認証フェデレーションにおけるIdPの属性制御

阿部, 英司

伊東, 栄典

笠原, 義晃

出版情報：電子情報通信学会2010年総合大会講演論文集. B, pp.S133-S134, 2010-03-19. 電子情報通信学会
バージョン：
権利関係：

認証フェデレーションにおける IdP の属性制御

Attribute control in IdP for federated web services

阿部 英司[*1]　伊東 栄典[*2]　笠原 義晃[*2]

[*1] 九州大学システム情報科学府情報理学専攻 〒812-8581 福岡市東区箱崎 6-10-1

[*2] 九州大学情報基盤研究開発センター 〒812-8581 福岡市東区箱崎 6-10-1

1. はじめに

近年 Web 上に多くの認証を要する情報サービスが公開されている。認証を要する情報サービスの増大に伴い，利用者における認証操作の煩雑さが問題になってきた。この煩雑さを軽減するために，利用者側からサービス連携の要求が高まってきている。そこで，一度の認証で複数のサービスを利用可能にする SSO（Single Sign-On）が研究開発されている。Google Apps の様に，同一組織が提供する Web 上の情報サービスでは，提供者側がサービスの効率化のために SSO の仕組みを構築している。

組織間で認証連携実現する方法として，SAML[1]，Shibboleth[2]などの技術がある。これらの技術の発達によりフェデレーションとよばれる共通ポリシーに基づく相互運用に同意した組織間連携が構築されるようになった。フェデレーション内の組織では相互に情報提供やサービス連携を行う。

我々は組織間認証連携についての研究を行ってきた[3]。本稿では，組織認証連携である Shibboleth について述べ，それを利用したフェデレーションについても述べる。日本におけるフェデレーションの属性に関する問題とその解決手法の提案を行い，九州大学における Shibboleth IdP の実装・評価・今後の課題についても述べる。

2. Shibboleth

米国の Internet2 プロジェクトでは、Shibboleth と名付けられた SSO，ID 連携による利用者認証・認可の基盤の提案および実装を行っている。Shibboleth における認証・認可情報の記述形式は、SAML 形式である。また Shibboleth では組織間での認証連携のために、WAYF（Where Are You From ?）と名づけられた仕組みを導入している。図 1 に Shibboleth の概念図を示す。WAYF を用いる場合、利用者は自分が所属する組織を指定する。これにより、どの組織の IdP を用いるのかを特定でき、その IdP で利用者認証を行う。

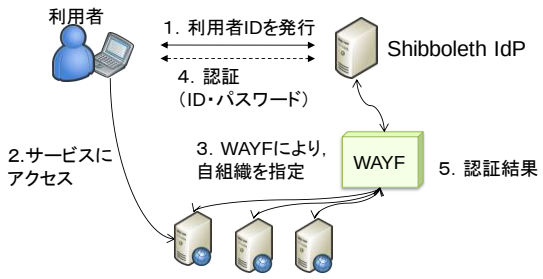

図 1　Shibboleth 概念図

3. UPKI-Fed

複数組織が共通ポリシーのもとで相互運用するフェデレーションが世界各国で構築されている。日本では国立情報学研究所を中心に「学術認証フェデレーション（愛称：UPKI-Fed: Japanese Academic Federation）」が発足し，九州大学を含む多くの大学が参画している。技術的には Shibboleth を用いて認証連携を実現している。

UPKI-Fed では IdP 側，SP 側，利用者それぞれにメリットがある。利用者側は自分の属する組織で認証を行うことができ，個人情報の露出機会を大幅に削減することができる。SP 側としても ID 管理や，利用者情報の管理から開放され，IdP 側は大学などの情報セキュリティ準拠，コンプライアンスを順守することができる。

しかし UPKI-Fed には属性交換の問題が 2 点残っている。1 つ目の問題点は，今までは各大学のポリシーに基づき使用していた属性を UPKI-Fed が推奨している属性へと変更しなければいけない点である。UPKI-Fed の推奨属性と大学のポリシーで定めた属性が異なる場合，LDAP などの属性管理システムや従来のサービスに変更を加えなければならない。2 つ目の問題点は，SP ごとに適切な属性提供をしなければいけない点である。組織内の SP ではどのように属性値を扱うか知ることができるが，組織外の SP ではどのように属性値を扱うか SP を運営する組織に委ねることとなる。そこで個人情報保護の観点から必要以上に外部組織に利用者情報を提供しないために，SP ごとにどの属性値を提供するか判断する方法が必要となっている。

4. 属性加工・提供方法の検討

4.1 属性加工の方法

各大学が利用者に提供している既存のサービスに大幅な変更を加えること無く，Shibboleth IdP の提供属性を UPKI-Fed の推奨属性に適応させる手法について 3 つの方法を検討した。

1 つ目の手法は，既存の属性情報に UPKI-Fed で推奨されている属性と属性値を追加する方法である。使用されていない属性の場合比較的容易に属性値追加は可能であるが，現在使われている属性値に違う値を入れなければいけない場合は組織内サービスを変更しなければならない。また，組織外に属性管理を外注している場合は変更に必要となる金銭的コストも問題になる。

2 つ目の手法は Shibboleth IdP と属性管理システムの間に LDAP proxy を介し，OpenLDAP のモジュール等で属性加工を行う方法である。これは既存の属性管理システムとサービスに変更を加える必要はない。しかし，新たに LDAP proxy を構築する煩雑さと運用するサーバ数が増加による管理側のコストの問題が発生する。

3 つ目の手法は，Shibboleth attribute resolver を利用する方法である。Shibboleth では属性値の静的追加やスクリプトによる動的追加が可能である。既存システムの変更や新たなサーバの管理コストは発生せず，他の手法と比べコストを抑えることができる。しかし，設定ファイルが XACML（eXtensible Access Control Markup Language）で書かれているため設定記述が煩雑である。

4.2 属性提供の方法

必要以上に外部 SP に属性値を提供しないために属性提供判断方法として 3 つの手法を検討した。

1 つ目の手法は組織内 SP 用の IdP と外部 SP 用の 2 つの IdP を構築する方法である[4]。外部 SP 用の IdP には必要最低限の属性しか提供しないように記述することで，外部に必要以上の属性値を提供することを避けることができる。

2 つ目の方法は，LDAP 側で属性をマッピングする方法である。どの SP が認証要求を行ったか判断し，動的に属性マッピングを行うことで，外部 SP に属性値を提供することを避ける。既存属性管理システムに変更を加えることが難しいなら LDAP proxy を仲介してもよい。この場合 LDAP 側がどの SP からアクセスしてきたかを判断する必要があり，そのための仕組みが必要となる。

3 つ目の手法は Shibboleth attribute filter による属性ごとにどの SP に属性を提供するかを記述する方法である。属性ごとにどの SP に提供するか記述する。しかし Shibboleth attribute resolver の場合と同様に記述が煩雑であり，SP 数が増加するとその煩雑さも増大する。

5. 九州大学における Shibboleth IdP の構築

九州大学では LDAP によって組織内利用者の情報を管理しており，その属性情報を用いて利用者に複数のサービス提供を行っている。今回のシステムでは，LDAP proxy を用いて属性加工を行い，Shibboleth attribute filter を用いて属性提供を決定する仕様にしている。これは LDAP の変更には金銭的コストがかかるため変更を加えず，既存のサービスは変更しない仕様にするためである。また今回はサーバの負担を減らすために Shibboleth IdP と LDAP proxy は異なるサーバで運用している。Shibboleth IdP の製作環境を表 1 に LDAP proxy の製作環境を表 2 に表す。

表 1　Shibboleth IdP 製作環境

shibboleth	2.0.0
CentOS	5.3
Apache	2.2.3
tomcat	6.0.18

表 2　LDAP proxy 製作環境

FreeBSD	7.1
OpenLDAP	2.3.43

現在，九州大学の図書館利用状況を管理できるサービスである MyLibrary が Shibboleth SP として提供されている。この SP の認証は上記の環境で構築された Shibboleth IdP を利用している。My Library にアクセスすると図 2 のように Shibboleth 認証画面となり，認証が成功する図 **3** とのようにログイン可能となっている。

図 2 Shibboleth 認証画面

図 3　My Library ログイン画面

Shibboleth 認証可能で，IdP から SP への属性提供もできており期待した動作を確認した。ただ試作システムの製作環境では，認証要求が一人の場合認証処理にかかる時間は 0.7 秒処理であったが一度に 10 人アクセスした場合認証処理にかかる時間は 15 秒となり，処理速度に問題があることがわかった。これはメモリが 256M しかないことに問題があると考えられる。大規模なアクセスが想定されるサービスに対応するためにはより性能の高いサーバで Shibboleth IdP を構築する必要があるだろう。

6. まとめ

本稿では我々が検討している UPKI-Fed における Shibboleth サーバ機構のアーキテクチャについて述べた。今後の計画として九州大学用の Google Apps や SNS サイトを九州大学の Shibboleth IdP 対応とする。また大人数利用者時における処理速度の計測やログの分析を行い，大規模利用者時でも安定的に運用できるように処理の分散化などを行う。研究成果や開発システムの公開と，九州大学 Shibboleth IdP の安定運用を実現していきたい。

7. 参考文献

[1] John Kemp, Security Assertion Markup Language,(SAML, 15 March 2005,

[2] The Shibboleth project (2000),http://shibboleth2.edu/.

[3] 阿部 英司, 伊東 栄典, 笠原 義晃, 中國 真教, “認証つきサービスにおける組織間連携のための PKI と OpenID の融合”, IOT-2-2008, pp.17-22, Jul. 2008.

[4] 金西計英，松浦健二，中川真宏，矢野米雄：大学間 Web サービス連携における間接的な認可の制御について,平成 21 年度情報教育研究集会 D1-5, Nov.14, 2009.