'il accède à un cours en ligne proposé par une université partenaire (dans le contexte d'une UNR [4] par exemple) ou par un chercheur accédant à un portail documentaire national. Faute de pouvoir interconnecter ces systèmes d'authentification locaux et les référentiels utilisateurs associés, les bases d'authentification dédiées se multiplient.

En particulier depuis l'avènement des UNR, les coopérations entre établissements sont de plus en plus nombreuses et étroites. Toutes les solutions artisanales qui consistaient en général à dupliquer les informations dans les systèmes d'information (SI) des établissements partenaires sont rendues caduques par le nombre : il ne s'agit plus aujourd'hui de régler quelques cas particuliers (étudiants aux inscriptions multiples pour des diplômes co-habilités ou personnels collaborant à des projets transversaux), mais de répondre à une problématique de nombre. Les techniques habituelles ont montré leurs limites et d'autres doivent les substituer afin de répondre aux ambitions politiques affichées en matière de collaboration.

La fédération d'identités répond à ce besoin d'interconnexion des systèmes d'authentification d'établissements en proposant deux services : la délégation de l'authentification et la propagation d'attributs utilisateur. Pour l'instant ces technologies se limitent au web mais des travaux sont en cours pour étendre leur champ d'application, notamment dans le domaine des grilles de calcul.

La **délégation de l'authentification** consiste à utiliser le service d'authentification proposé par l'établissement de rattachement de l'utilisateur, même lorsque l'application requérant cette phase d'authentification est un service extérieur à l'établissement. La phase d'orientation de l'utilisateur vers son « fournisseur d'identités » utilise des mécanismes standard (redirections HTTP [5], JavaScript et *cookies* [6]) ainsi qu'un service de découverte propre à la fédération. Le gestionnaire d'un service peut dès lors envisager plus largement l'accès à un service sans avoir à gérer des comptes utilisateurs pour des populations extérieures.

Le second service amené par la fédération d'identités est la **propagation d'attributs utilisateur**. Alors que la phase d'authentification ne fournit aux applications qu'un identifiant (éventuellement anonyme) pour l'utilisateur, ce second service consiste à collecter d'autres attributs relatifs à l'utilisateur auprès de son établissement de rattachement. Ces attributs sont de deux types :

- Ceux permettant de personnaliser le service (nom, prénom, adresse email…)
- Ceux requis pour effectuer un contrôle d'accès (catégorie d'utilisateur, formation, rôles …).

1.2 Quelques scenarii d'utilisation de la fédération d'identités

1.2.1 Ouvrir l'accès à une ressource locale (thèses, cours en ligne) à d'autres établissements

Dans ce cas de figure les services proposés ne sont pas sensibles mais l'université étant à l'origine de leur production désire en garder le contrôle et identifier au moins le profil et la provenance des utilisateurs y accédant. Les techniques de contrôle d'accès anciennement utilisées pour ce type de service contraignent l'utilisateur potentiel à se créer un compte d'authentification au niveau de la ressource ; ce compte étant mis en relation avec son adresse électronique.

L'utilisation de la fédération d'identités permet de se baser sur le système d'authentification de l'utilisateur dans son établissement. Cette solution permet surtout de baser le contrôle d'accès sur des attributs de l'utilisateur, issus de son établissement de rattachement. On peut ainsi ouvrir l'accès à la ressource sans devoir connaître a priori la population concernée, mais en ayant la possibilité de définir finement le profil des utilisateurs.

1.2.2 Gérer un intranet pour une population disséminée dans plusieurs établissements

On considère ici un groupe de personnes appartenant à différents établissements et amenés à travailler ensemble donc à partager des documents, des outils de travail collaboratif (forums, *wikis*, gestionnaires d'enquêtes, autres outils métier). L'accès à ces documents et outils doit être restreint aux membres du groupe de travail, ce qui pose le problème d'un référentiel d'authentification.

Ce scénario est différent du précédent dans la mesure où la population des utilisateurs est limitée et on peut baser le contrôle sur une énumération des membres du groupe maintenue localement. La valeur ajoutée de la fédération d'identités est ici le principe de délégation de l'authentification auprès du « fournisseur d'identités » de chaque utilisateur. En revanche son utilisation impose que tous les établissements de rattachement concernés opèrent un service de type « fournisseur d'identités ».

1.2.3 Au sein d'un établissement, étendre les fonctions du Single Sign-On *via l'accès aux attributs utilisateur*

Les logiciels de *Single Sign-On* [7] se cantonnent généralement à un service d'authentification des utilisateurs et n'assurent pas la remontée d'attributs dont les applications ont pourtant besoin. En l'absence d'un service commun, on condamne chaque application à devoir directement interroger l'annuaire LDAP et les autres référentiels pour accéder aux attributs utilisateur. Cette tâche peut être partiellement mutualisée au niveau du portail (comme le fait le projet ESUP-Portail [8] à l'aide des groupes uPortal [9] par exemple).

L'utilisation du service de fédération d'identités (voire uniquement de sa composante de gestion d'attributs) à l'échelle d'un établissement pourrait combler ce vide fonctionnel.

1.2.4 Gérer l'authentification pour des populations à la frontière de l'établissement (anciens ou futurs étudiants)

Les applications de pré-inscription des étudiants ou d'enquêtes auprès des anciens étudiants concernent des population qui ne sont pas encore ou plus gérées dans le système d'information de l'établissement. On ne dispose donc pas de service d'authentification pour ces utilisateurs qui ne rentrent pas forcément dans le moule (déjà complexe) des utilisateurs (étudiants, chercheurs, enseignants, autres personnels…).

Dans la mesure ou un service d'authentification dédié (procédures de gestion des comptes allégées) peut être mis en place pour ces utilisateurs, la fédération d'identités facilite la cohabitation des différents service d'authentification au sein du même établissement.

1.2.5 Accès aux périodiques électroniques depuis un ENT

Les établissements de la communauté enseignement supérieur / recherche souscrivent auprès de fournisseurs de documentation électronique du secteur privé (Elsevier, JSTOR…). Le contrôle d'accès à ces « périodiques électroniques » est souvent basé sur la restriction par plages d'adresses IP. Ce mode de contrôle d'accès s'adapte mal à des utilisateurs nomades, alors que les projets d'ENT visent à favoriser ces pratiques via leur portail web. Des solutions artisanales de type reverse proxy sont mises en place par certains établissements pour permettre l'accès à ces périodiques via l'ENT.

En utilisant le service de fédération d'identités pour authentifier l'utilisateur auprès des fournisseurs de documentation électronique, on supprime la limitation liée au nomadisme. Cette solution aurait le mérite de standardiser les modes d'authentification pour accéder à la documentation électronique.

2 Les solutions techniques

Au fil des années, différentes solutions pour authentifier et contrôler l'accès aux applications ont été mis en œuvre. Si elles apportent satisfaction dans certains cas, elles présentent également des défauts que nous présentons ici :

- **Ne pas contrôler l'accès** à l'application : la rendre publique ou « cacher » son URL d'accès mais cette URL est diffusable, et elle peut être référencée par des moteurs de recherche ;
- Offrir un **compte générique partagé** entre plusieurs utilisateurs : diffusion incontrôlable du mot de passe associé au compte, nécessité de le renouveler si un utilisateur quitte le groupe, nouvel identifiant et mot de passe à retenir pour les utilisateurs ;
- Pour chaque utilisateur **un compte spécifique** à la ressource : gestion fastidieuse de ces comptes (création des comptes, allocation des mots de passe, gestion des pertes de ces mots de passe, mise à jour de la liste des utilisateurs, etc.), nouvel identifiant et mot de passe à retenir pour les utilisateurs ;
- Contrôle **par adresse IP** : il faut gérer les plages d'adresses IP autorisées, le nomadisme est impossible (sauf via un *reverse proxy* associé à un SSO), la ressource ne connaît pas l'identité de l'utilisateur ;
- **Certificat personnel** : il faut que tous les utilisateurs soient dotés d'un certificat personnel ce qui est rarement le cas car leur déploiement est lourd, de plus l'utilisation d'un certificat n'est pas triviale ;
- Mise en place d'un **méta annuaire** : multiplication des méta annuaires au fil des partenariats, exposition des mots de passe, il faut l'accord des gestionnaires des annuaires pour constituer ce méta annuaire, en sus d'un travail d'intégration des schémas d'annuaires s'ils divergent.

Face à ces difficultés l'idée est venue de se reposer sur l'établissement de rattachement d'un utilisateur pour l'authentifier, c'est **la délégation d'authentification** : il s'agit de déporter la phase d'authentification préalable à l'accès à une application au service d'authentification de l'établissement de rattachement de l'utilisateur. L'application se contente d'une assertion d'authentification de la part de ce service, lui indiquant si l'utilisateur a été correctement authentifié ou non. L'application reste maîtresse de son contrôle d'accès, mais estime que l'établissement de rattachement d'un utilisateur est le plus à même d'authentifier correctement ce dernier.

Techniquement, des mécanismes web classiques permettent d'envisager une telle délégation, et la généralisation des *Single Sign-On* au sein des établissements assure que l'on puisse se reposer sur un service d'authentification fiable pour la majorité sinon la totalité des utilisateurs. Cependant dans un tel contexte où se répartissent des tâches entre différents systèmes, des normes standard sont nécessaires.

2.1 SAML

SAML (*Security Assertion Markup Language* [10]) a été initialement conçu pour permettre entre autres la délégation d'authentification. C'est devenu un standard OASIS [11] en 2002. Il s'agit d'un ensemble de spécifications qui définissent comment des services peuvent s'échanger des assertions de sécurité (authentification, autorisation, attributs), indépendamment des technologies utilisées par chacun de ces services (PKI [12], SSO, LDAP, Kerberos, etc.). SAML ne couvre donc pas tout le spectre de la gestion des identités, par exemple ne définit pas de protocole de SSO ou une sémantique d'attributs standard. Il s'appuie sur des standards pré existants (XML, SSL, etc.) et a été conçu avec suffisamment d'abstraction pour rendre inter opérable des systèmes hétérogènes, et s'articuler au mieux avec d'autres mécanismes de gestion d'identités.

SAML est constitué de différents protocoles, qui correspondent aux différents cas d'usage adressés par ce standard. Un protocole SAML décrit de façon abstraite comment une entité interagit avec

un système SAML, généralement sous la forme d'une séquence de requêtes et de réponses. Un *protocol binding* est la traduction d'un tel protocole abstrait en un protocole de communication implémentable informatiquement, par exemple sous la forme de *Web Services* SOAP [13]. De plus, SAML étant très abstrait pour assurer l'interopérabilité des systèmes (notamment sur la composition des messages), il existe des « profils SAML » qui restreignent (ou étendent) la variabilité d'un protocole de base pour des usages particuliers. En s'accordant sur l'utilisation d'un certain profil, deux entités voulant communiquer en SAML se simplifient l'interopérabilité.

Voici un exemple d'assertion (d'authentification SAML) :

```
<saml:Assertion
    MajorVersion="1"
    MinorVersion="0"
    AssertionID="128.9.167.32.12345678"
    Issuer="Comite Reseau des Universites"
    IssueInstant="2002-03-21T10:02:00Z">

    <saml:Conditions
        NotBefore="2002-03-21T10:02:00Z"
        NotAfter="2002-03-21T10:07:00Z" />

    <saml:AuthenticationStatement
        AuthenticationMethod="password"
        AuthenticationInstant="2002-03-21T10:02:00Z">

        <saml:Subject>

            <saml:NameIdentifier
                SecurityDomain="www.cru.fr"
                Name="dupont" />
        </saml:Subject>

    </saml:AuthenticationStatement>

</saml:Assertion>
```

Il s'agit d'un document XML signé et envoyé par un service A à un autre service, et qui dit en substance « cet utilisateur a été correctement authentifié par le service A ». SAML ne sert pas à authentifier l'utilisateur (n'importe quel système d'authentification peut être utilisé), mais à communiquer le fait qu'il a été correctement authentifié. L'assertion contient ici des informations sur la date et le mode d'authentification, et possède une durée de validité limitée.

Outre SAML, il existe aussi les spécifications portées par le consortium WS-I (*Web Services interoperability*) [14], notamment WS-Security (*Web Services Security*) [15] et WS-Federation (*Web Services Federation Language*) .

SAML est déjà implémenté dans beaucoup de produits, et sert de fondation à deux autres normes, Shibboleth [17] et Liberty Alliance [18].

2.2 Shibboleth

Shibboleth [17] est développé depuis 2001 par Internet2 [19] et désigne à la fois une norme et un produit (open source). C'est une extension de SAML qui enrichit ses fonctionnalités de fédération d'identités en facilitant pour un ensemble de partenaires la mise en place de deux fonctionnalités importantes, la délégation d'authentification et la propagation d'attributs. Shibboleth a été conçu pour répondre aux besoins des communautés de l'enseignement supérieur et est déjà utilisé dans plusieurs pays : Etats-Unis, Angleterre, Suisse, Finlande, etc.

2.3 Liberty Alliance

Liberty Alliance [18] est un consortium d'entreprises fondé en 2001 qui a produit plusieurs spécifications (des *frameworks*) sur la gestion d'identités. ID-FF (*Identity Federation*

Framework [20]) enrichit SAML pour permettre la fédération de comptes, la délégation d'authentification, la propagation de fin de session. ID-WSF (*Identity-based Web Services Framework* [21]) offre la propagation d'attributs utilisateur, la recherche de service d'identités, etc. De nombreux produits propriétaires et plusieurs solutions *open source* implémentent tout ou partie des spécifications Liberty Alliance.

2.4 Le choix de Shibboleth

Le Comité Réseau des Universités a retenu Shibboleth pour construire une infrastructure de fédération d'identités pour l'enseignement supérieur français. Les fonctionnalités offertes par Shibboleth sont moins étendues que celles de Liberty Alliance, mais correspondent aux principaux cas d'usages de notre communauté. Surtout la topologie d'une fédération de type Shibboleth correspond bien à la structuration d'un ensemble d'établissements d'enseignement supérieur.

De plus c'est un produit *open source*, soutenue par une communauté active et ouverte, et qui s'interface bien avec les briques pré existantes d'un système d'information. Enfin Shibboleth et Liberty Alliance ayant un socle commun, SAML, les besoins d'interopérabilité seront très probablement satisfaits quand ils se poseront. Notamment la version 2.0 de Shibboleth sera compatible avec SAML 2.0.

3 Le système Shibboleth

La terminologie de Shibboleth a évolué ces derniers mois pour se stabiliser dans un document [22] de septembre 2005. Les termes utilisés par Shibboleth ont été traduits en français pour simplifier la lecture par notre communauté francophone. Nous donnons néanmoins systématiquement les équivalents anglophones afin de lier cet article aux documentations officielles du projet Shibboleth.

L'objectif de cette partie est de montrer les interactions entre les acteurs du système qui permettent la délégation de l'authentification et la propagation des attributs utilisateurs.

Afin de mieux appréhender un fonctionnement globalement complexe, nous présentons d'abord les acteurs du système, puis détaillons les interactions entre les acteurs du système lors de l'accès par un navigateur à une ressource web.

3.1 Les acteurs du système

*3.1.1 Le navigateur (***User Agent***)*

Shibboleth répond à la problématique des applications web. Le premier acteur de l'architecture Shibboleth est donc logiquement le navigateur de l'utilisateur. Le navigateur doit répondre aux exigences habituelles en matière de navigation web, notamment l'interprétation des codes de retour HTTP (redirections [5]), ainsi que l'acceptation et la transmission des *cookies* selon les normes en vigueur [6] (cela vaut pour la très grande majorité des navigateur web du marché).

*3.1.2 Le fournisseur de services (***Service Provider** *ou SP)*

Une entité proposant des ressources web sur la base d'un contexte de sécurité SAML est appelée « fournisseur de services » (ou *Service Provider)*, et sera par la suite nommée SP.

Le fournisseur de ressource a en particulier la charge de donner ou non l'accès aux ressources, en fonction des attributs utilisateur.

Notons dès à présent que les ressources web sont *a priori* quelconques ; on trouvera ainsi :

- Des **applicatifs web**, dont le contrôle d'accès peut être effectué indifféremment par un pré-filtre (tel un module Apache) ou au sein même de la logique applicative. Ces applicatifs pourront également utiliser les attributs utilisateur à d'autres fins que le contrôle d'accès (la personnalisation de l'interface ou la définition des rôles des utilisateurs par exemple) ;
- Des **pages statiques**, dont le contrôle d'accès doit être effectué par un pré-filtre.

*3.1.3 Le fournisseur d'identités (***Identity Provider** *ou IdP)*

Une entité authentifiant les utilisateurs et fournissant leurs attributs est appelée « fournisseur d'identités » (ou *Identity Provider*) et sera par la suite notée IdP.

Le fournisseur d'identités s'appuie sur le SI de l'établissement, tant pour l'authentification que pour la récupération des attributs utilisateur à propager. De ce fait, il est en général situé au plus proche du SI.

3.1.4 Le WAYF

Le WAYF (pour *Where Are You From?*, « d'où êtes-vous ? ») est un service dont le but est d'orienter l'utilisateur vers son IdP. Ce service est optionnel est discuté en détail dans la partie 3.5.

3.2 Le fonctionnement de Shibboleth sans SSO

3.2.1 Première requête vers un SP

Dans ce premier cas d'étude, nous considérons que le SP connaît l'établissement de rattachement de l'utilisateur, c'est-à-dire l'établissement qui pourra l'authentifier. Cela correspond exactement au cas d'un SP qui propose des ressources aux utilisateurs d'un seul établissement, le sien.

Le navigateur effectue donc une requête HTTP vers le SP afin d'accéder à une ressource (cf Figure 1).

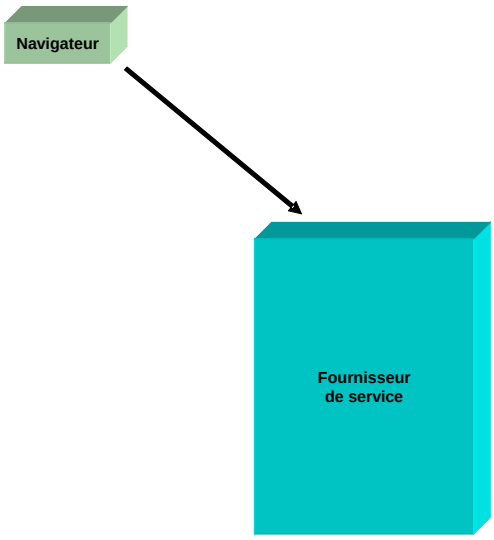

Figure 1 *Premier accès du navigateur au SP*

Le SP, sans information d'authentification, redirige vers l'IdP de l'établissement de rattachement de l'utilisateur (cf Figure 2).

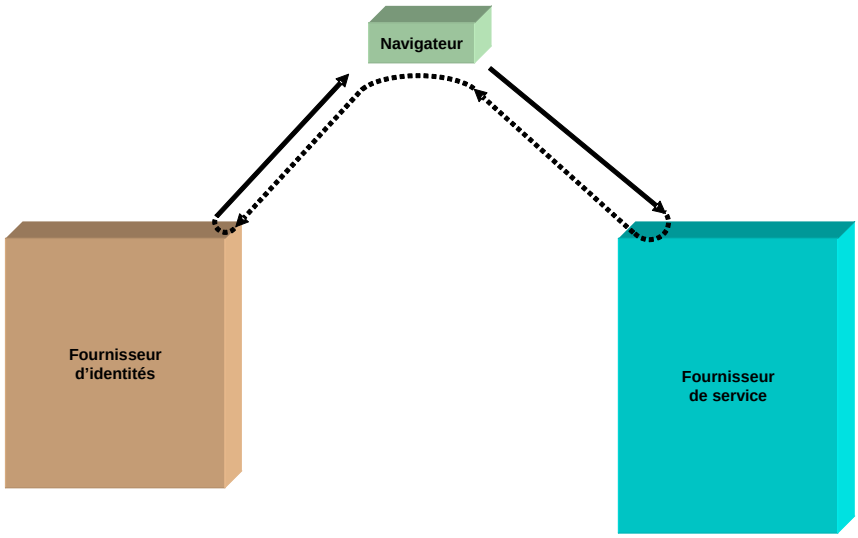

Figure 2 *Redirection du SP vers l'IdP*

Hors contexte SSO (qui sera étudié ultérieurement), la réponse de l'IdP est une demande d'authentification, sous la forme d'une erreur *401 Unauthorized* ou d'un formulaire web. L'utilisateur soumet alors son identifiant et son mot de passe (cf Figure 3).

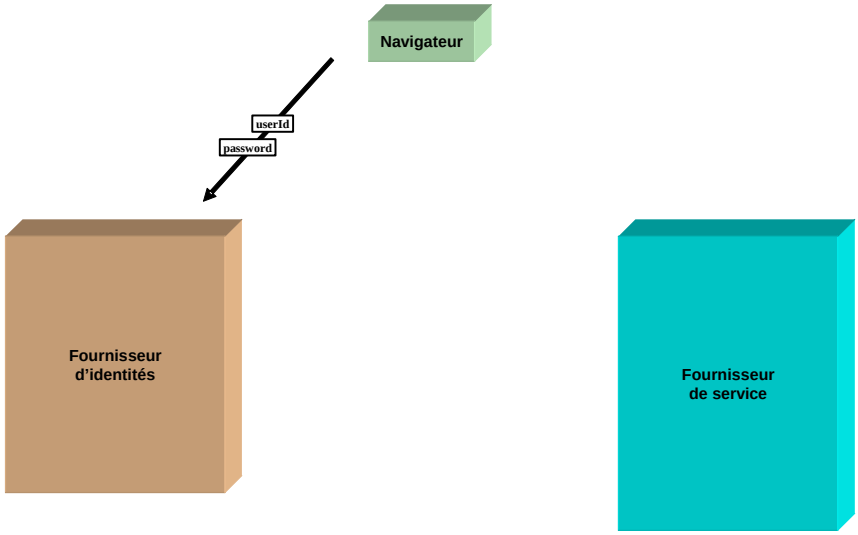

Figure 3 *Authentification de l'utilisateur auprès de l'IdP*

Une fois l'utilisateur authentifié, l'IdP redirige alors le navigateur vers le SP, accompagné d'une assertion SAML. Cette assertion est signée par l'IdP, le SP pourra donc faire confiance à l'assertion. Elle contient un identifiant appelé *nameIdentifier* (cf Figure 4).

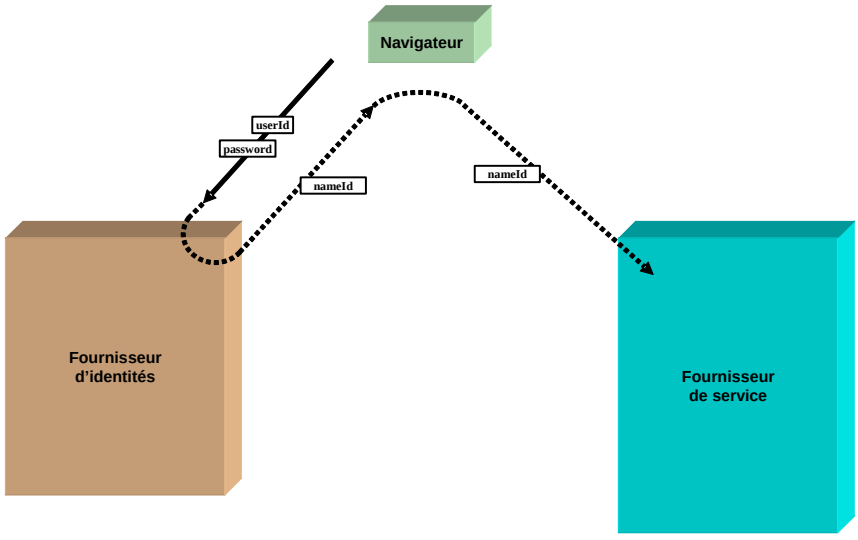

Figure 4 *Transmission de l'identifiant opaque(nameIdentifier) de l'IdP vers le SP*

Cet identifiant est opaque, c'est-à-dire qu'il ne contient pas d'information personnelle concernant l'utilisateur. Il n'est utilisé que dans le cadre des échanges entre les différentes briques de Shibboleth, et n'est connu ni de la ressource accédée ni du SI de l'établissement. Un exemple de *nameIdentifier* est montré ci-dessous.

```
<saml:NameIdentifier
    Format="urn:mace:shibboleth:1.0:nameIdentifier"
    NameQualifier="https://idp.example.org/shibboleth">
3f7b3dcf-1674-4ecd-92c8-1544f346baf8
</saml:NameIdentifier>
```

C'est cet identifiant opaque qui va permettre au SP de récupérer les attributs de l'utilisateur auprès de l'IdP. Les attributs de l'utilisateur sont transmis au SP par l'IdP, via l'appel d'un *Web Service*, et en échange du *nameIdentifier* (cf Figure 5).

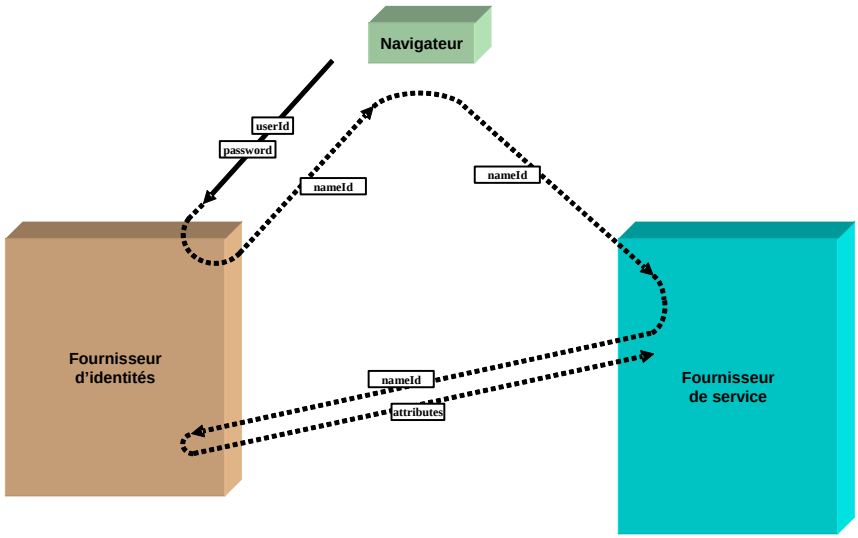

Figure 5 *Récupération des attributs de l'utilisateur par le SP auprès de l'IdP*

Le SP peut alors effectuer le contrôle d'accès, éventuellement utiliser les attributs de l'utilisateur dans la logique applicative, puis retourner une réponse au navigateur (cf Figure 6).

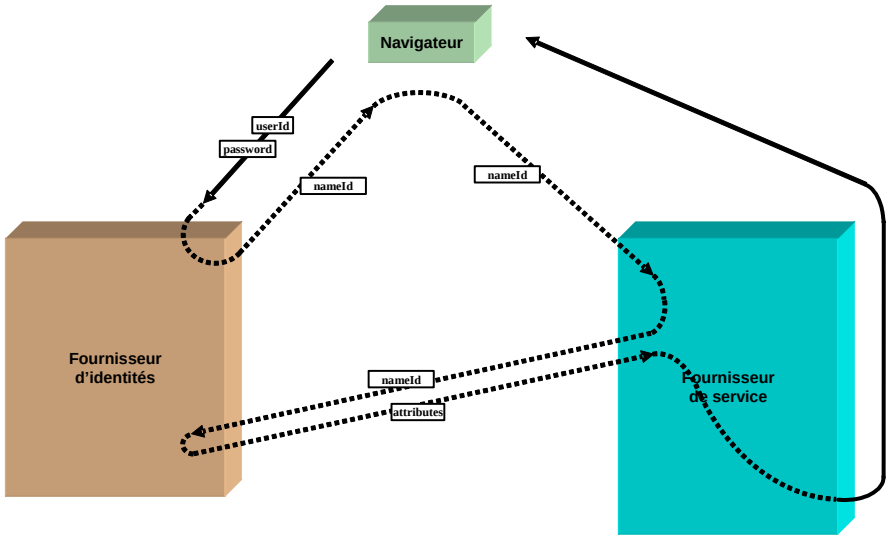

Figure 6 *Envoi de la réponse du SP au navigateur*

Pour résumer, du point de vue de l'utilisateur (cf Figure 7), celui-ci :

– Effectue une requête auprès du SP (1) et reçoit une demande d'authentification de l'IdP (2) ;

– S'authentifie auprès de l'IdP (3) et reçoit une réponse du SP (4).

En fonction du résultat retourné par le contrôleur d'accès, la réponse du SP sera une page web ou bien une erreur HTTP (en général *403 Not Allowed*).

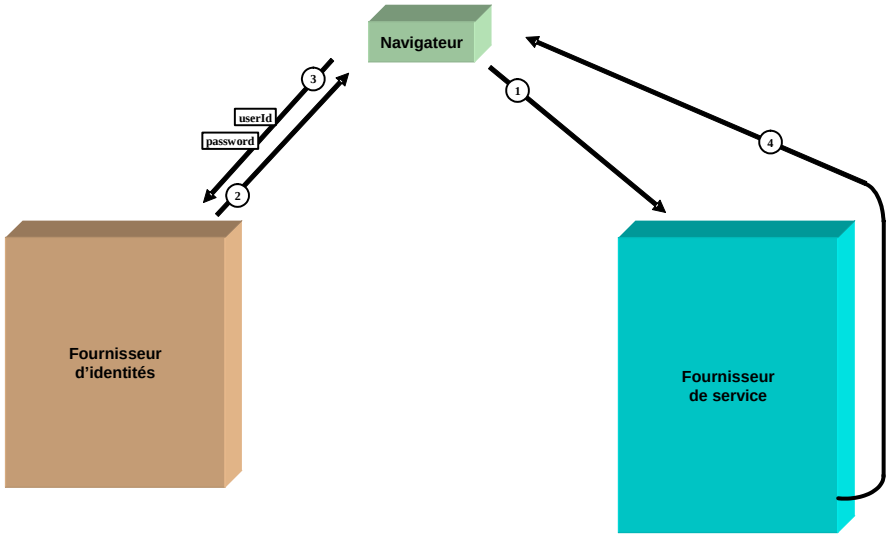

Figure 7 *Point de vue de l'utilisateur*

3.2.2 Requêtes suivantes vers le même SP

Lors de la première requête, le SP mémorise le fait que l'utilisateur a bien été authentifié, ce qui est fait dans la session mise en place entre le navigateur et le SP. De cette manière, aucune redirection n'est plus nécessaire pour toutes les requêtes suivantes.

L'IdP n'est ainsi utilisé que lors de la première requête au SP (cf Figure 8).

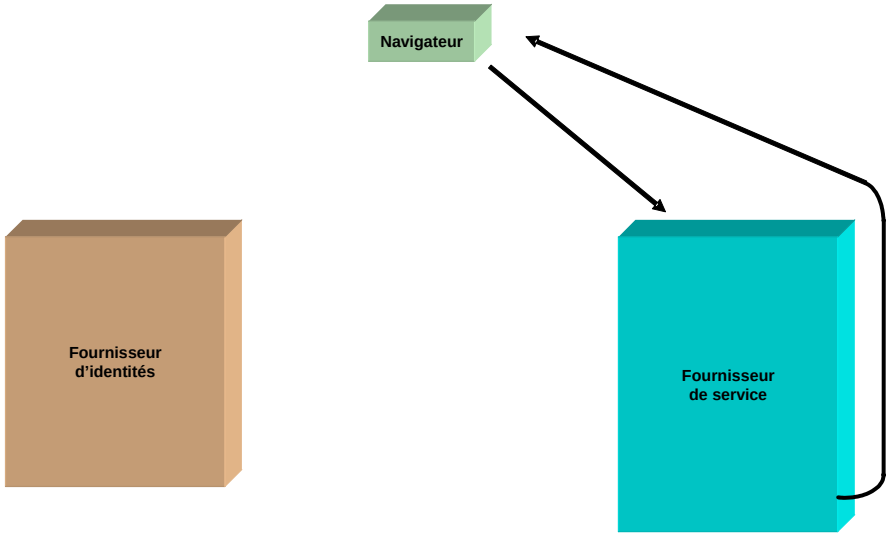

Figure 8 *Requêtes suivantes vers le même SP*

3.2.3 Architecture logique du fournisseur de services (SP)

Le fournisseur de services est composé de trois briques logicielles :

- Le consommateur d'assertions (*Assertion Consumer Service*),
- Le demandeur d'attributs (*Attribute Requester*),
- Le contrôleur d'accès.

Le **consommateur d'assertions** agit comme un pré-filtre. C'est lui qui redirige vers l'IdP lorsque l'utilisateur n'est pas authentifié. Il peut être implémenté au niveau du serveur HTTP (par un module Apache ou un filtre J2EE par exemple) ou encore par une librairie, appelée par un applicatif web. Lorsque l'utilisateur est authentifié, alors le consommateur d'assertions transmet le *nameIdentifier* au demandeur d'attributs.

Le **demandeur d'attributs** est chargé de la récupération des attributs des utilisateurs auprès de l'IdP. Il peut être implémenté comme un démon (dédié, interrogeable par les processus du SP) ou par une librairie, interrogeable par un applicatif web. Les attributs récupérés par le demandeur d'attributs sont fournis au contrôleur d'accès.

Le **contrôleur d'accès** est chargé d'autoriser ou non l'accès aux ressources demandées. Il peut être implémenté au niveau du serveur HTTP (par un module Apache ou un filtre J2EE par exemple) ou encore par une librairie, appelée par un applicatif web.

La Figure 9 montre l'architecture logique d'un SP et son fonctionnement interne lors de la phase d'authentification.

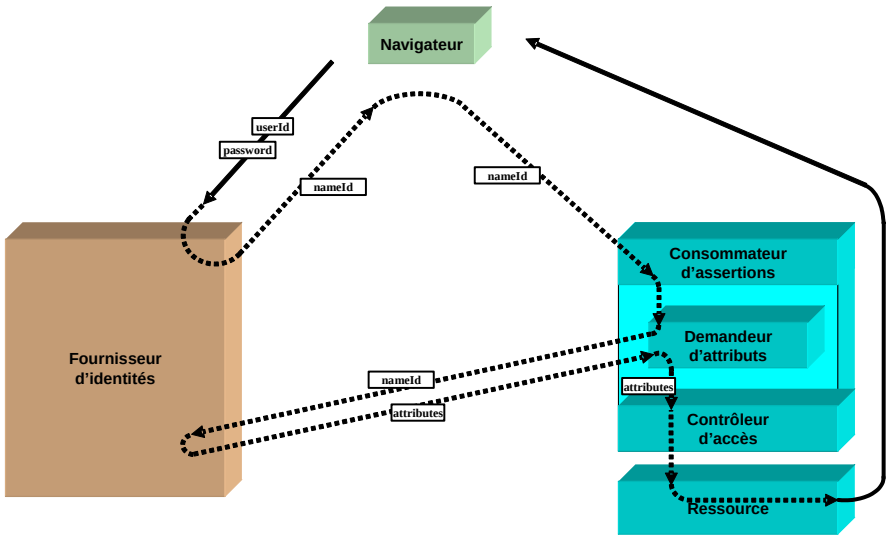

Figure 9 *Architecture logique et fonctionnement interne d'un SP*

3.2.4 Architecture logique du fournisseur d'identités (IdP)

Un fournisseur d'identités est composé de trois briques logicielles :

- Le service d'authentification (*Authentication Service*),
- L'autorité d'authentification (*Authentication Authority*),
- L'autorité d'attributs (*Attribute Authrority*).

Le **service d'authentification** est chargé de l'authentification des utilisateurs vis-à-vis de l'ensemble de l'IdP. C'est lui qui, par exemple, demande à l'utilisateur un couple *user*/*password*, puis le valide auprès de la base d'authentification du SI. Les implémentations du service

d'authentification peuvent être très variées, depuis un module Apache authentifiant les utilisateurs auprès d'un annuaire LDAP, jusqu'à un client de Single Sign-On comme nous le verrons ultérieurement.

Le service d'authentification n'est pas, si l'on se réfère aux spécifications de Shibboleth [17], partie intégrante de l'IdP ; on ne peut néanmoins pas concevoir d'IdP sans service d'authentification. N'importe quel système d'authentification web est utilisable.

Le service d'authentification est chargé de transmettre à l'autorité d'authentification l'identifiant unique de l'utilisateur au sein du SI. N'importe quel système d'authentification web peut être utilisé (formulaire applicatif, royaume HTTP, certificat X509 [12], *Single Sign-On*).

L'**autorité d'authentification** associe le *nameIdentifier* à l'identifiant de l'utilisateur.

L'**autorité d'attributs** délivre, en réponse à une demande d'un SP, les attributs de l'utilisateur correspondant à un *nameIdentifier*, l'association entre l'identifiant de l'utilisateur et le *nameIdentifier* étant maintenue par l'autorité d'authentification. Les attributs de l'utilisateur sont récupérés dans le SI de l'établissement, plusieurs sources pouvant être envisagées (annuaire LDAP, base de données…).

La Figure 10 montre l'architecture logique d'un IdP et son fonctionnement interne lors de la phase d'authentification.

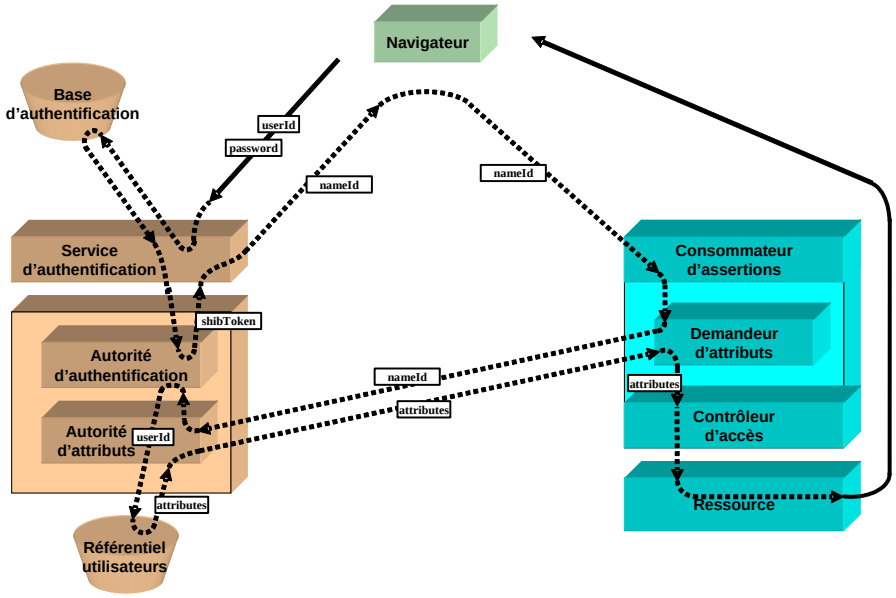

Figure 10 *Architecture logique et fonctionnement interne d'un IdP*

3.3 Le fonctionnement de Shibboleth avec SSO

Nous supposons que l'établissement est pourvu d'un système de SSO, tel CAS (*Central Authentication System* [23][24]).

3.3.1 Première requête vers un SP

Dans le modèle de CAS, l'authentification n'est pas directement prise en charge par le service d'authentification de l'IdP ; celui-ci ne fait que rediriger le navigateur vers le serveur de SSO, qui renvoie alors à l'utilisateur un formulaire d'authentification (cf Figure 11).

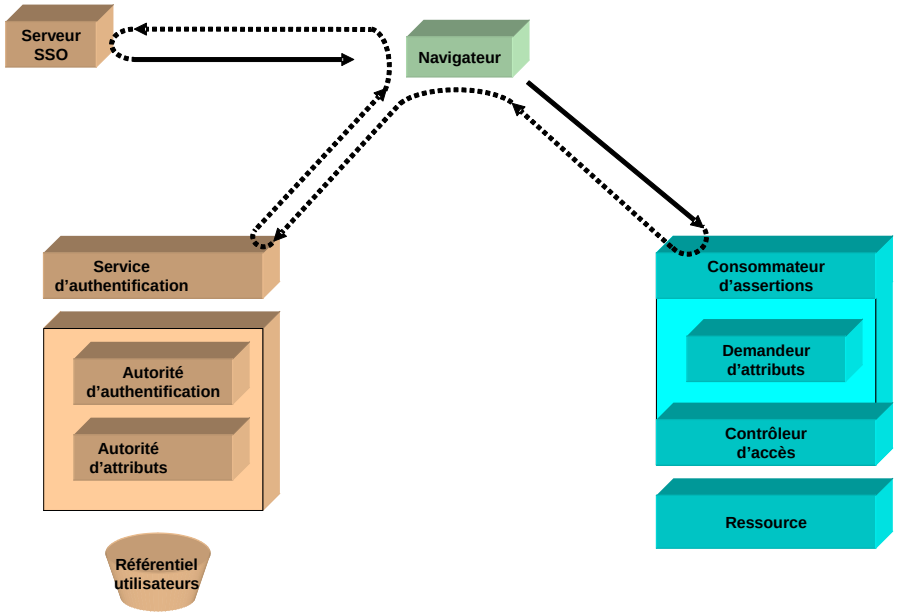

Figure 11 *Première requête à un SP dans un contexte SSO*

Le navigateur remplit alors le formulaire et effectue une nouvelle requête vers le serveur SSO, qui le redirige vers l'IdP. Le service d'authentification de l'IdP, client SSO, effectue alors une nouvelle redirection vers le SP et l'authentification se déroule ensuite comme vu précédemment (cf Figure 12).

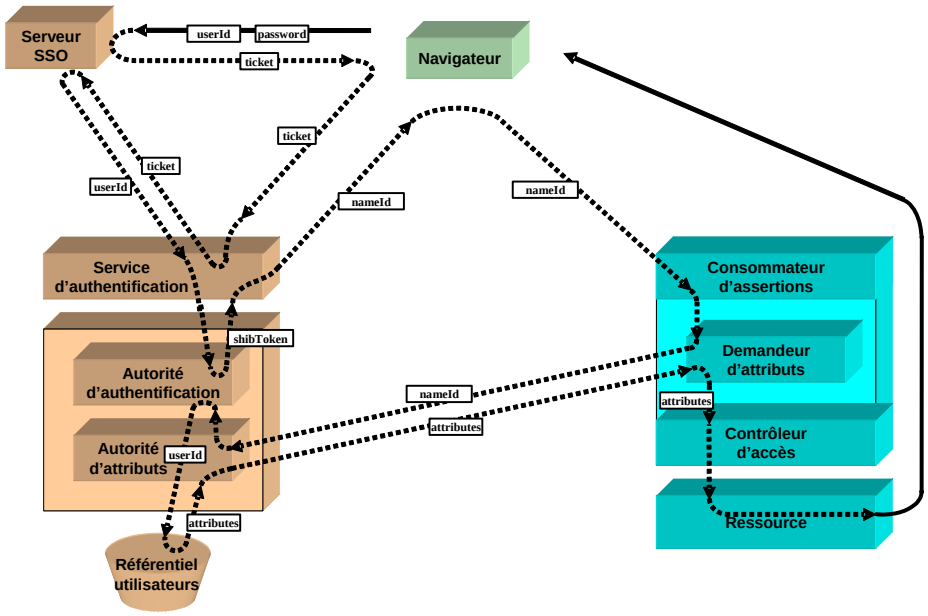

Figure 12 *Redirection vers un SP par le serveur SSO*

Notons que, dans ce cas, l'identifiant de l'utilisateur n'est pas fourni par le navigateur, mais récupéré auprès du serveur CAS par l'IdP.

Pour résumer, du point de vue de l'utilisateur (cf Figure 13), celui-ci :

- Effectue une requête auprès du SP (1) et reçoit une demande d'authentification du serveur SSO (2) ;
- S'authentifie auprès du serveur SSO (3) et reçoit une réponse du SP (4), qui donne accès ou non à la ressource demandée.

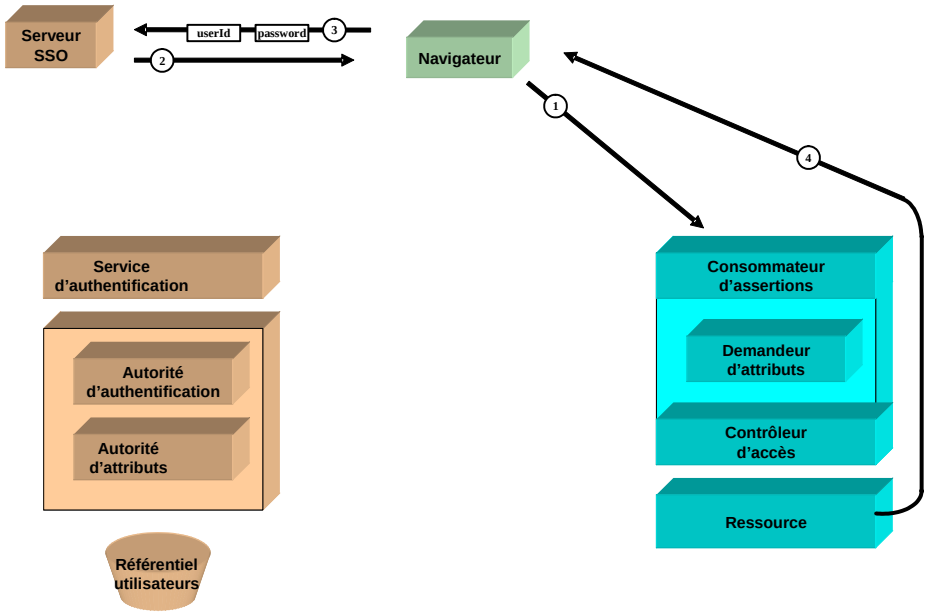

Figure 13 *Point de vue de l'utilisateur dans un contexte SSO*

3.3.2 Requêtes suivantes au même SP

Comme vu précédemment, une session étant mise en place entre le navigateur et le SP (en fait, le consommateur d'assertions du SP), ni l'IdP ni le serveur SSO n'interviennent plus par la suite pour l'accès au même SP (cf Figure 14).

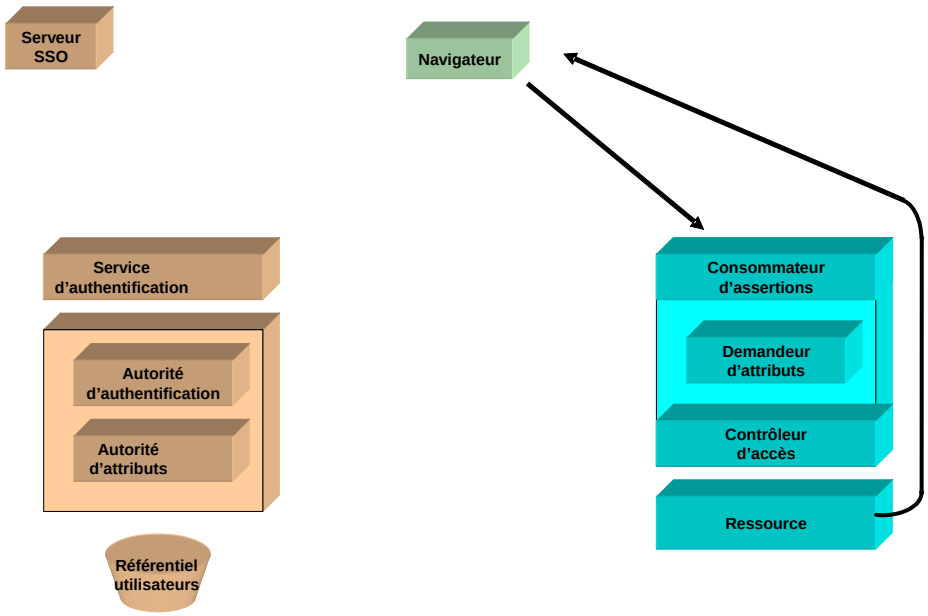

Figure 14 *Requêtes suivantes vers le même SP dans un contexte SSO*

3.3.3 Requêtes suivantes vers un autre SP

Lorsque l'utilisateur est déjà authentifié auprès du serveur SSO, le navigateur dispose d'un identificateur de session (par exemple un TGC pour CAS) qui lui permet de ne pas avoir à s'authentifier à nouveau (cf Figure 15).

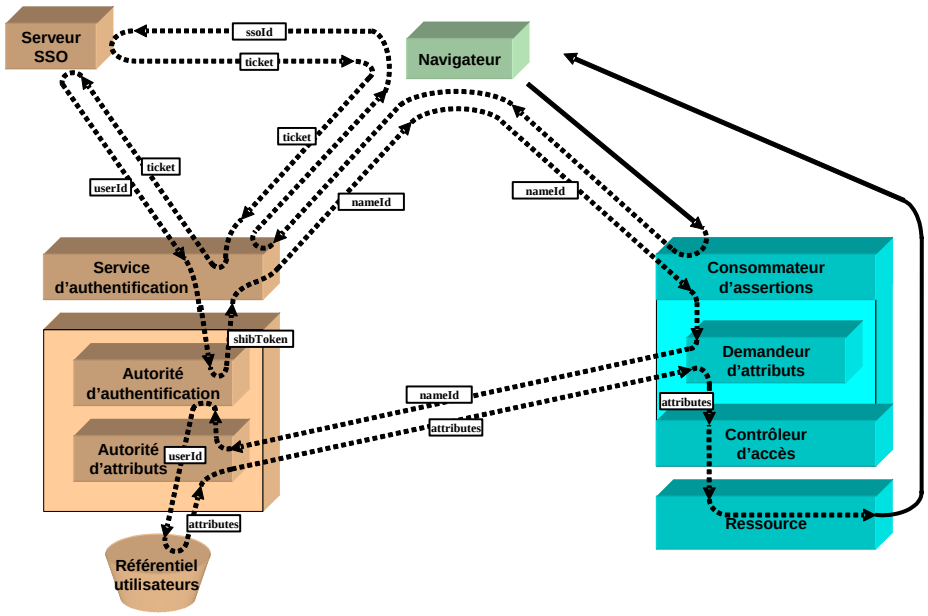

Figure 15 *Requêtes suivantes vers un autre SP dans un contexte SSO*

L'authentification de l'utilisateur est alors complètement transparente (cf Figure 16).

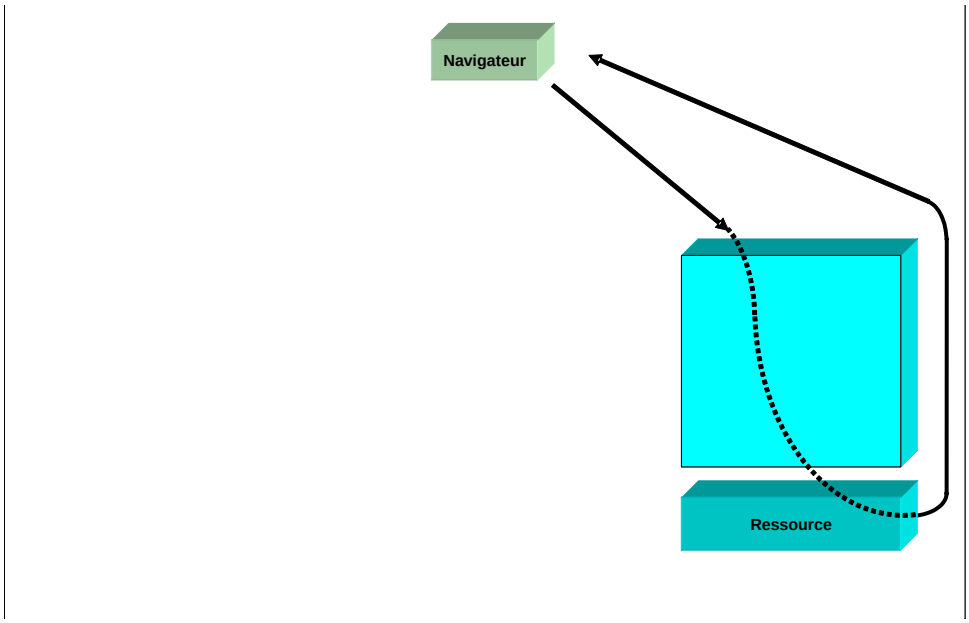

Figure 16 *Point de vue de l'utilisateur pour les requêtes suivantes vers un autre SP dans un contexte SSO*

3.4 Le fonctionnement de Shibboleth avec SSO et WAYF

Nous nous plaçons maintenant dans le cas où un SP est accessible à des utilisateurs rattachées à des établissements différents. Cela est par exemple le cas d'une université souhaitant mettre à disposition de tous les personnels de l'enseignement supérieur de sa région les archives de ses thèses et publications scientifiques.

Le problème qui se pose alors est que le SP ne sait pas vers quel IdP rediriger le navigateur pour l'authentification. Il est résolu grâce au WAYF, dont le rôle est d'orienter les utilisateurs pour sélectionner leur IdP.

3.4.1 Première requête vers un SP

Lors de la première requête au SP, celui-ci ne sachant pas quel IdP sera utilisé, le SP redirige le navigateur vers le WAYF (cf Figure 17).

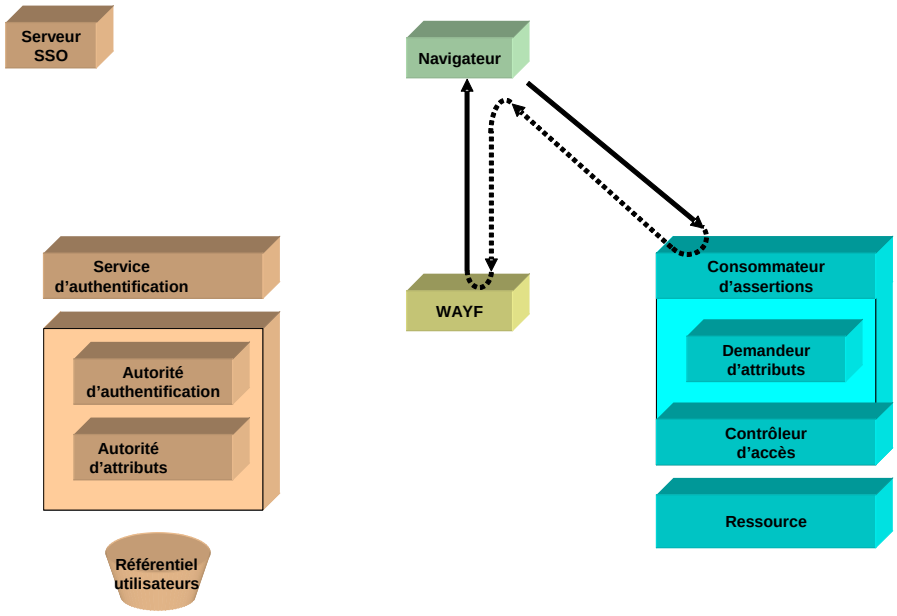

Figure 17 *Redirection du SP vers le WAYF*

Le WAYF affiche alors à l'utilisateur alors une liste d'IdP possibles. La requête suivante, vers le WAYF, redirige le navigateur vers l'IdP choisi par l'utilisateur, qui a son tour redirige le navigateur vers le serveur SSO, qui propose alors un formulaire d'authentification (cf Figure 18).

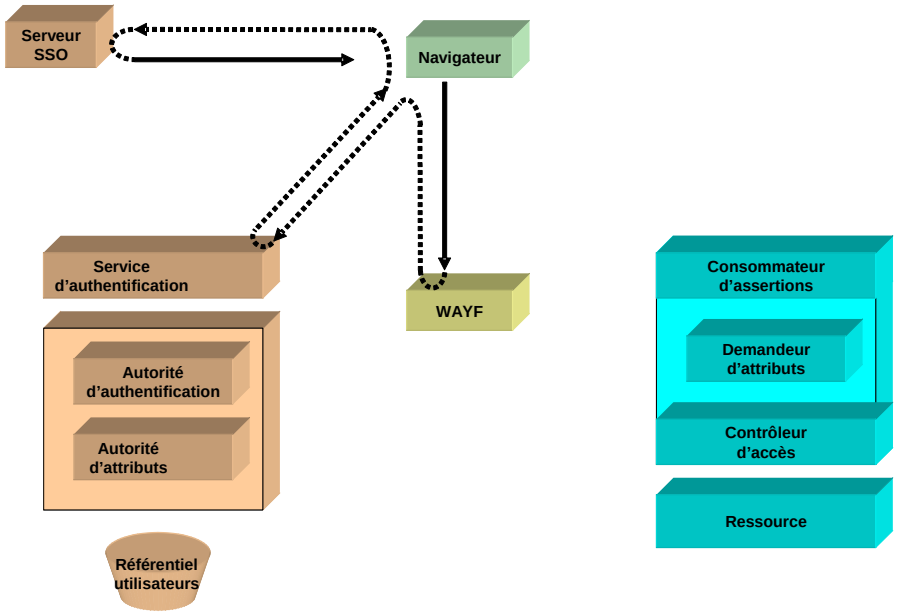

Figure 18 *Redirection du WAYF vers l'IdP, puis le serveur SSO*

Le navigateur s'authentifie alors auprès du serveur SSO, et l'authentification se déroule ensuite comme vu précédemment (cf Figure 19).

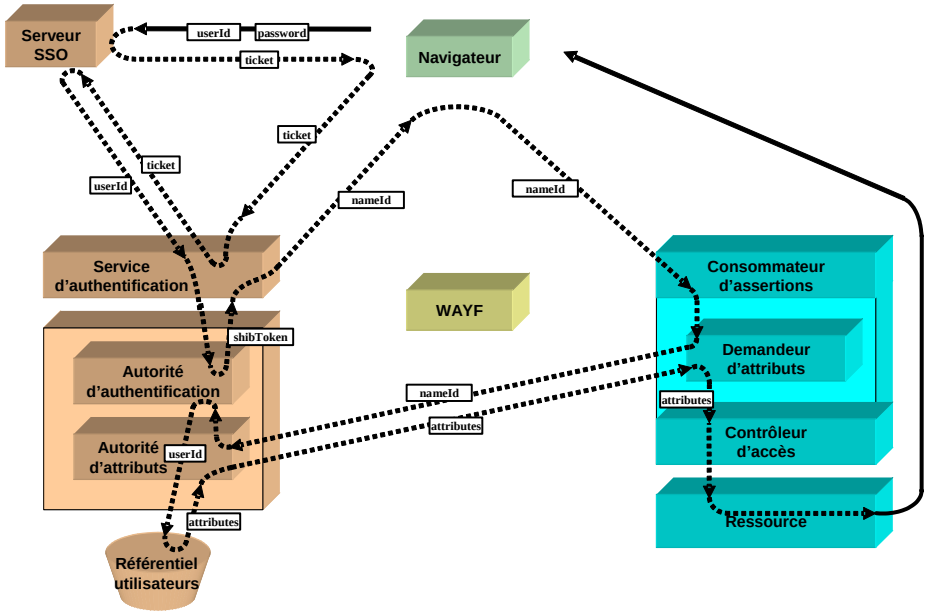

Figure 19 *Redirection du serveur SSO vers l'IdP, puis le SP*

Pour résumer, du point de vue de l'utilisateur (cf Figure 20), celui-ci :

– Effectue une requête auprès du SP (1) et reçoit une demande d'aiguillage du WAYF (2) ;

– Sélectionne son IdP auprès du WAYF (3) et reçoit une demande d'authentification du serveur SSO (4) ;
– S'authentifie auprès du serveur SSO (5) et reçoit une réponse du SP (6), qui autorise ou non l'accès à la ressource demandée.

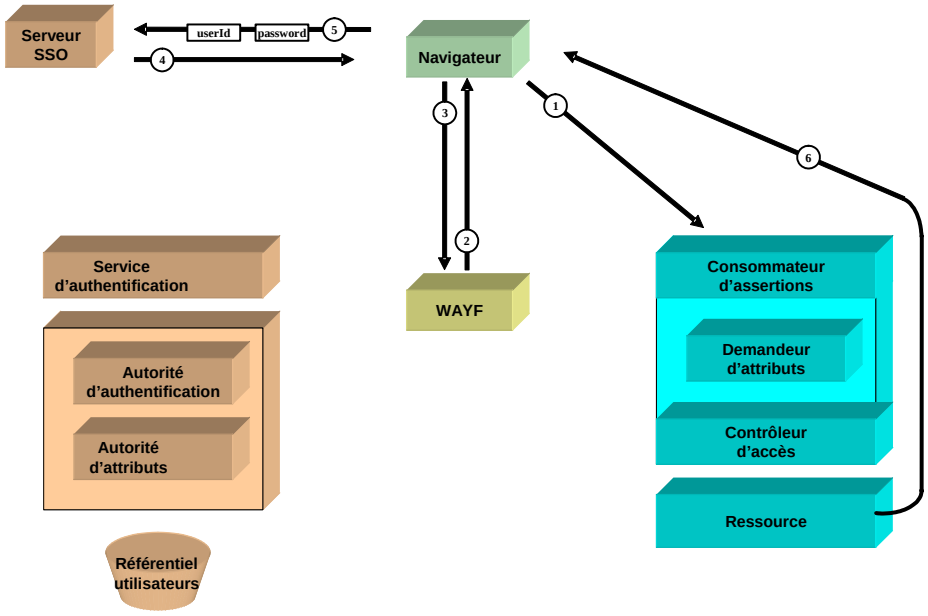

Figure 20 *Point de vue de l'utilisateur dans un contexte SSO et WAYF*

3.4.2 Requêtes suivantes vers le même SP

Comme vu précédemment, une session étant mise en place entre le navigateur et le SP (en fait, le consommateur d'assertions du SP), ni le WAYF, ni l'IdP ni le serveur SSO n'interviennent plus par la suite pour l'accès au même SP (cf Figure 21).

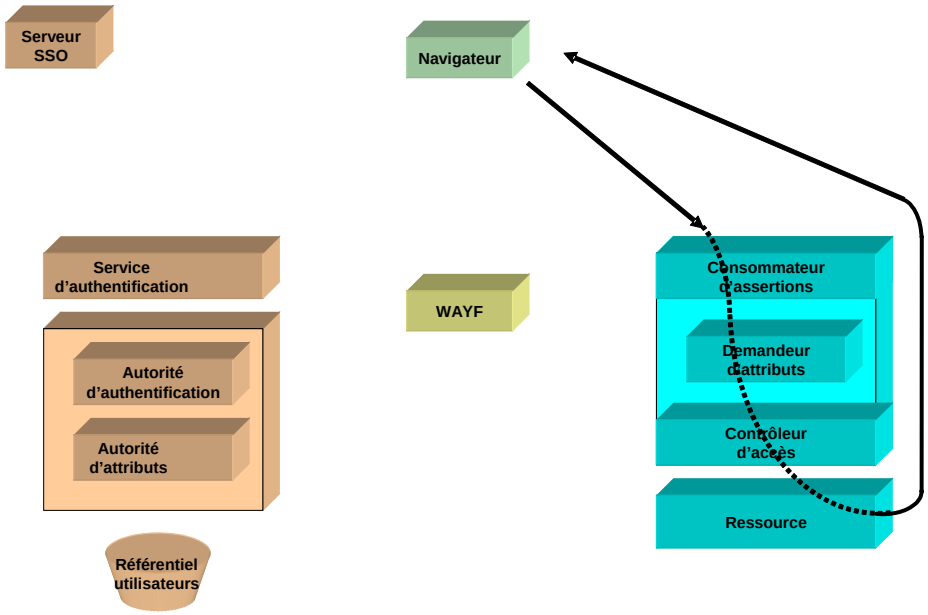

Figure 21 *Requêtes suivantes vers le même SP dans un contexte SSO et WAYF*

3.4.3 Requêtes suivantes vers un autre SP

Lors du choix de l'IdP par l'utilisateur, il est possible pour le WAYF de mémoriser ce choix dans le navigateur (à l'aide d'un *cookie*). Dans ce cas, le WAYF peut utiliser ultérieurement cette information, et faire en sorte que les requêtes suivantes soient non bloquantes (en redirigeant automatiquement vers l'IdP choisi la première fois). La Figure 22 montre comment, dans ce cas, l'authentification de l'utilisateur est totalement transparente lors de l'accès à un autre SP.

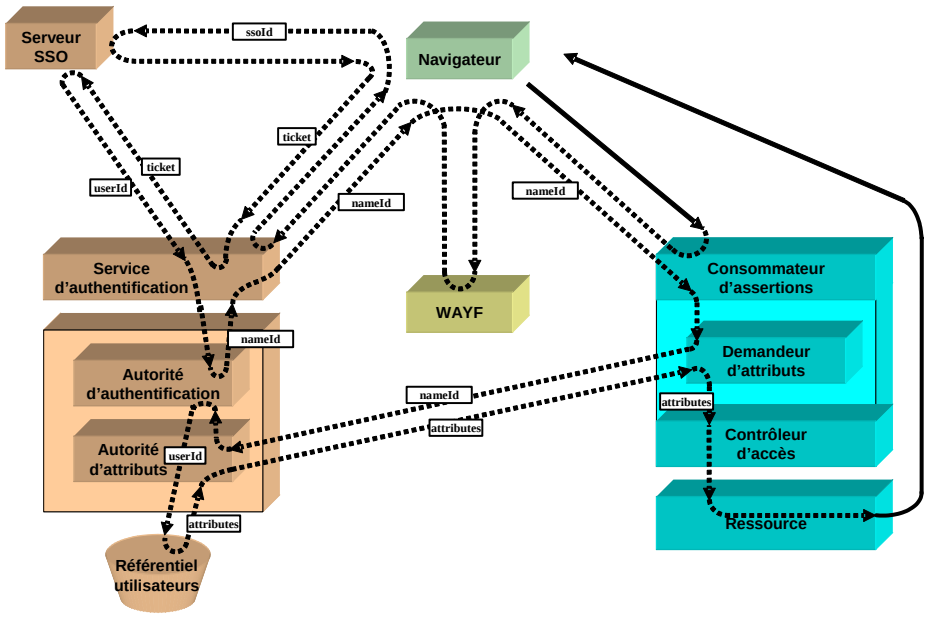

Figure 22 *Requêtes suivantes vers un autre SP dans un contexte SSO et WAYF*

3.5 A propos du WAYF

3.5.1 La place du WAYF dans le système Shibboleth

Plusieurs questions se posent quant à la place du WAYF :

- Combien faut-il de WAYF au sein d'une fédération ?
- Où faut-il le(s) placer ? Dans quel réseau ? À côté de quels autres acteurs du système Shibboleth ?

Les impératifs guidant les réponses à ces questions sont[1] :

- La disponibilité du service ;
- Le coût de déploiement (installation et configuration) ;
- La souplesse de maintenance (essentiellement la maintenance de la liste des IdP).

Les deux derniers impératifs militent pour un déploiement minimal, à savoir un seul WAYF au sein d'une fédération (partagé entre plusieurs établissements).

Il va de soi que la redondance d'un WAYF n'est envisagée que dans un but de tolérance aux fautes : tout comme les serveurs SSO, la montée en charge n'est jamais un souci.

[1] Nous ne prenons pas en compte ici le coût de développement (qui est fixe, alors que le coût de déploiement et de maintenance est linéaire). Ce coût est néanmoins à considérer tant que le seul WAYF proposé aujourd'hui est centralisé et ergonomiquement pauvre.

De manière native, Shibboleth ne permet pas de déclarer, pour un SP, plusieurs WAYF : un SP doit rediriger vers un seul WAYF. Notons que c'est la nature même de la redirection HTTP qui empêche un SP de s'appuyer sur plusieurs WAYF (ou plusieurs IdP en l'absence de WAYF). Cette redondance, habituelle pour d'autre protocoles comme LDAP ou DNS, rend d'emblée le système Shibboleth peu tolérant aux fautes du WAYF ou des IdP[2].

On pourrait alors penser à minimiser le coût de déploiement en centralisant à outrance, c'est-à-dire déployer par exemple un seul WAYF pour toute une communauté (regroupant plusieurs établissements). Si le réseau est aujourd'hui un prédicat, il n'en reste pas moins qu'il est parfois coupé. La coupure du réseau sur lequel est implanté le WAYF priverait alors tous les SP d'authentification, ce qui nous fait rejeter cette solution[3].

En poussant ce raisonnement, on voit ainsi aisément qu'il est souhaitable de placer le WAYF d'un SP sur le même réseau que le SP lui-même. De cette manière, la disponibilité du service est celle du SP, ce qui revient à dire que la disponibilité globale du service pour un utilisateur est celle combinée du SP et de son IdP. On voit ainsi que, lorsque le WAYF d'un SP est implanté au plus près du SP lui-même, le système Shibboleth n'introduit pas d'autre risque de panne réseau[4].

On peut donc, suivant ce principe, penser à intégrer le WAYF au sein même du SP. Si l'on veut minimiser le coût de maintenance, il est néanmoins intéressant de mutualiser les WAYF d'un même réseau. L'architecture d'une offre applicative d'un établissement dont l'authentification est confiée à Shibboleth sera donc souvent celle décrite par la Figure 23.

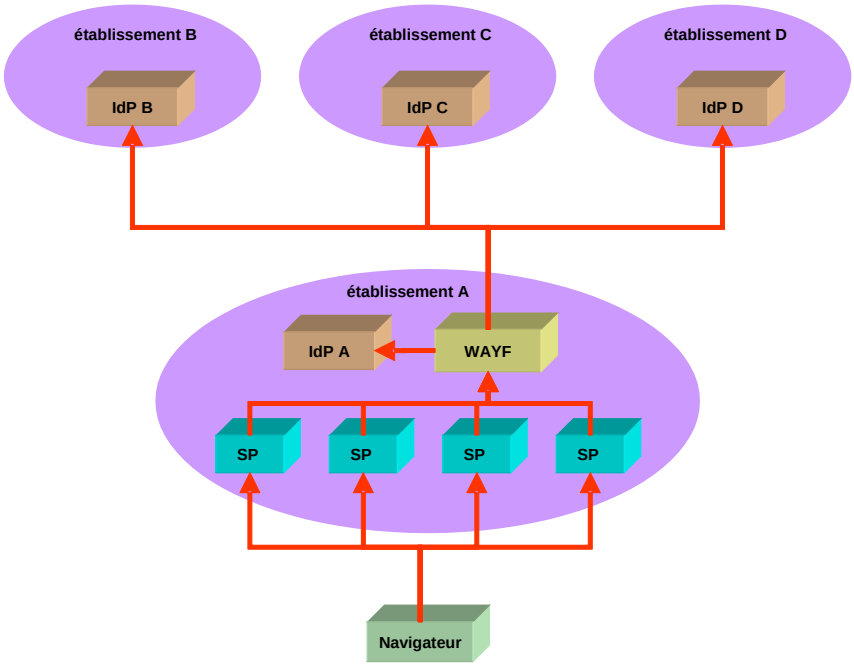

Figure 23 *Architecture classique d'une offre applicative basée sur Shibboleth*

[2] Le même problème est rencontré pour la redondance des serveurs SSO, dont le fonctionnement est également basé sur des redirections HTTP.

[3] Au moment de l'écriture de cet article, c'est pourtant cette solution qui a été retenue pour la fédération pilote du CRU [31], car elle facilite l'intégration de nouveaux partenaires au sein de la fédération. Cette solution devra être revue dans l'avenir.

[4] Les pannes logicielles ne sont pas discutées ici, puisque tous les logiciels utilisés ici sont du monde libre, donc améliorables à souhait (si nécessaire). Les pannes dues au système seront minimisées en s'appuyant sur des systèmes réputés stables, et les pannes matérielles seront minimisées grâce à une architecture redondante des serveurs (alimentation et baies disque notamment).

3.5.2 Le WAYF : un concept, plusieurs implémentations possibles

Seul l'objectif du WAYF est défini dans les spécifications de Shibboleth :

- Proposer aux utilisateurs une liste d'IdP, parmi lesquels les utilisateurs sont invités à sélectionner celui de leur établissement de rattachement ;
- Une fois le choix effectué, rediriger les utilisateurs vers l'IdP correspondant à leur choix.

Rien n'est spécifié en particulier sur les interactions de l'utilisateur avec le WAYF. On peut dès lors envisager toutes sortes de scenarii.

Par exemple, **en milieu confiné**, c'est-à-dire pour lesquels ont est absolument sûr de l'établissement de rattachement des utilisateurs, on peut imaginer qu'il n'y ait **aucune interaction entre l'utilisateur et le WAYF**. On peut, par exemple, considérer le cas de machines libre-service dans un établissement scolaire, pour lequel le WAYF redirigerait automatiquement vers un IdP déterminé en fonction de l'adresse IP du client sur lequel est exécuté le navigateur.

Le cas ci-dessus est évidemment assez rare, et c'est en général l'utilisateur seul qui connaît l'IdP auprès duquel il s'authentifie. De manière triviale, la liste des IdP sera présentée sous la forme d'une **liste déroulante**.

Dans le cadre d'une fédération importante, le nombre d'IdP présenté peut rendre l'ergonomie du WAYF incompatible avec la simple présentation d'une liste déroulante. On peut alors envisager tous les types de pré-aiguillage des utilisateurs, par exemple :

- Un ou plusieurs **niveaux thématiques**, proposant une liste de groupes d'IdP (par exemple par UNR), et un second niveau permettant la sélection de l'IdP ;
- Un ou plusieurs **niveaux géographiques** proposant une liste de régions (par exemple sous forme de cartes cliquables), puis un dernier niveau permettant de sélectionner un IdP parmi ceux d'une région.

La Figure 24 montre une page possible d'interaction de l'utilisateur avec un WAYF proposant le choix de la région, puis du département, puis de l'établissement par ceux d'un département (ici le choix du département).

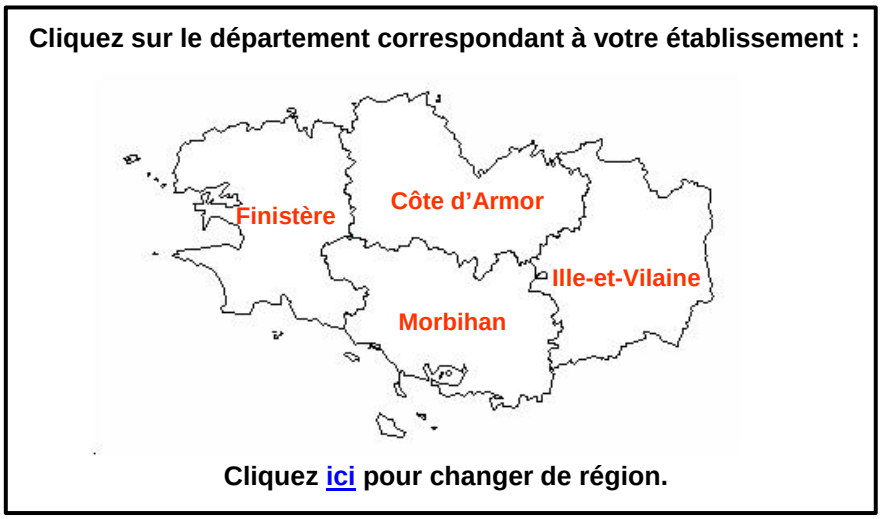

Figure 24 *Exemple d'interaction élevée entre les utilisateurs et un WAYF*

On peut également imaginer que le WAYF, surtout s'il possède un niveau élevé d'interaction tels que ceux décrits ci-dessus, pré-positionne la sélection de l'utilisateur, en fonction de la provenance attendue des utilisateurs, selon le client sur lequel est exécuté le navigateur, ou encore selon le

choix effectué par les utilisateurs précédemment (utilisation d'un *cookie*). Un tel pré-positionnement peut également être à l'initiative du SP, par le passage d'un paramètre supplémentaire.

Comme nous l'avons vu, la personnalisation à l'extrême d'un WAYF est possible. Le seul motif politique qui peut faire décider de la place d'un WAYF est le nom DNS du serveur présentant le choix de l'IdP aux utilisateurs. L'expérience montre que l'ergonomie du WAYF est, aux yeux de nos décideurs, bien plus importante que son nom DNS.

3.6 Intégration dans le SI

Shibboleth a été conçu pour s'intégrer de la façon la plus transparente possible dans le SI d'un fournisseur d'identités ou de services. Ses briques logicielles gèrent les fonctionnalités de fédération d'identités en s'interfaçant avec les autres briques de gestion d'identités qui existent déjà dans un SI.

3.6.1 Intégration dans le SI d'un fournisseur d'identités

La brique logicielle Shibboleth pour un fournisseur d'identités est une application JAVA qui nécessite un s*ervlet container* (conteneur de servlets), tel Tomcat. Apache peut être utilisé ou non en frontal de Tomcat. Tout système d'exploitation sur lequel sont disponibles JAVA et Tomcat est utilisable ; c'est le cas notamment de Linux, Mac OS et Windows.

Shibboleth peut s'appuyer sur n'importe quel système d'authentification web déployé dans l'établissement : système d'authentification du serveur web (*mod_auth* d'Apache par exemple), SSO web (CAS par exemple), Kerberos, un CGI…

Pour récupérer des attributs sur les utilisateurs, Shibboleth peut s'interfacer avec un ou plusieurs référentiels simultanément, notamment des annuaires LDAP ou des bases SQL. Des connecteurs JDBC (*Java DataBase Connectivity*) et JNDI (*Java Naming and Directory Interface*) en extraient les attributs pour les rendre exportables sous forme SAML. Par exemple, pour propager les attributs d'un étudiant, il est possible d'extraire son identité (nom, prénom, courrier électronique) de l'annuaire LDAP de l'établissement et les cours auxquels il est inscrit d'une base SQL métier.

Nous présentons ci-après un exemple de définition des sources d'attributs d'un fournisseur d'identités. L'administrateur définit les sources de données (annuaire LDAP dans notre exemple), puis les attributs utilisateur en indiquant qu'ils dépendent de cette source. Dans la définition de la source de données, on fait référence à l'identifiant de l'utilisateur (`%PRINCIPAL%`), issu de la phase préalable d'authentification, dans un filtre LDAP ou une requête SQL.

```
<!-- Extrait de resolver.xml -->
<SimpleAttributeDefinition
    id="urn:mace:dir:attribute-def:mail">
  <DataConnectorDependency requires="supann-test"/>
</SimpleAttributeDefinition>
<SimpleAttributeDefinition
    id="urn:mace:dir:attribute-def:eduPersonPrincipalName"
    smartScope="univ-test.fr">
  <DataConnectorDependency requires="supann-test"/>
</SimpleAttributeDefinition>
<JNDIDirectoryDataConnector id="supann-test">
  <Search filter="cn=%PRINCIPAL%">
    <Controls searchScope="SUBTREE_SCOPE" returningObjects="false" />
  </Search>
  <Property
      name="java.naming.factory.initial"
      value="com.sun.jndi.ldap.LdapCtxFactory" />
  <Property
      name="java.naming.provider.url"
      value="ldap://ldap.univ-test.fr/dc=univ-test,dc=fr,ou=people" />
  <Property
      name="java.naming.security.protocol"
      value="ssl" />
  <Property
      name="java.naming.security.principal"
      value="cn=admin,dc=example,dc=edu" />
  <Property
      name="java.naming.security.credentials"
      value="examplepw" />
</JNDIDirectoryDataConnector>
```

3.6.2 Intégration dans le SI d'un fournisseur de services

La brique logicielle Shibboleth pour un fournisseur de services est disponible sous forme d'un module pour Apache ou pour IIS. Il permet de déléguer la phase d'authentification vers un fournisseur de services. Ce module peut aussi effectuer du contrôle d'accès sur la base des attributs propagés par les fournisseurs d'identités

L'application web offrant la ressource que l'on souhaite rendre accessible via Shibboleth doit être modifiée si l'on souhaite lui communiquer des attributs propagés via Shibboleth (à l'image d'une application que l'on « CAS-ifie »). Ces attributs sont présentés à l'application sous la forme d'en-têtes HTTP. Plusieurs applications ont déjà été rendues compatibles avec Shibboleth [25].

Une version JAVA de la brique fournisseur de services est déjà disponible. Fournie sous la forme d'un module Apache, cette implémentation est fonctionnellement très pauvre, puisque les règles de contrôle d'accès sont définies dans la configuration du serveur web. De gros progrès sont nécessaires dans ce domaine, par exemple la place un service dédié au contrôle d'accès, éventuellement à l'échelle d'un établissement. Ce service devrait permettre de déléguer cette tâche au plus près des producteurs de contenus publiés sur le web.

3.7 Contrôle de la diffusion/réception des attributs utilisateur

Les données transmises par un fournisseur d'identités à un fournisseur de services constituent un ensemble d'attributs utilisateur. Ces attributs utilisateur peuvent inclure des données nominatives (nom, prénom, identifiant, adresse email…) ou non (organisme de rattachement, catégorie d'usager, rôles…) selon le fournisseur de services et la relation contractuelle qui le lit au fournisseur d'identités. Avec la notion d'ARP (*Attribute Release Policy*), Shibboleth permet à un fournisseur d'identités de filtrer les attributs utilisateur propagés en fonction du fournisseur de services demandeur.

Une ARP peut contenir un ensemble de règles ; chaque règle définit un contexte d'application et spécifie les attributs qui peuvent ou non être divulgués. On peut définir les valeurs autorisées pour chaque attribut. On peut également spécifier des comportements par défaut pour tous les fournisseurs de services. Ci-dessous un extrait d'ARP illustrant la richesse d'expression : l'attribut `supannOrganisme` est délivré à tous les fournisseurs de services mais l'attribut mail n'est divulgué que dans le cadre de l'accès à la ressource `https://sp-univx.fr/ressource-y`. On peut ainsi filtrer les attributs nominatifs des utilisateurs dans le cadre de certains partenariats (accès à des fournisseurs de documentation électronique du secteur privé par exemple).

```
<Rule>
  <Target>
    <AnyTarget/>
  </Target>
  <Attribute name="urn:mace:cru.fr:attribute-def:supannOrganisme">
    <AnyValue release="permit"/>
  </Attribute>
</Rule>

<Rule>
  <Target>
    <Requester matchFunction="urn:mace:shibboleth:arp:matchFunction:exactShar">
      https://sp-univx.fr/ressource-y
    </Requester>
  </Target>
  <Attribute name="urn:mace:dir:attribute-def:mail">
    <AnyValue release="permit"/>
  </Attribute>
</Rule>
```

Une ARP spécifique peut également être définie pour chaque utilisateur, permettant un filtrage par l'utilisateur des données le concernant. Cependant les briques logicielles fournies avec Shibboleth n'incluent pas, pour l'instant, l'interface graphique requise pour permettre à l'utilisateur de définir ses propres règles de filtrage.

Il est important de noter que, au niveau d'un fournisseur d'identités, la politique de divulgation des attributs (via les ARP) est donc paramétrable indépendamment de la définition des sources de données pour ces attributs. Cette séparation peut faciliter le déploiement des ARP par défaut dans le contexte d'une fédération.

Un mécanisme équivalent aux ARP est disponible au niveau d'un fournisseur de services, permettant de filtrer les attributs reçus en provenance d'un fournisseur d'identités. Ce mécanisme, nommé AAP (*Attribute Acceptance Policy*), permet de filtrer les attributs reçus du fournisseur de services, en fonction de leur identifiant ou de leur valeur, avant qu'ils soient transmis à la ressource web qui les exploitera. On peut ainsi empêcher qu'un fournisseur d'identités fournisse des attributs semblant provenir d'un autre fournisseur d'identités. Exemple : l'IdP de l'université X fournissant un attribut `étudiant@univ-Y`.

3.8 Accès anonyme à un fournisseur de services

Shibboleth a été conçu pour permettre un accès authentifié et contrôlé à un fournisseur de services mais sans nécessairement diffuser des informations nominatives des utilisateurs à ce fournisseur.

Lors de la première phase de délégation d'authentification, le fournisseur de services ne récupère du fournisseur d'identités qu'un identifiant opaque et non persistant, le *nameIdentifier*. Cet identifiant lui permet d'obtenir ensuite des attributs sur l'utilisateur délivrés par le fournisseur d'identités. Ce dernier choisit quels attributs sont ainsi propagés, en fonction du fournisseur de services.

Parmi ces attributs certains peuvent être nominatifs (par exemple le nom ou l'adresse de courrier électronique) mais pas nécessairement : une application peut ainsi se contenter d'informations non nominatives pour contrôler l'accès. Par exemple un fournisseur commercial de ressources documentaires peut restreindre l'accès à son application aux seuls étudiants en troisième cycle de médecine de certaines universités. Dans ce cas il n'a pas besoin de connaître les identités des utilisateurs se connectant, mais juste leur étape de formation et leur université de rattachement.

Il n'empêche qu'un fournisseur de services peut souhaiter proposer des services personnalisés à ses utilisateurs se connectant de façon anonyme via Shibboleth, par exemple simplement leur offrir la possibilité de conserver des préférences d'affichage d'une session à l'autre. Dans ce cas, Shibboleth peut propager l'identifiant eduPersonTargetedId qui est un identifiant d'utilisateur unique, opaque et persistant partagé entre un fournisseur d'identités et un fournisseur de services. Il permet à un fournisseur de services d'avoir un identifiant anonyme pour un utilisateur. Un fournisseur d'identités choisit s'il diffuse cet attribut ou non. eduPersonTargetedID est construit automatiquement par Shibboleth, il n'a pas à être stocké dans un référentiel d'établissement.

3.9 Configuration technique des relations de confiance

Puisque la fédération d'identités permet à une entité de s'appuyer sur un service d'authentification hors de son domaine de sécurité, ce mécanisme a un besoin accru de sécurisation et d'authentification. Nous aborderons plus tard (cf 4.1) la formalisation des relations de confiance entre les membres d'une fédération.

3.9.1 Les méta-données

Ces méta-données sont composées d'une liste des membres de la fédération (fournisseurs d'identités et fournisseurs de services) d'une part et d'une liste des autorités de certification de confiance d'autre part. La liste des membres est un fichier XML dont chaque enregistrement fournit les informations requises pour chaque entité à savoir :

- L'identifiant du service, sous la forme d'un URN (*Uniform Resource Name*) ;
- L'intitulé du service ;
- Le contact technique pour le service ;
- L'URL des services correspondants : autorité d'authentification et autorité d'attributs pour un fournisseur d'identités, consommateur d'assertions pour un fournisseur de services.

L'exemple simplifié ci-dessous se limite à la définition d'un fournisseur d'identités et un fournisseur de services.

```
<EntitiesDescriptor
       Name="urn:mace:cru.fr:exemple-federation"
<EntityDescriptor entityID="https://idp.univ-exemple.fr/shibboleth">
      <Organization>
          <OrganizationName xml:lang="fr">Exemple de fournisseur
d'identités</OrganizationName>
          <OrganizationDisplayName xml:lang="fr">Université
d'exemple</OrganizationDisplayName>
          <OrganizationURL xml:lang="en">http://idp.univ-
exemple.fr/</OrganizationURL>
      </Organization>
      <ContactPerson contactType="technical">
          <SurName>Support technique</SurName>
```

```
            <EmailAddress>support@idp.univ-exemple.fr</EmailAddress>
      </ContactPerson>
      <SingleSignOnService
      Binding="urn:mace:shibboleth:1.0:profiles:AuthnRequest"
      Location="https://idp.example.org/shibboleth-idp/SSO"/>
      <AttributeService
      Binding="urn:oasis:names:tc:SAML:1.0:bindings:SOAP-binding"
      Location="https://idp.univ-exemple.fr:8443/shibboleth-idp/AA"/>
</EntityDescriptor>
<EntityDescriptor entityID="https://sp.example.org/shibboleth">
      <Organization>
            <OrganizationName xml:lang="en">Example Service
Provider</OrganizationName>
            <OrganizationDisplayName xml:lang="en">Services 'R'
Us</OrganizationDisplayName>
            <OrganizationURL
xml:lang="en">http://sp.example.org/</OrganizationURL>
            </Organization>
            <ContactPerson contactType="technical">
                  <SurName>Technical Support</SurName>
                  <EmailAddress>support@sp.example.org</EmailAddress>
            </ContactPerson>
      <AssertionConsumerService
      index="1" isDefault="true"
      Binding="urn:oasis:names:tc:SAML:1.0:profiles:browser-post"
      Location="https://sp.example.org/Shibboleth.sso/SAML/POST"/>
      </EntityDescriptor>
</EntitiesDescriptor>
```

Les méta-données sont gérées de façon centralisée pour la fédération et partagées par tous les sites participants qui doivent mettre en place une tâche périodique de synchronisation. Chaque site peut enrichir ses méta-données s'il participe à plusieurs fédérations ou pour les besoins de relations bilatérales. Ces méta données n'obligent pas un site à avoir des relations avec tous les autres sites qui y sont listés : chaque site est libre de définit avec quels partenaires il travaille

3.9.2 Intégrité et authentification des assertions SAML

Suite à la phase d'authentification de l'utilisateur, une assertion SAML d'authentification est délivrée par le fournisseur d'identités au fournisseur de services. Cette assertion est signée par le fournisseur d'identités (utilisation d'un certificat X509) afin de permettre ultérieurement au fournisseur de services d'en vérifier à la fois la provenance et l'intégrité. L'autorité de certification ayant délivré le certificat du fournisseur d'identités doit être reconnue par le fournisseur de services. La signature de l'assertion est essentielle car le fournisseur de services n'est pas en

contact direct avec le fournisseur d'identités pendant cette phase (la transmission se fait par l'intermédiaire du navigateur web, cf Figure 4).

Le fournisseur de services ne se contente pas d'authentifier l'émetteur de l'assertion, il vérifie également que cet émetteur est référencé dans les méta-données référençant les fournisseurs d'identités de confiance.

3.9.3 Authentification lors de l'accès aux attributs utilisateur

Suite à la phase d'authentification, le fournisseur de services contacte directement (via une requête SOAP) l'autorité d'attributs du fournisseur d'identités pour obtenir les attributs de l'utilisateur. Le fournisseur de services exploite les méta-données de la fédération pour connaître l'URL de l'autorité d'attributs. Afin que l'autorité d'attributs puisse l'identifier, le fournisseur de services doit s'authentifier au moyen d'un certificat client X509. Le certificat ou son autorité de certification doivent être référencées dans les méta-données ; le CN (*Common Name*) du certificat est également référencé dans les méta-données.

4 La fédération pilote du CRU, illustration de la mise en place d'une fédération d'identités

Au travers de sa participation à Internet2 [19] et à TERENA [26], le CRU [27] a suivi les projets de fédération d'identités menés notamment aux Etats-Unis [28][29] et en Suisse [30] depuis 2002. Les retours d'expérience étaient positifs et les besoins satisfaits étaient les mêmes que ceux qui émergent actuellement au sein de notre communauté sur l'authentification et le contrôle d'accès à des applications dont les utilisateurs sont répartis dans plusieurs établissements. Aussi fin 2004 le CRU a étudié différentes solutions techniques pour répondre à ces besoins, dont Liberty Alliance et Shibboleth. Shibboleth a été retenu, et depuis avril 2005 le CRU propose un service de propagation d'identités et d'attributs nommé « **la fédération pilote du CRU** » [31].

Si Shibboleth est désormais éprouvé et satisfait des besoins importants, cela ne résout que la problématique technique de la mise en œuvre d'une fédération. La mise en place d'une fédération opérationnelle soulève en effet d'autres questions que nous abordons ici. Leur discussion va nous permettre notamment de définir précisément ce qu'est une fédération et de présenter la fédération pilote du CRU. Pour faciliter la lecture, nous définissons pour le moment une fédération simplement comme « un regroupement de fournisseurs de services et de fournisseurs d'identités ».

4.1 Quelles relations de confiance entre les membres d'une fédération ?

Cette question est cruciale, autant pour les fournisseurs d'identités que pour les fournisseurs de services :

- Un fournisseur de services se repose sur les fournisseurs d'identités pour assurer une authentification sûre de ses utilisateurs. Il leur fait également confiance sur la qualité des attributs utilisateurs qu'ils propagent. Un fournisseur de services dépend aussi fortement de la disponibilité des services d'authentification et de propagation d'attributs des fournisseurs d'identités.
- Un fournisseur d'identités délivre des attributs sur ses utilisateurs à des fournisseurs de services. Il leur fait confiance pour que ces derniers les utilisent uniquement pour les usages prévus.

Toutes ces relations de confiance doivent être formalisées pour offrir à des fournisseurs d'identités ou de services souhaitant rejoindre une fédération la connaissance claire d'un niveau de confiance minimale sur laquelle ils peuvent se reposer. Cette formalisation peut être plus ou moins forte, et donc revêtir différentes formes : simple engagement sur le respect de bonnes pratiques, signature d'une convention, engagement contractuel avec régime de responsabilités, etc.

Voici quelques exemples d'engagements possibles pour un fournisseur d'identités :

- Disponibilité et sécurisation du service d'authentification ;
- Bon approvisionnement du référentiel, mise à jour (par exemple suppression rapide des comptes de utilisateurs qui ne sont plus rattachés à l'établissement) ;
- Propagation des attributs de ses utilisateurs dans un nommage et une sémantique communs (cf 4.2.1).

On constate immédiatement la nécessité d'un acteur définissant ces engagements et centralisant leur gestion, afin d'éviter la multiplication des relations bilatérales entre les différents fournisseurs de la fédération. Son rôle est également d'opérer tous les services centraux de la fédération : distribution des méta données, exploitation d'un éventuel WAYF central. Il doit aussi respecter des engagements car son action a aussi une influence sur la confiance au sein de la fédération. Cet acteur peut être vu comme le « conseil d'administration » de la fédération ou son « opérateur » ou même être désigné comme « la fédération », selon le modèle retenu.

Le dernier point montre qu'outre la formalisation des relations de confiance, une fédération doit définir son statut administratif, les services offerts, son périmètre, les modalités d'entrée et de sortie de la fédération, etc.

4.1.1 SWITCHaai, un exemple de fédération académique

SWITCHaai [30] est la fédération académique suisse, opérée par SWITCH [32], l'opérateur du réseau académique suisse, aai signifiant *Authentication, Authorization Infrastructure*. Elle existe depuis 2003 et regroupe des fournisseurs d'identités (désignés comme *Home Organizations*) et des fournisseurs de services (les *Ressources Owners*). En octobre 2005, SWITCHaai comptait 10 fournisseurs d'identités [33]. Environ 10000 utilisateurs actifs accèdent à des ressources via cette fédération :

- Des **plateformes d'enseignement à distance** comme OLAT (*Online Learning And Training*), WebCT (*Web Course Tools*) et Moddle ;
- Des **fournisseurs de documentation électronique** comme ScienceDirect et SpringerLink, via un proxy (EZproxy) intégrant la technologie Shibboleth.

Deux comités, nommés *Operations Committee* et *Advisory Committee*, représentent les membres de la fédération, et ont un rôle de conseil.

- Le premier est consulté pour les décisions à court terme ainsi que pour des aspects opérationnels et techniques (définition des meilleures pratiques liées à la fédération d'identités, sécurité, définition des attributs, calendrier des migrations logicielles..) ;
- Le second est consulté pour la gestion stratégique à long terme de la fédération et sur les projets inter institutionnels qui y sont liés (financements, définition des politiques, accréditation des autorités de certification utilisables...).

SWITCH coordonne les actions de ces deux comités et assure la promotion et le développement de la fédération. De plus il opère le WAYF central de la fédération[5], assiste les fournisseurs pour le déploiement et l'utilisation de Shibboleth, gère un registre des membres à la fédération et des méta-données qui y sont associées.

L'adhésion d'un fournisseur d'identités ou de services à la fédération est actée par la signature d'un protocole d'accord entre le fournisseur et SWITCH (*Service Agreement* [34]).

Les fournisseurs d'identités doivent pouvoir fournir un jeu minimal d'attributs (*Authorization Attribute Specification* [35]). Ils ont l'obligation de communiquer leurs procédures d'enregistrement et d'authentification de leurs utilisateurs à un autre membre de la fédération si ce dernier le demande. Ils doivent également conserver les journaux d'authentification de leurs utilisateurs sur une durée d'au moins six mois.

Les fournisseurs de service déclarent à SWITCH les attributs dont ils ont besoin. Ils doivent conserver leurs journaux de connexion pour une durée d'au moins six mois, et doivent les communiquer à un fournisseur d'identités dans le cas d'un abus d'utilisation de la part d'un utilisateur.

Tous les membres de la fédération doivent utiliser des certificats émis par les autorités de certification accrédités au sein de la fédération (*CA Acceptance Policy* [36]).

SWITCH fournit une clause type à annexer à une charte d'établissement sur l'utilisation des ressources informatiques. Elle permet à un utilisateur d'autoriser la propagation d'attributs le concernant et limite la responsabilité des membres de la fédération en cas de préjudices consécutifs à l'utilisation des services de la fédération.

[5] Des WAYF dédiés à des fournisseurs de services peuvent néanmoins co-exister.

4.1.2 La fédération pilote du CRU

La fédération pilote du CRU est un service destiné aux applications web (« fournisseur de services ») dont les utilisateurs sont issus d'établissements différents. Elle permet à une telle application :

1. de déléguer la phase d'authentification aux établissements de rattachement des utilisateurs (service de délégation d'authentification) ;
2. éventuellement de récupérer de la part de ces établissements des attributs décrivant ces utilisateurs (service de propagation d'attributs).

Formellement, la fédération pilote du CRU est « un ensemble de fournisseurs d'identités capables de propager avec un certain niveau de confiance des assertions d'authentification et des attributs utilisateur avec un nommage standard ». Autrement dit c'est un regroupement d'établissements qui se sont engagés sur le respect de bonnes pratiques concernant la gestion des identités de leurs utilisateurs, et aptes techniquement à les authentifier pour le compte d'applications externes et éventuellement fournir à ces dernières des attributs décrivant leurs utilisateurs. Les bonnes pratiques de gestion des identités font référence à la complétude des référentiels décrivant des utilisateurs rattachés à l'établissement, leur mise à jour régulière, la fiabilité des attributs qui y sont contenus, la sécurité de ces référentiels, la sécurité du service d'authentification de l'établissement, etc.

Un fournisseur d'identités de la fédération pilote du CRU s'engage à respecter ces bonnes pratiques via la signature d'une convention d'adhésion à la fédération pilote.

Chaque fournisseur, d'identités ou de services, choisit les fournisseurs avec lesquels il travaille :

- Un fournisseur de services choisit les fournisseurs d'identités dont les utilisateurs peuvent accéder à ses ressources ;
- Un fournisseur d'identités choisit quels fournisseurs de services auront accès aux données de ces utilisateurs.

Les documents administratifs (convention, règles d'usages, définition de la fédération pilote) sont en cours de rédaction et seront disponibles prochainement [31].

La fédération pilote du CRU se limite à un service technique de propagation d'identités et d'attributs et ne couvre aucun des autres aspects qui peuvent régir la relation entre un fournisseur de services et des fournisseurs d'identités, par exemple les aspects fonctionnels, financiers, administratifs, etc. L'utilisation de la fédération pilote du CRU peut être mentionnée dans un contrat qui lie un ou plusieurs établissements à un fournisseur de services, mais la fédération pilote n'est pas partie signataire du contrat.

Au sein de la fédération, les rôles du CRU sont de :

- Définir les règles et bonnes pratiques qui régissent les relations entre les membres de la fédération ;
- Assister et conseiller les établissements participant à la fédération pour le déploiement des composants logiciels nécessaires au service et leur intégration dans le système d'information ;
- Définir le nommage et la sémantique des attributs propageables au sein de la fédération ;
- Opérer le service de découverte de la fédération.

Le CRU opère également une fédération de test pour évaluer Shibboleth et aider à son installation et sa configuration.

4.2 Quelle sémantique pour les attributs utilisateurs ?

4.2.1 Le besoin d'un nommage et d'une sémantique communs d'attributs

Un fournisseur de services peut recevoir de la part de fournisseurs d'identités des attributs décrivant les utilisateurs. Il est nécessaire pour un fournisseur de services qu'un attribut soit propagé sous le même nom par tous les fournisseurs d'identités. Sinon le fournisseur de services devrait interpréter chaque attribut en fonction du fournisseur d'identités, ce qui n'est pas viable. Au sein d'une fédération il faut donc un nommage commun dans lequel tous les fournisseurs d'identités sont capables de propager les attributs de leurs utilisateurs.

Le ou les nommages d'attributs d'un fournisseur d'identités peuvent différer du nommage de la fédération. Avant de rejoindre une fédération, un fournisseur d'identités doit adapter son nommage ou implémenter une correspondance dynamique entre son nommage et celui utilisé au sein de la fédération[6].

Indirectement le nommage d'attribut d'une fédération sert aux fournisseurs de services à connaître tous les attributs qu'ils peuvent potentiellement obtenir des fournisseurs d'identités. En pratique la constitution et l'évolution du nommage suivent généralement les besoins exprimés par les fournisseurs de services.

Les attributs récupérés par un fournisseur de services peuvent lui permettre de personnaliser le service, mais également d'en contrôler l'accès. Par exemple une plate-forme d'enseignement à distance pour les étudiants en médecine contrôlera l'accès en vérifiant que la valeur de l'attribut *discipline* de l'étudiant corresponde à une formation en médecine. Si les différents établissements renseignent cet attribut avec autant de nomenclatures différentes, la plate-forme doit lister toutes ces nomenclatures et suivre leurs évolutions, ce qui n'est pas viable. Aussi, outre la définition d'un nommage commun d'attributs, une fédération doit également définir la sémantique précise de ces attributs et les nomenclatures associées.

4.2.2 Les attributs au sein de la fédération pilote du CRU

La fédération pilote du CRU utilise SupAnn [2] comme base de son nommage d'attributs. SupAnn est le schéma d'annuaire préconisé pour l'enseignement supérieur et propose des attributs standardisés pour décrire un personnel ou un étudiant. Cependant il ne couvre pas tous les usages pressentis (au moins dans sa version actuelle), notamment les formations suivies par les étudiants. Le nommage de la fédération pilote du CRU sera donc étendu au-delà de SupAnn pour satisfaire les besoins exprimés par les fournisseurs de services.

Chaque attribut est doté d'un URN (*Uniform Resource Name*) qui est un identifiant global, unique et persistant. Ils sont issus des espaces de nommage `urn:mace:attribute-def` et `urn:mace:cru.fr:attribute-def` [37]. Si un fournisseur d'identités souhaite que ses utilisateurs puissent accéder à un fournisseur de services avec lequel il a passé un accord, il doit être capable de propager les attributs requis dans le nommage de la fédération pilote. Dans la fédération pilote du CRU, aucun attribut n'est diffusé par défaut, c'est-à-dire sans le consentement explicite des fournisseurs d'identités. Il est possible de définir des attributs utilisables pour un sous-ensemble de fournisseurs d'identités et de services, par exemple dans le cadre d'une UNR.

[6] Shibboleth permet d'implémenter des mécanismes de translation du nommage des attributs entre le(s) référentiel(s) de l'établissement et le nommage de la fédération.

4.3 Déclarations CNIL et respect de la vie privée des utilisateurs

4.3.1 Déclarations CNIL

Shibboleth permet la propagation dynamique d'attributs décrivant un utilisateur, hors de l'établissement de rattachement de l'utilisateur. Cette propagation se fait sous le contrôle de l'établissement qui définit quels attributs il s'autorise à diffuser, ce en fonction de chaque fournisseur de services. La diffusion du contenu d'un référentiel d'établissement est un usage du référentiel qui peut être nouveau par rapport à ses usages initiaux. Dans ce cas il est nécessaire d'adapter et refaire valider la déclaration CNIL concernant ce référentiel.

Notons que cette propagation d'attributs n'est effectuée que lorsqu'un utilisateur accède à un service, jamais à un autre moment, et que cette propagation est limitée aux attributs strictement nécessaires à l'application. En particulier Shibboleth offre un mécanisme d'anonymisation qui permet de délivrer des attributs décrivant un utilisateur sans dévoiler son identité si cela n'est pas nécessaire. C'est donc une exportation des attributs qui est beaucoup plus contrôlée que celle effectuée par exemple lors de la constitution d'un méta annuaire partagé entre plusieurs établissements.

Le CRU est actuellement en contact avec la CNIL pour valider ce procédé de propagation d'attribut contrôlé, et proposer une formulation type à annexer à une déclaration CNIL de référentiel d'établissement.

4.3.2 Respect de la vie privée des utilisateurs

La réglementation européenne offre à ses citoyens des droits quant au respect de leur vie privée et le partage d'informations les concernant entre des partenaires (publics ou privés) [38]. Notamment ils doivent être avertis dès qu'une information les concernant est susceptible d'être communiquée à un tiers, et ont le droit de le refuser, quitte à ne pas pouvoir utiliser un service nécessitant cette communication d'information. En théorie, un fournisseur de services devrait donc alerter un utilisateur avant sa connexion que l'accès est conditionné par le fait que certains attributs le concernant vont lui être communiqués par son établissement de rattachement. C'est une démarche respectueuse de la réglementation, mais qui ergonomiquement peut être rebutante à la longue.

Dans le cadre de sa fédération d'identités académique basée sur Shibboleth, la Finlande développe actuellement une extension à la brique logicielle du fournisseur d'identités pour permettre à un utilisateur de gérer lui-même les attributs le concernant pouvant être diffusés. Cette extension permettra le respect de la réglementation européenne tout en en réduisant l'impact ergonomique.

5 Perspectives

Il serait bien imprudent de tirer dès maintenant des conclusions, aussi préférons nous présenter ici des perspectives concernant Shibboleth :

– Un projet de délégation entre fournisseurs de services ;

– Le concept de fournisseur d'identités virtuel.

Nous montrons finalement les projets en cours et premiers retours d'expérience.

5.1 La problématique de délégation

Des évolutions de Shibboleth sont envisagées pour permettre de s'adapter à des architectures multi-tiers.

Dans ce type d'architectures, l'utilisateur contacte un fournisseur de services qui lui-même fait appel à un second fournisseur de services (*backend SP*). Comme l'utilisateur n'est pas directement en contact avec le second fournisseur de services, le premier fournisseur de services joue le rôle d'intermédiaire entre le fournisseur d'identités et le fournisseur de service final. Les assertions SAML délivrées par le fournisseur d'identités pourront être chiffrées à destination du second fournisseur de services, afin d'en assurer la confidentialité (vis-à-vis du premier fournisseur de services).

Ces évolutions impliquent une extension des spécifications SAML, actuellement à l'étude [39]. Remarquons néanmoins que la prise en compte des architectures multi-tiers paraît a priori plus simple que celle qui est mise en œuvre dans CAS et décrite dans [23][24]. En effet, l'utilisation des certificats serveur des acteurs de Shibboleth fait que le fournisseur de service n'a pas besoin d'être en lien direct avec le fournisseur d'identités (cf Figure 25), simplifiant ainsi le nombre d'interactions et le protocole.

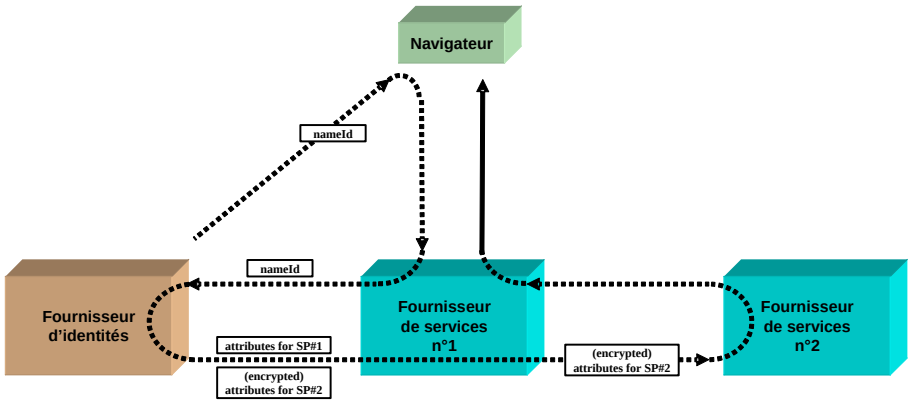

Figure 25 *Architecture simplifiée de la délégation avec Shibboleth*

Ces mécanismes de délégation peuvent trouver leur application par exemple dans des portails documentaires agrégeant des contenus issus de différentes sources et nécessitant un contrôle d'accès. La Figure 26 montre un tel exemple.

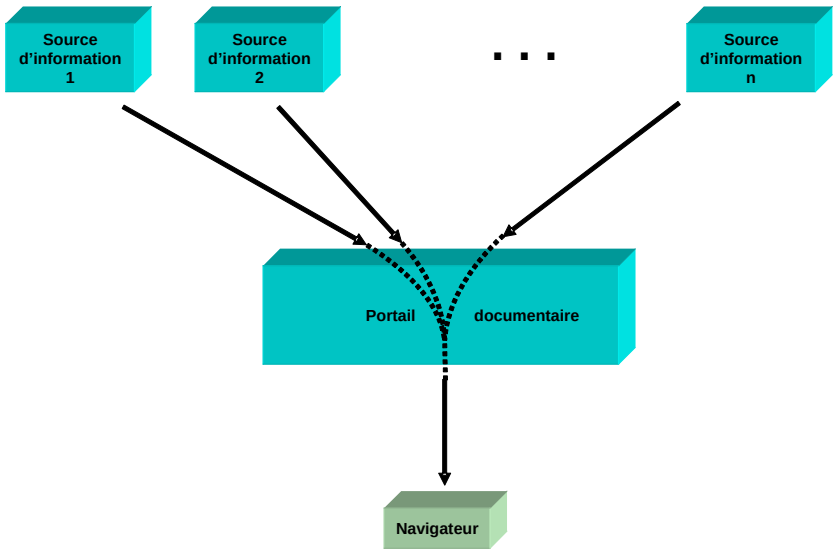

Figure 26 *Exemple d'utilisation possible de la délégation dans Shibboleth*

5.2 Fournisseur d'identités virtuel

Une fédération s'agrandit progressivement, au fur et à mesure que des fournisseurs d'identités la rejoignent et que des services sont proposés. À un moment donné, il se peut donc qu'une partie des utilisateurs d'une application ne soient pas inscrits dans un fournisseur d'identités de la fédération. Soit parce que leur établissement de rattachement n'est pas (encore) fournisseur d'identités, soit par ces utilisateurs n'ont pas d'établissement de rattachement qui puisse être fournisseur d'identités.

Dans le cas où le gestionnaire d'une application souhaite gérer l'accès de ces utilisateurs via la fédération d'identités (comme pour les autres utilisateurs), il existe une solution pour « faire comme si » ces utilisateurs appartenaient à un fournisseur d'identités : c'est le fournisseur d'identités virtuel. Il s'agit d'une application tierce, centrale au sein d'une fédération, offrant une interface d'administration qui permet d'enregistrer des utilisateurs, leur allouer un compte (identifiant et mot de passe) pour leur permettre de s'y authentifier. Elle intègre également la brique logicielle Shibboleth IdP, et peut donc authentifier les utilisateurs inscrits pour le compte d'autres applications, comme un fournisseur d'identités classique.

Ce fournisseur d'identités virtuel peut constituer une brique de base d'une solution plus globale de gestion de groupes virtuels (*Virtual Organization*) en y intégrant des mécanismes de gestion d'attributs, de délégation de gestion de comptes, d'outils de communication...

Références

[1] *SDET : Schéma Directeur des Espaces numériques de Travail*, www.educnet.education.fr/equip/sdet.htm.

[2] *Annexe SupAnn du SDET, recommandations pour les annuaires de l'enseignement supérieur*, www.educnet.education.fr/chrgt/SUPANN-V10.pdf.

[3] *Annexe AAS du SDET, recommandations pour l'authentification, les autorisations et le Single Sign-On* www.educnet.education.fr/chrgt/AAS-V10.pdf.

[4] *Les Universités Numériques en Région*, www.educnet.education.fr/superieur/unr.htm.

[5] *RFC2616: Hypertext Transfer Protocol – HTTP/1.1*, www.w3.org/Protocols/rfc2616/rfc2616.html.

[6] *Persistent client state, HTTP cookies*, www.netscape.com/newsref/std/cookie_spec.html.

[7] Olivier Salaün, *Introduction aux architectures web de Single Sign-On*. Dans Actes de la conférence JRES2003, Lille, France, décembre 2003, 2003.jres.org/actes/paper.116.pdf.

[8] Consortium ESUP-Portail, ESUP-Portail, Espace Numérique de Travail d'accès intégré aux services pour les étudiants et le personnel de l'enseignement supérieur, www.esup-portail.org.

[9] uPortal by JASIG, *Evolving portal implementations from participating universities and partners*, www.uportal.org.

[10] *OASIS Security Services (SAML)*, www.oasis-open.org/committees/tc_home.php?wg_abbrev=security.

[11] OASIS, *development, convergence and adoption of e-business standards*, http://www.oasis-open.org.

[12] Serge Aumont, Claude Gross, Philippe Leca, *Certificats X509 et infrastructure de gestion de clés*. Dans Actes de la conférence JRES2001, Lyon, France, décembre 2001, www.cru.fr/igc/JRES01.tutoriel.IGC.pdf.

[13] Stéphane Bortzmeyer, *Les Web Services*. Dans actes de la conférence JRES2003, Lille, France, décembre 2003, 2003.jres.org/TUTORIELS/paper.C.pdf.

[14] WS-I, *Web Services Interoperability Organization*, www.ws-i.org.

[15] *WS-Security specifications*, schemas.xmlsoap.org/specs/ws-security/ws-security.htm.

[16] *WS-Federation public draft, msdn.microsoft.com/library/default.asp?url=/library/en-us/dnglobspec/html/ws-federation.asp*

[17] *The Shibboleth Project*, shibboleth.internet2.edu.

[18] The Liberty Alliance Project, www.projectliberty.org.

[19] Internet2, advanced network applications and technologies for research and higher education, www.internet2.edu.

[20] *ID-FF, Identity Federation Framework*, www.service-architecture.com/web-services/articles/identity_federation_framework_id-ff.html.

[21] *ID-WSF, Identity Web Services Framework*, www.service-architecture.com/web-services/articles/identity_web_services_framework_id-wsf.html.

[22] Nathan Dors, *Shibboleth architecture, technical overview*, working draft, shibboleth.internet2.edu/docs/draft-mace-shibboleth-tech-overview-latest.pdf.

[23] Vincent Mathieu, Pascal Aubry et Julien Marchal, *Single Sign-On open-source avec CAS (Central Authentication Service)*. Dans actes du congrès JRES2003, Lille, décembre 2003, 2003.jres.org/actes/paper.139.pdf.

[24] Pascal Aubry, Vincent Mathieu et Julien Marchal, *ESUP-Portail: open-source Single Sign-On with CAS (Central Authentication Service)*. Dans Actes de EUNIS2004, Bled, Slovénie, juin 2004, perso.univ-rennes1.fr/pascal.aubry/presentations/cas-eunis2004.

[25] *Shibboleth enabled applications and services*, shibboleth.internet2.edu/seas.html.

[26] TERENA, Trans-European Research and Education Networking Association, www.terena.nl.

[27] *CRU, Comité Réseau des Universités*, www.cru.fr.

[28] The InQueue federation, operated by internet2, *http://inqueue.internet2.edu*.

[29] *InCommon makes sharing protected inline resources easier*, www.incommonfederation.org/.

[30] *SWITCH Authentication and Authorization Infrastructure (AAI)*, www.switch.ch/aai/.

[31] Fédération pilote du CRU, service de propagation d'identités et d'attributs, federation.cru.fr.

[32] *SWITCH, the Swiss Education & Research Network*, www.switch.ch.

[33] *SWITCHaai Federation Members*, www.switch.ch/aai/agreement/signers.html.

[34] *SWITCHaai Service Agreement*, www.switch.ch/aai/agreement/index.html.

[35] *SWITCHaai Authorization Attribute Specification*, www.switch.ch/aai/docs/AAI_Attr_Specs.pdf.

[36] *SWITCHaai CA Acceptance Policy*, http://www.switch.ch/aai/ca-acceptance-policy.html.

[37] Espace de nommage `urn:mace:cru.fr`, www.cru.fr/urn/.

[38] Directive 2002/58/CE du Parlement européen et du Conseil du 12 juillet 2002 concernant le traitement des données à caractère personnel et la protection de la vie privée dans le secteur des communications électroniques (directive vie privée et communications électroniques), europa.eu.int/smartapi/cgi/sga_doc?smartapi!celexapi!prod!CELEXnumdoc&lg=FR&numdoc=32002L0058&model=guichett

[39] Scott Cantor, *SAML 2.0 Single Sign-On with Constrained Delegatio*n, working draft, shibboleth.internet2.edu/docs/draft-cantor-saml-sso-delegation-01.pdf

Liste des figures

Sigles utilisés

AAP	*Attribute Acceptance Policy* (politique d'acceptation des attributs, cf 3.7)
AAS	Authentification - autorisation – SSO [3]
ARP	*Attribute Release Policy* (politique de diffusion des attributs, cf 3.7)
CA	*Certificate Authority* (autorité de certification)
CAS	*Central Authentication Service* [23][24]
CGI	*Common Gateway Interface* (hoohoo.ncsa.uiuc.edu/cgi)
CN	*Common Name*
CNIL	Commission Nationale « Informatique et Liberté » (www.cnil.fr)
CRU	Comité réseau des universités [27]
DNS	*Domain Name System* (système de noms de domaine)
ENT	Espaces numériques de travail [1]
HTTP	*HyperText Transfer Protocol* [5]
IdP	Identity Provider (fourniseeur d'identités, cf 3.1.3 et 3.2.4)
IGC	Infrastructure de gestion de clés publiques [12]
J2EE	*Java 2 Platform, Enterprise Edition* (java.sun.com/j2ee)
JDBC	*Java DataBase Connectivity* (java.sun.com/products/jdbc)
JNDI	*Java Naming and Directory Interface* (java.sun.com/products/jndi)
LDAP	*Lightweight Directory Access Protocol*
OASIS	*Organization for the Advancement of Structured Standards* (www.oasis-open.org)
PKI	*Public Key Infrastructure* (Infrastructure de Gestion de Clés publiques) [12]
SAML	*Security Assertion Markup Language*
SDET	Schéma Directeur des Espaces numériques de Travail [1]
SI	Système d'Information
SOAP	*Simple Object Access Protocole* [13]
SP	*Service Provider* (fournisseur de services, cf 3.1.2 et 3.2.3)
SQL	*Structured Query Language*
SSL	*Secure Sockets Layers*
SSO	*Single Sign-On* (équivalent français : authentification unique) [7]
TERENA	*Trans European Research and Education Networking Association* [26]
TGC	*Ticket Granting Cookie* [23][24]
UNR	Université Numérique en Région [4]
URL	*Uniform Resource Locator*
URN	*Uniform Resource Name*

XML	*eXtended Markup Language*
WAYF	*Where Are You From* (cf 3.1.4 et 3.4)