学認との融合化を視野に入れた金沢大学統合認証基盤の構築と運用

Unified Authentication Infrastructure for Seamless Connection among Online Services inside Kanazawa University and Federation “GakuNin”

松平 拓也†，笠原 禎也†，高田 良宏†，東 昭孝†，二木 恵†，
Takuya MATSUHIRA†, Yoshiya KASAHARA†, Yoshihiro TAKATA†,
Akitaka HIGASHI†, Megumi FUTATSUGI†

takusng@kenroku.kanazawa-u.ac.jp, kasahara@is.t.kanazawa-u.ac.jp, yosihiro@kenroku.kanazawa-u.ac.jp,
higashi@staff.kanazawa-u.ac.jp, futamegu@staff.kanazawa-u.ac.jp

†金沢大学総合メディア基盤センター

†Information Media Center, Kanazawa University

概要

我々は，学認環境と金沢大学統合認証環境の間をシームレスに接続できる環境を実現することを目的として，これまでそれぞれ異なる ID を用いて運用していたものを，生涯 ID である金沢大学 ID に統一化した．さらに，平成 23 年度総合情報基盤システム更新に伴い，学認および金沢大学統合認証環境におけるサーバ群を仮想環境に移行し，システムの可用性を高めることを実現した．本稿では，金沢大学 ID へ統一したことによる学認への対応方法について述べ，構築した仮想環境について説明する．さらに，システムの運用実績を示すとともに，今後の課題についても考察する．

キーワード

学認，Shibboleth，シングルサインオン，認証，認可

Abstract

We unified the authentication ID to “Kanazawa University ID”, which is a lifelong ID, both for the online services in our university and those served by “GakuNin (the Academic Access Management Federation in Japan) in order to improve the convenience of the users by realizing the seamless environment between these services. Furthermore, we renewed the comprehensive information system in the university in FY2011 changing over the server group to a virtual environment aiming at the improvement of the availability. In the present paper, we describe the adaptation method of Kanazawa University ID to GakuNin environment. We also introduce the unified authentication service constructed on the virtual environment. Finally we show the operational results of the developed system and discuss our future works.

Keywords

GakuNin, Shibboleth, Single Sign On, Authentication, Authorization

1. はじめに

近年，“学認[1]”の認知度の高まりとともに，学認へ参加する機関が増加している．学認とは，学術認証フェデレーション（以降，GakuNin と記載）の略称で，国立情報学研究所（以降，NII と記載）が中心となって進められている，学術サービスを利用する機関，学術サービスを提供する機関・出版社等から構成された連合体を指す．各機関は GakuNin に参加することで，相互に認証連携を行うことが可能となる．具体的には，各機関で Identity Provider（以降，IdP と記載）と呼ぶ認証サーバを構築し，各機関でそれぞれ認証を行うことで，GakuNin 内でのシングルサインオンを実現する．また，フェデレーション内で提供される情報サービスを Service Provider（以降，SP と記載）と呼ぶ．金沢大学（以降，本学と記載）も，平成 20 年度に NII が中心となって実施された「UPKI 認証連携基盤によるシングルサインオン実証実験[2]」から GakuNin に積極的に関わっており，GakuNin が本格運用に入った平成 21 年からも，運用フェデレーションに参加し，IdP を運用するとともに，3 つの SP も提供している[3][4]．GakuNin では，認証基盤のミドルウェアとして Shibboleth[5]を採用している．Shibboleth は，オープンソースなため，安価にシングルサインオン環境を構築できることから，GakuNin 環境に限らず，学内の統合認証基盤においても採用している機関も増えてきている．本学は，金沢大学統合認証基盤（Kanazawa University Single Sign On（以降，KU-SSO と記載））を，Shibboleth をベースに構築し，平成 22 年 3 月から本格稼働を開始している[6]． KU-SSO への学内情報システムのつなぎこみ（Shibboleth SP 化）は，全学の情報化計画全体を統括する「情報戦略本部」の配下に位置する，「統合認証・ポータル整備 WG」が主導となり進めている．我々は，本 WG のメンバーとして，全学的なイニシアティブを確保したうえで，各部局が保持している情報システムの SP 化を実現してきた．平成 24 年 5 月末現在，本学の各情報システムへの入り口として機能する「アカンサスポータル」をはじめとし，約 30 の情報システムを SP 化し，KU-SSO によるシングルサインオンを実現している．そのため，KU-SSO 環境は，日々重要度が増してきており，365 日，24 時間運用が強く求められてきている．

一方で，本学の GakuNin 環境は，これまで，KU-SSO とは全く別に構築して運用を行ってきていた．同じ Shibboleth をベースとしていながら，同一の環境を利用することは困難であった．その理由として，KU-SSO と GakuNin の対象ユーザの相違が挙げられる．KU-SSO では，既卒者・退職者も含む，金沢大学に関わる全てのユーザを対象としているのに対し，GakuNin では原則として，在籍する教職員・学生を対象としている点である．これまで GakuNin では IdP で認証が成功すると，属性を必要としない SP へアクセスすることができる仕様であった．そのため，KU-SSO で利用する，「金沢大学 ID」と呼称される生涯 ID を利用することができず，GakuNin では，「ネットワーク ID」と呼称される，本学に在籍する教職員・学生のみが利用可能なネットワーク認証用の ID を使用していた．次に，KU-SSO と GakuNin の個人情報の扱いの相違も理由として挙げられる．KU-SSO 環境では，個人情報のやり取りは学内のみで完結される．そのため，学内の SP においては，個人を特定できる属性を IdP から流すことを前提に構築している．一方で，GakuNin 環境では，我々の管理が及ばない学外の SP が主な利用対象となるため，個人情報の送付を無条件に許可できず，個人情報の送付をユーザが承諾していることが原則となる．そのため，学内用に KU-SSO できめ細かく定義した個人情報属性は GakuNin で利用する属性では必ずしも必要ではなく，結果的に両者の属性定義が大きく異なっていた．さらに，KU-SSO と GakuNin の運用ポリシーの違いも大きな要因である．KU-SSO は，ユーザの利便性を第一に考えた結果，SP 間での情報共有やシングルログアウト[6]など，IdP と SP が密に連携した設計・運用が行われている．一方，GakuNin では個々の IdP や SP を運用する主体が一般に異なるため，独立して運用されることが望まれている．それは，ユーザが多種の SP を渡り歩いても，SP 間が結託することでユーザの嗜好性が特定されるべきではないという，ユーザの匿名性を第一に設計されていることが大きい．しかしながら，将来的には，同一の IdP によって，学内の SP はもちろん，GakuNin に参加している SP もシームレスに利用できる環境を構築することが，ユーザの利便性向上とシステム管理負荷低減の双方の観点から望ましい．

我々は，KU-SSO と GakuNin の環境の融合化を進めるべく，両環境で利用する ID 体系を「金沢大学 ID」に統一化を行い，ユーザの利便性を高めるとともに，LDAP サーバにかかる人的・金銭的コストを省力化した．さらに，KU-SSO および GakuNin 環境において，平成 24 年 3 月に総合メディア基盤センターの総合情報基盤システムを更新（以降，System12 と記載）し，その際に，KU-SSO 環境及び GakuNin 環境のサーバ群の仮想化を行い，システムの可用性向上を実現した．

本稿では，これらの課題に対する取り組みについて示し，Shibboleth を学内環境に適用することを検討している，あるいは運用中の他大学の一助となるとともに，GakuNin 普及を促進させることも目的としている．

2. KU-SSO および GakuNin 環境概要

本章では，本研究で我々が構築した，KU-SSO および GakuNin 環境の概要を示す．図 1 に本研究で構築した，KU-SSO および GakuNin 全体図を示す．本環境の説明に先立ち，Shibboleth の概要を以下に示す．

Shibboleth は SAML2.0[7]をベースとした，異なる情報システム間でのシングルサインオンおよび属性共有を実現するオープンソースソフトウェアである．

Shibboleth は 3 つのシステムから構成される．

- IdP
  - ユーザを認証する．
  - ユーザ属性情報を SP に送信する．
- SP
  - ユーザの認証を IdP に要求する．
  - ユーザの属性を IdP から受信し，アプリケーションに渡す．
- Discovery Service (DS)
  - 複数の IdP が存在する場合に，ユーザが適当な IdP を決定するための情報を提供する．

我々は，KU-SSO と GakuNin の環境の融合化に際し，KU-SSO および GakuNin で使用する ID を，金沢大学 ID に統一した．具体的には，GakuNin 用 IdP において，ネットワーク ID を管理する LDAP サーバから，金沢大学 ID を管理する LDAP サーバを参照するように変更した．このことで，本学のユーザは，KU-SSO 環境と GakuNin 環境ともに，金沢大学 ID で利用可能となった．

さらに，重要サービスを担うサーバ群の信頼性向上とともに強く求められていた，365 日，24 時間運用に近づけるべく，GakuNin および KU-SSO 環境の IdP サーバ，金沢大学 ID を管理する LDAP サーバにおいて，System12 の移行に伴い，仮想環境上に配置し，可用性の向上を行った．

3. KU-SSO と GakuNin の融合化

本章では，KU-SSO と GakuNin 環境の融合化を進めるために行った，利用 ID の統一に際し問題となる種々の対策について説明する．特に，GakuNin 環境で金沢大学 ID を利用できるようにするための課題とその解決策について，次節以降詳しく説明する．

3.1. 金沢大学 ID とロール

金沢大学 ID は，全学的な情報システムの一元化および統合認証に用いる生涯 ID として KU-SSO 環境で使用されている．金沢大学 ID は，常勤教職員・非常勤教職員，学生・研究生などを問わず，本学に関わる全構成員に対して 1 人に 1 つずつ付与する ID である．ユーザの重要な個人情報を含む情報サービスの利用に使われることを考慮し，金沢大学 ID の採番方法は，ランダムに与えたアルファベット 3 桁と数字 5 桁の 8 桁とし，他人の ID を容易に推測できないようにしている．また，転学類に伴う学籍番号変更や，卒業後に本学に就職した場合な

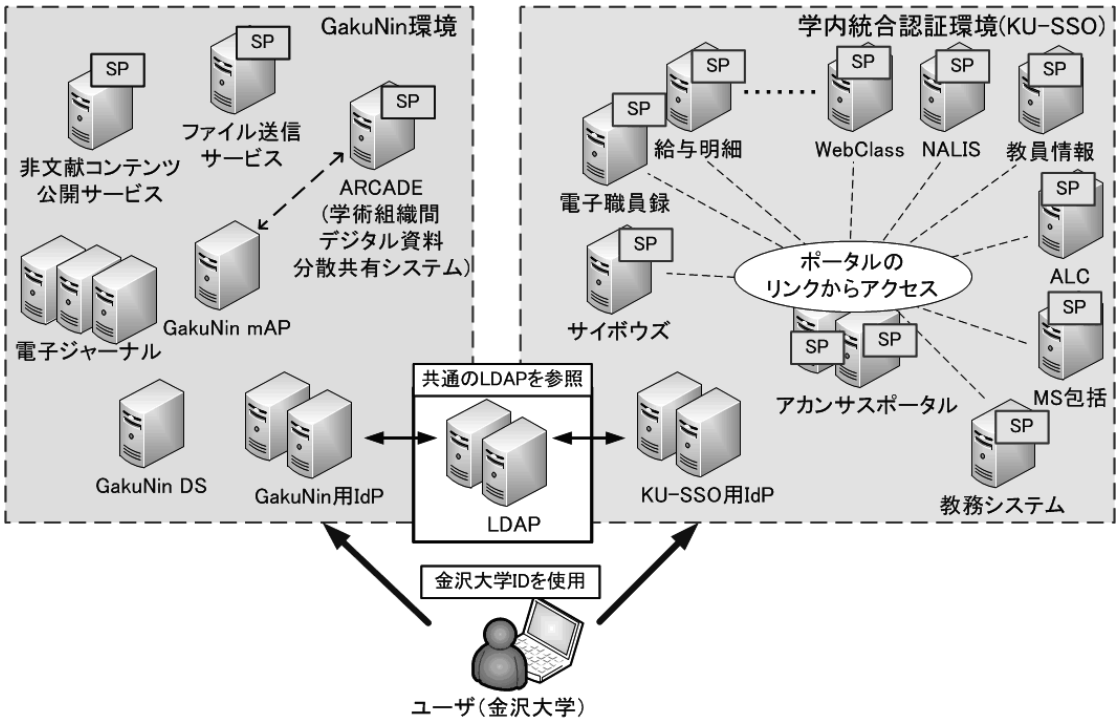

図 1 KU-SSO および GakuNin 全体図

どの場合においても，同一 ID を使用でき，卒業・退職後も ID を抹消されず，“1 ユーザ 1ID” を実現している．

一方，KU-SSO 環境の SP には，学生用の履修登録システムや，教職員用の給与明細システムなど，教育・研究・業務に関する様々な種類があるため，それらを使用できるユーザは，SP の利用目的によってそれぞれ異なる．そのため，各 SP が，ユーザの職分等の属性情報から，当該 SP を利用できる権限があるかどうかを判断できるように，ユーザの属性を“ロール” という名称で定義し，それぞれの SP で必要とされる区分分けを行っている．表 1 にロールの一覧を示す．学生は 4 パターン，教員は 10 パターン，職員は 8 パターン，その他 7 パターンを設定し，全部で 29 パターンに区分した．また，ユーザが複数のロールを持つことも想定している．たとえば，本学の学部を卒業し，博士前期課程を修了した後に本学の職員（学務系以外）になった場合は，学生（既卒），学生（既卒），職員（常勤学務系以外）という 3 つのロールが付与されることになる．なお，学部を卒業した場合と博士前期課程を修了した場合に分類されるロールは「学生（既卒）」で同一であるが，それぞれのロールに付随する学籍番号や所属等の情報が異なるため，SP から得られる情報はそれぞれ異なる．但し，複数ロールを持つユーザであっても，金沢大学 ID は 1 つである．

表 1 ロール一覧表

区分	ロール名	ロール番号
学生	入学前	0
	在学	1
	既卒	2
	退学	3
教員	常勤	4
	教諭	18
	研究員	20
	非常勤講師	38
	名誉教授	37
	医員	5
	TA	32
	RA	33
	退職・転出	6
	その他（学外教員等）	39
職員	常勤学務係	9
	常勤学務係以外	10
	非常勤学務係	11
	非常勤学務係以外	12
	非職員学務係・秘書含	22
	非職員学務係以外・秘書含	13
	教務補佐員	19
	退職	14
その他	管理者	15
	管理者（システム別）	23
	学外学生（公開講座）	16
	ゲスト	17
	学外学生修了（公開講座）	24
	研究協力員（共同研究）	26
	家族等	25

3.2. 金沢大学 ID の学認対応における課題

GakuNin 環境において金沢大学 ID を利用可能にするためにまず留意すべき事項は，前述の特徴を持つ ID による認証でも，「学術認証フェデレーション システム運用基準（以降，運用基準と記載） Ver1.2[8]」に適合する必要がある点である．運用基準の「3.2) 属性情報の信頼性」に「IdP は，自組織に所属する利用者の属性を保証すべきである．また，自組織に所属しない利用者の属性を保証すべきではない．例えば，A 大学の IdP が B 大学の学生の属性を保証すべきではない．ただし，自組織に所属しない利用者を自組織が管理する場合，SP に対する不正なアクセスが発生しないよう特に属性管理に注意することで，そのような利用者の属性を保証してもよい．」と記載されている．「自組織に所属する利用者」を，「在籍する教職員・学生」という解釈は最小構成として揺るぎないもので，そこを確実に保証することは，全ての大学に対して求められている．後半部分において，大学として責任がとれる範囲で、大学の裁量でそれ以外の範囲も含めることができる旨記載されているが，本学では認めていない．

上記を踏まえ，一つ目の課題として，利用者を在籍する教職員および学生のみに制限できるかいうことが挙げられる．前節で説明したロールで言うと，学生（在学），教員（常勤，教諭，研究員，医員），職員（常勤学務係，常勤学務係以外，非常勤学務係，非常勤学務係以外，教務補佐員）が該当する．選定の理由は職員証または学生証が交付されているかどうかである．これらのロールを持つ利用者のみ GakuNin 環境を利用できるように設計を行う必要がある．

二つ目の課題として，LDAP サーバを，金沢大学 ID を管理している LDAP サーバに変更する際に，「属性情報仕様一覧[9]」に記載されている，GakuNin を利用する際に必要となる属性値を賄えるかどうかということが挙げられる．

次節以降，これらの解決方法について説明する．

3.3. GakuNin 利用可能者の制限方法

GakuNin 環境においても金沢大学 ID を利用可能とするためには, GakuNin 用 IdP サーバが金沢大学 ID を管理しているLDAP サーバを参照するように変更する必要がある．但し，変更を行った際，在籍する教職員または学生のみ利用可能という条件を維持するためには，uid の属性値（金沢大学 ID）だけではなく，在籍する教職員または学生のみが持つ属性情報を制限条件に用いる必要がある．本学の場合，ロール属性がそれに該当する．しかし，標準的な IdP の設定では，一旦 uid とパスワードによってユーザの認証が成功すると，IdP 側では属性情報による絞り込みができず，ユーザの属性情報を一切要求しない SP であった場合，たとえ在籍の教職員または学生以外のロールを付与されていても，金沢大学 ID さえ所有していれば, SP を利用できてしまう. 解決策として, IdP から SP に対してロール属性情報を送付し，SP 側で制御を行ってもらう方法がある．しかし，GakuNin 環境では，我々の管理が及ばない SP が大半であるため，現実にはSP 側で利用制限を行うという対応は困難である．そのため，本学の IdP 側で SP の利用制限を行う必要があった.

そこで我々は，SampleFilterPerSP [10]という，IdP のプラグイン機能を利用して本課題の解決を行った．本プラグインを利用することにより，これまで困難であった IdP 側での SP へのアクセス制御を，ユーザの属性情報に基づいて実現することが可能となる. そこで, 本学の IdP での認証後に，SampleFilterPerSP を用いて，3.2 節で述べたロール属性をもつユーザのみ GakuNin 環境の SP へのアクセスを許可するように実装した．ロール属性を用いた SampleFilterPerSP の設定ファイルの記述例を図 2 に示す．この例では，NII が提供する，Fshare[11]という，大容量ファイル転送サービスを利用する場合に SampleFilterPerSP が動作する.「roleNumber」は金沢大学 ID を管理している LDAP サーバに持つロール属性で, 表 1 のロール番号が該当する．図 2 のように記載することで，学生（在学），教員（常勤，教諭，研究員，医員），職員（常勤学務係，常勤学務係以外，非常勤学務係，非常勤学務係以外，教務補佐員）のいずれかのロール属性を持つユーザだけが Fshare を利用することができる．そして，それ以外のロールのユーザについては，図 3 に示す画面を表示し, このSP を利用できないようにできる．このように，設定ファイルに記述されている SP 毎に，特定の属性値を持つユーザについてのみ SP の利用を許可することを IdP 側で制限でき, 金沢大学 ID を利用した場合でもGakuNin のポリシーに準拠することを実現した.

```
<EntityDescriptor entityID="https://fshare.sinet.ad.jp/shibboleth">
  <Attribute attributeID="roleNumber">1</Attribute>
  <Attribute attributeID="roleNumber">4</Attribute>
  <Attribute attributeID="roleNumber">5</Attribute>
  <Attribute attributeID="roleNumber">9</Attribute>
  <Attribute attributeID="roleNumber">10</Attribute>
  <Attribute attributeID="roleNumber">11</Attribute>
  <Attribute attributeID="roleNumber">12</Attribute>
  <Attribute attributeID="roleNumber">18</Attribute>
  <Attribute attributeID="roleNumber">19</Attribute>
  <Attribute attributeID="roleNumber">20</Attribute>
</EntityDescriptor>
```

図 2 SampleFilterPerSP 記述例

図 3 アクセス拒否画面

3.4. GakuNin 利用属性の扱い

二つ目の課題として，LDAP サーバを，金沢大学 ID 管理用に変更しても，GakuNin で必要な属性を利用できるようにする必要がある．LDAP を変更することで対応が必要であった GakuNin で必要な属性である，eduPersonPrincipalName（以降，ePPN と記載），eduPersonAffiliation（以降，ePA と記載），eduPersonScopedAffiliation（以降，ePSA と記載）についてそれぞれ説明する.

- ePPN

ePPN は，GakuNin 内で一意な，かつ，永続的な利用者識別子であり，GakuNin 内での一意性を保証する必要がある．金沢大学 ID は 1 ユーザ 1ID で，永続的に一意性が保証されているため，この条件についてはクリアしている．但し，金沢大学 ID は生涯 ID であり，変更できないため，そのままの値で送出すると，万一漏えいがあった場合に非常に問題となる．そこで，IdP の設定ファイルの一つである「attribute-resolver.xml」内で金沢大学 ID に変換処理をかけて，ePPN から金沢大学 ID を割り出せない形で送信するように設計した．attribute-resolver.xml の記述を図 4 に示す．このように，attribute-resolver.xml 内ではScript タグ内においてJavaScript が利用可能である．セキュリティ上，一部の変換部分は記載していないが，まず，uid 属性である金沢大学 ID を取得し，金沢大学 ID に対して特定の変換処理をかけた後に，ePPN として各 SP に対して送出している．例として，

123abc456def789hig@kanazawa-u.ac.jp

というような，金沢大学 ID を推測できない形で，かつ GakuNin 内で一意性を保証した上で送信を行っている.

```
<!-- Attribute Definition for eduPersonPrincipalName -->
  <resolver:AttributeDefinition id="principalName" xsi:type="Script"
xmlns="urn:mace:shibboleth:2.0:resolver:ad"

            sourceAttributeID="uid">
    <resolver:Dependency ref="myLDAP" />

    <resolver:AttributeEncoder xsi:type="SAML1String"
xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
        name="urn:mace:dir:attribute-def:eduPersonPrincipalName" />

    <resolver:AttributeEncoder xsi:type="SAML2String"
xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
        name="urn:oid:1.3.6.1.4.1.5923.1.1.1.6"
friendlyName="eduPersonPrincipalName" />
    <Script>
      <![CDATA[

importPackage(Packages.edu.internet2.middleware.shibboleth.common.attribute.
provider);
          importPackage(Packages.org.apache.commons.codec.digest);
          uniqueValue = uid.getValues().get(0);


          principalName.getValues().add(localpart + "@kanazawa-u.ac.jp");
      ]]>
    </Script>
  </resolver:AttributeDefinition>
```

図 4 ePPN 記述方法

なお，不正アクセスなどにより，SP 側からユーザの特定を要求された場合は，本学側で，発行済みの金沢大学 ID に対して，ePPN を生成している変換処理を順次かけていき，マッチングを行うことで，当該 ePPN を利用している金沢大学 ID を割り出し，ユーザの特定を行う．

- ePA および ePSA

ePA は，GakuNin 内における利用者の職種等を表す．GakuNin では，ePA には「faculty」，「staff」，「student」，「member」，「無し（空白）」の値が利用可能である．ePSA に設定する属性値は ePA と同値であるが，利用者が所属する機関を示すため，@以下にスコープを付加したものとなる（例：staff@kanazawa-u.ac.jp）． 我々は，現段階で ePA および ePSA を要求している SP の状況を鑑み，教職員を「staff」，学生を「student」としている．ただし，ここで考慮すべきこととして，一つの金沢大学 ID に二つ以上のロールを持つユーザの対応がある．例として，本学教員でありながら，大学院に社会人入学している場合は，教員かつ学生のロールを持つため，staff と student 両方の属性を SP に送出する必要がある．staff しか利用できない SP も存在するため，student しか送出されないことによる不利益が生じないように，両方の属性を送出する必要がある．

そこで，我々は，attribute-resolver.xml に図 5 のように記述することで対応した．まず，ユーザが持つロール数を取得し，「size」に格納する．そして，ロール数分 for 文を通す． ロール番号が 1 であれば「student」，4,5,9,10,12,18,19,20 のいずれかであれば，「staff」属性が ePA に付加される．それ以外のロール番号であれば何も属性が付加されない．先の例のように，教員かつ学生の場合は staff と student の両方が SP に送出されることになる．

```
<!-- Attribute Definition for eduPersonAffiliation -->
  <resolver:AttributeDefinition id="eduPersonAffiliation" xsi:type="Script"
xmlns="urn:mace:shibboleth:2.0:resolver:ad" sourceAttributeID="roleNumber">
    <resolver:Dependency ref="myLDAP" />
    <resolver:AttributeEncoder xsi:type="SAML1String"
xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
        name="urn:mace:dir:attribute-def:eduPersonAffiliation" />
    <resolver:AttributeEncoder xsi:type="SAML2String"
xmlns="urn:mace:shibboleth:2.0:attribute:encoder"
        name="urn:oid:1.3.6.1.4.1.5923.1.1.1.1"
friendlyName="eduPersonAffiliation" />
    <Script>
      <![CDATA[

importPackage(Packages.edu.internet2.middleware.shibboleth.common.attribute.
provider);

    eduPersonAffiliation = new BasicAttribute("eduPersonAffiliation");

    size = roleNumber.getValues().size();

    for (i=0;i<size;i++){
      if (roleNumber.getValues().get(i) == "1" ) {
        eduPersonAffiliation.getValues().add("student");
      }
      if (roleNumber.getValues().get(i) == "4" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "5" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "9" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "10" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "11" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "12" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "18" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "19" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
      if (roleNumber.getValues().get(i) == "20" ) {
        eduPersonAffiliation.getValues().add("staff");
      }
    }
      ]]>
    </Script>
  </resolver:AttributeDefinition>
```

図 5 ePA 記述方法

図 6 EV-SSL 証明書

このように，GakuNin で必要とされる属性で，LDAP サーバに持たない属性の取り扱いにおいてもクリアすることができた．

3.5. EV-SSL 証明書

このように, GakuNin 環境においても金沢大学 ID を利用することにより，ユーザは，一つの ID のみを管理すればよく，利便性が向上する．しかし，金沢大学 ID を対学外のサービスにも適用することで，ユーザは，認証画面が表示されると，金沢大学 ID を条件的に入力してしまい，フィッシング詐欺の被害にあう危険性が想定される．KU-SSO と GakuNin で IdP を分けて構築しているのは，そのあたりをユーザに理解してもらうという側面がある．そこで，我々は，KU-SSO 用 IdP，GakuNin 用 IdP および多くの重要な情報を扱う，アカンサスポータルに対して，EV-SSL 証明書[12]の導入を行った．EV-SSL は Extended Validation-Secure Socket Layer の略で，証明書に記載される組織が法的かつ物理的に実在し，その組織が証明書に記載されるドメインの所有者であることを認証する．ドメインの所有者の確認は，WHOIS データベースに照会し，組織の法的実在性の確認は，公的書類や組織の実運用性を第三者データベース（帝国データバンク・DUNS・職員録）へ照会をかける．なお，公的書類には印鑑証明書だけでなく，登記事項証明書等も含まれる．さらに，EV-SSL 証明書は個人で取得することはできず，法人のみ取得可能となっている．世界標準の認証ガイドラインで，サーバ証明書の中で最も厳格な審査が行われている. 図 6 に EV-SSL 証明書を導入した KU-SSO 用 IdP の画面を示す．このように，EV-SSL 証明書を導入しているサイトにアクセスすると，アドレスバーが緑色に変化する．さらに，アドレスバーの右側に，サイトを運営している大学名が表示され, そこをクリックすると, 認証局の情報やサイト運営組織の所在地も確認できる．このように，EV-SSL 証明書を導入することで，ユーザに対して，アドレスバーの色とサイトの運営組織を確認したうえで，金沢大学 ID とパスワードを入力するように指導するようにした．

なお，今回，我々は EV-SSL 証明書を GMO グローバルサイン株式会社[13]（以降，グローバルサインと記載）から購入した. グローバルサインを選択した理由として, ライセンス体系が挙げられる．我々は，IdP サーバを冗長化して運用しており，他の認証局ではその場合 2 ライセンス必要であった．一方でグローバルサインは 1 コモンネームあたりの金額で，複製してインストール可能であり，冗長化したサーバにインストールするためのライセンスのコストを抑えることができた．

4. System12 への移行

4.1. システム構成

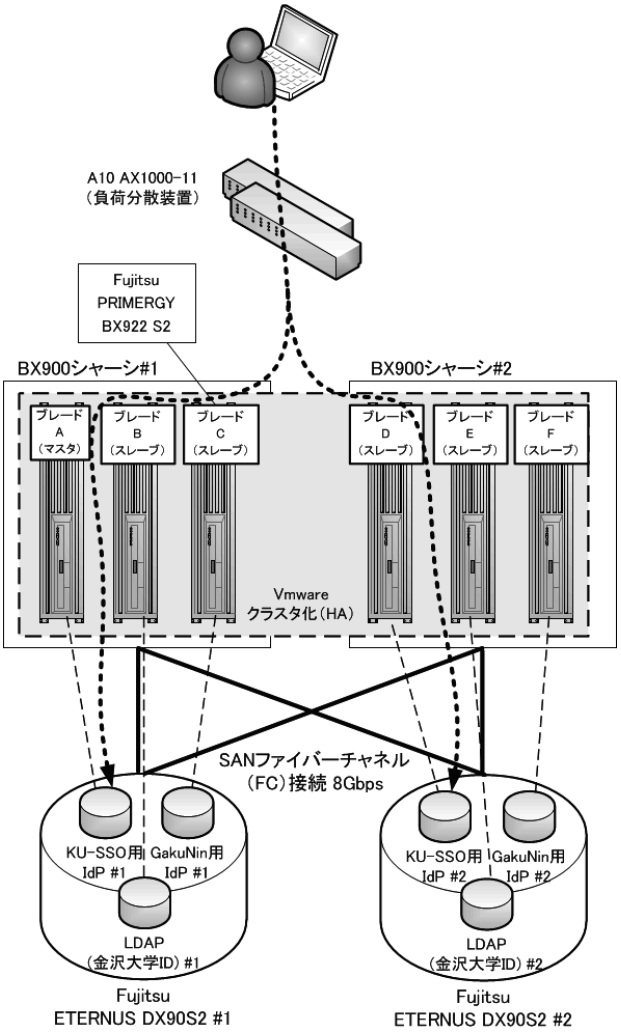

図 7 System12 システム構成図

我々は，System12 への移行に伴い，KU-SSO および GakuNin の IdP サーバ，金沢大学 ID を管理する LDAP サーバを仮想環境に移行して可用性の向上を目指した．System12 への移行前は，これらのサーバは，占有ハードウェアによる構成であった．そのため，ハードウェアに障害が発生した場合は，障害部分を修理するまで復旧は困難であった．なお，KU-SSO および GakuNin 両環境で金沢大学 ID を利用できるようにしたことで，今後はネットワーク ID を管理していた LDAP サーバ 4 台について意識する必要がなくなるとともに各種メンテナンスも不要となり, 金銭的･人的コストの削減に寄与している．System12 におけるシステム構成の概念図を図 7 に示す．System12 における仮想環境には，VMware vSphere 5 Standard[14]を採用している．なお，統合認証関係には，全部で 6 台のブレードサーバが割り当てられている．ブレードサーバのハードウェア諸元は以下のとおりである．

- Fujitsu PRIMERGY BX922 S2[15]
  - CPU：Xeon X5690 (3.46GHz/12MB/6 コア)
  - メモリ：48GB

ブレードサーバ自体はディスクを持たず，Fujitsu ETERNUS DX90S2[16]（以降，DX90S2 と記載）と SAN ファイバーチャネル（8Gbps）で接続されている．各ブレードサーバにおいて仮想サーバを立ち上げるためのハイパーバイザである ESXi ホストは, DX90S2 上にインストールする．なお，ブレード A～C の ESXi ホストは DX90S2 #1, ブレード D～F の ESXi ホストは DX90S2 #2 上で稼働する．そして，ブレード A～C の ESXi ホスト上で動作する仮想サーバは DX90S2 #1 に，ブレード D～F の ESXi ホスト上で動作する仮想サーバは DX90S2 #2 上にそれぞれインストールを行う．また，負荷分散装置として，A10 ネットワークス株式会社の AX1000-11 [17] を用いて，SSL アクセラレータによるクライアント単位での負荷分散を行う設計とする．SSL アクセラレータを用いて負荷分散を行うことで，プライベート配下のクライアントからアクセスが来た場合でも，均等に負荷分散を行うことが可能となる．なお，AX1000-11 自身もクラスタ化を行い，冗長性を高めるようにする．さらに，VMware の機能の一つである, vSphere High Availability（以降，HA と記載）を使用する．HA を構成すると，1 つの ESXi ホストがマスタとなり，その他の ESXi ホストがスレーブに設定される．マスタホストは，スレーブホストの状態を監視し，スレーブホストに障害が発生した場合に，当該ホスト上で稼働している仮想サーバをフェイルオーバーするホストを決定し，仮想マシンをすべて，ほかのホストでただちに再起動する．

4.2. IdP サーバの動作

本節では，KU-SSO および GakuNin 環境における IdP サーバの構成について説明する．それぞれの環境における IdP サーバの構成を以下に示す．

- ハードウェア割り当て（1 ゲスト OS あたり）
  - CPU：2vCPU（仮想ソケット数 1，コア数 2）
  - メモリ：8GB
  - HDD：120GB（シンプロビジョニング）
- OS
  - Red Hat Enterprise Linux 6.2 (x64)
- ミドルウェア
  - Shibboleth IdP 2.3.6
  - Apache 2.2.15
  - Jdk 1.6.0update31
  - Tomcat 6.0.35
  - Ant 1.8.3
  - Terracotta 3.6.1

図 7 に基づき，KU-SSO 用 IdP サーバの可用性について説明する．KU-SSO 用 IdP は，ブレード A の ESXi ホスト上に一台（IdP #1），ブレード D の ESXi ホスト上にもう一台（IdP #2）構築する．AX1000-11 を介して負荷分散を行う設計とするが，IdP はセッション情報を持つため，単純に負荷分散することはできない．そこで，Terracotta[18]を用いて IdP の冗長化を行うようにする．Terracotta は複数の JavaVM 上で同じ Java オブジェクトを共用できるオープンソースのミドルウェアである．Terracotta を用いることで，IdP 間でセッションを共有できるようになる．どちらかの DX90S2 に障害が発生した場合，AX1000-11 により，障害が発生している IdP には振り分けられなくなるようにし，IdP の動作を継続させるようにする. また, ブレードDに障害が発生した場合, HA 機能により，ブレード D 以外の ESXi ホスト上で，自動で IdP #2 が立ち上がるようにし，IdP #1 が一台で動作する時間を数分程度で済む設計とする．ただ，この間も，AX1000-11 で振り分けを行い，IdP の動作が継続するようにする．

このように，様々な部分で障害が発生した場合でも，極力管理者の手を介することなくサービスを継続できる設計とする．

5. 運用実績

本章では，これまで述べてきたシステムの運用実績を示す．System12 による運用は，平成 24 年 3 月 19 日から開始している（AX1000-11 のみ平成 24 年 7 月 8 日運用開始予定で，それまでは DNS ラウンドロビンで対応）．GakuNin と KU-SSO の IdP サーバは，ハードウェアおよびソフトウェアの構成が同じで，GakuNin よりも KU-SSO の認証処理が圧倒的に多いことから，KU-SSO の IdP で評価を行った．平成 24 年 3 月 19 日から平成 24 年 6 月 24 日までの IdP での認証数の推移を図 8 に示す．4 月は全入学生対象の情報処理基礎の講義や，学生の履修登録, さらにはアカンサスポータルを経由しての LMS の利用があるため，一年で一番 IdP での認証が行われる月である．多い時で，一日あたり 2 万件を超える IdP での認証処理が行われており, 平成 24 年 4 月 19 日に 21,477 回の認証処理が行われていることがわかる．そこで，平成 24 年 4 月 19 日の IdP #1 における IdP の認証数とそれに伴う CPU 負荷率およびメモリ使用率を図 9 に示す. 認証数が増加した場合でも，CPU 負荷率およびメモリ使用率に変化がほとんど見られず，どちらも十分に余力があることがわかり，IdP ではリソースに負荷を与えることなく運用できていることを確認することができた．

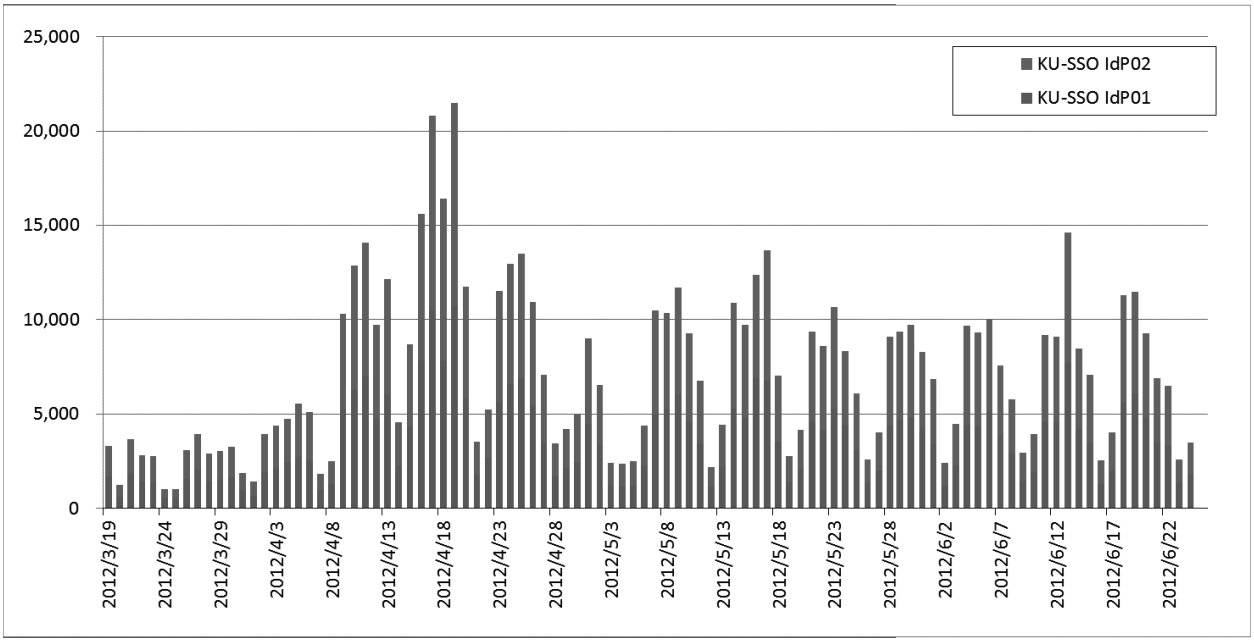

図 8 KU-SSO 用 IdP における認証数の推移（2012/3/19～2012/6/24）

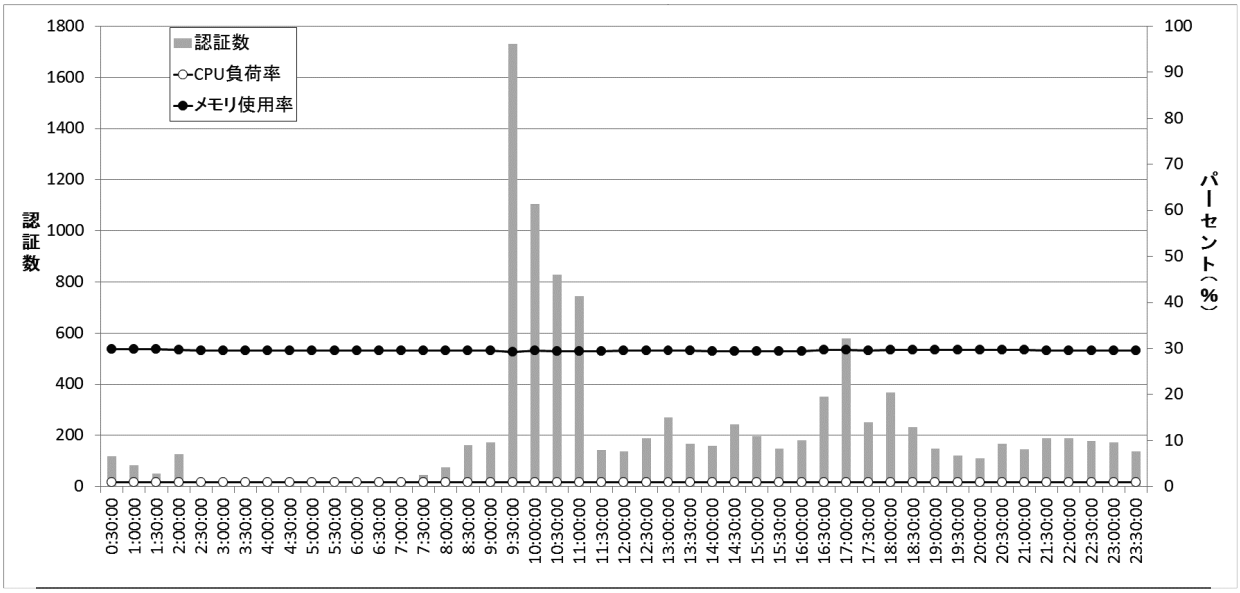

図 9 KU-SSO の IdP サーバにおける認証数と CPU 負荷率及びメモリ使用率（2012/4/19）

6. まとめ

今回，KU-SSO と GakuNin で使用する ID を金沢大学 ID に統一したことにより，ユーザは金沢大学 ID で GakuNin と KU-SSO のどちらもアクセスできるようになり，GakuNin と KU-SSO 環境の融合化を進めることができた．ID 統一化の際に発生した，GakuNin と KU-SSO の利用者範囲の相違の問題においては，SampleFilterPerSP を用いることで，本学 IdP 側で，特定のロールをもつユーザだけが個々の SP の利用を制御できるようにすることで解決した．また，GakuNin で参照する LDAP サーバを変更することにより生じた，GakuNin で必要ないくつかの属性の摺合せにおいては，LDAP 内でもつ属性を attribute-filter.xml 内で加工することで対応した．結果として，金沢大学 ID を管理する LDAP サーバのみを意識するだけでよくなり，人的負荷や運用コストの削減を実現できた．さらに，金沢大学 ID を対学外用サービスで用いることによるフィッシング対策として，EV-SSL 証明書を導入したことで，ユーザに対して視覚的に注意を促すことができた．また，本学において重要度が高い KU-SSO 環境および GakuNin 環境において，System12 への移行に伴い，KU-SSO 環境と GakuNin 環境で使用しているサーバ群を VMware で仮想化したことにより，システムの可用性を高めることができた．さらに，構築したシステムの評価を行ったことで，実運用においても十分耐えうるシステムであることを実証できた．

今後の課題として，KU-SSO および GakuNin の IdP サーバの完全統合による両サービスのシングルサインオン化が挙げられる．現在のところ，それぞれの環境をまたぐ際には，認証画面を表示することで，ユーザに注意喚起を促しているが，EV-SSL によるユーザリテラシーが向上したのち，進めていきたいと考えている．ただし，現在は KU-SSO 環境においてはシングルログアウトを実現しているが，GakuNin 環境では実現できていない．KU-SSO と GakuNin 環境をシームレスに利用できる環境を目指すために，両環境におけるシングルログアウトを実現できる機構を考案する必要があると考えている．海外では，各大学に統合認証システムを持っており，認証を行うと学内の情報サービスはもちろんのこと，国のフェデレーションにあるサービスもシームレスに利用可能なところも多い．我々も，将来的には KU-SSO に一本化し，KU-SSO と GakuNin でそれぞれ提供している SP の一元化及びシームレスに利用できる環境の構築に努めていきたい．

謝辞

本研究は科研費 若手研究 B （22700809）の助成を受けたものである．

参考文献

[1] GakuNin，http://www.gakunin.jp/docs/fed (accessed 2012.6)
[2] 平成 20 年シングルサインオン実証実験報告書，https://www.gakunin.jp/docs/open/fed/6 (accessed 2012.6)
[3] 松平 拓也, 笠原 禎也, 高田 良宏, 井町 智彦, “UPKI 認証連携基盤に基づく安全なデータ共有システム構築の試み”，学術情報処理研究，No13，pp.84-90，(2009)
[4] IdP, SP 一覧，https://www.gakunin.jp/docs/fed/participants (accessed 2012.6)
[5] Shibboleth，http://shibboleth.net/ (accessed 2012.6)
[6] 松平 拓也, 笠原 禎也, 高田 良宏, 東 昭孝, 二木 恵, 森 祥寛，“大学における Shibboleth を利用した統合認証基盤の構築”，情報処理学会論文誌，Vol.52 No.2，pp.703-713，(2011)
[7] SAML2.0, http://www.oasis-open.org/specs/index.php (accessed 2012.6)
[8] 学術認証フェデレーション システム運用基準 (Ver. 1.2)，https://www.gakunin.jp/docs/files/GakuNin_System_SpecV1.2.pdf (accessed 2012.6)
[9] GakuNin 属性リスト，https://www.gakunin.jp/docs/fed/technical/attribute (accessed 2012.6)
[10] SampleFilterPerSP，https://www.gakunin.jp/docs/fed/technical/idp/customize/knowhow/authorization (accessed 2012.6)
[11] Fshare，https://fshare.sinet.ad.jp/ (accessed 2012.6)
[12] EV-SSL 証明書，http://jp.globalsign.com/service/ssl/ev.html (accessed 2012.6)
[13] GMO グローバルサイン株式会社，http://jp.globalsign.com/ (accessed 2012.6)
[14] VMware vSphere 5 Standard，http://www.vmware.com/jp/products/datacenter-virtualization/vsphere/mid-size-and-enterprise-business/overview.html (accessed 2012.6)
[15] PRIMERGY BX922 S2，http://jp.fujitsu.com/platform/server/primergy/product-navi/html/bx922s2-201011-pgx9s24ba.html (accessed 2012.6)
[16] ETERNUS DX90S2，http://storage-system.fujitsu.com/jp/products/diskarray/dx-entry/ (accessed 2012.6)
[17] A10 AX1000-11，http://www.a10networks.co.jp/press/pressrelease/12/2/2011.html (accessed 2012.6)
[18] Terracotta，http://terracotta.org/ (accessed 2012.6)