Vol.2013-CSEC-61 No.28
Vol.2013-IOT-21 No.28
2013/5/10

学割サービス実現のための SAML-OpenID ゲートウェイの試作

中村素典[†1]　西村健[†1]　山地一禎[†1]　佐藤周行[†2]　岡部寿男[†3]

近年，シングルサインオンの技術を応用し，組織をまたがった認証連携を行う取り組みが広がりを見せている．民間では OpenID の仕組みを活用した認証連携が進む一方，学術では SAML (Security Assertion Markup Language)を用いた認証連携の枠組みが，国を単位として世界的に立ち上がってきている．OpenID や SAML では，認証結果とともに，認証された利用者に関する属性情報をサービス提供側に受け渡す仕組みが用意されているため，単なる利用者の本人確認にとどまらない，高度な情報連携の可能性を秘めている．本報告では，大学等が運用管理する認証サーバと，民間のサービス提供サーバとを連携させ，大学が保持する職種（学生，職員，教員など）に関する属性情報を提供することにより，学割サービスを実現するためのプロトコルゲートウェイの設計および試作について述べる．

Trial Implementation of SAML-OpenID Gateway for realizing Student Discount Services

Motonori NAKAMURA[†1]　Takeshi NISHIMURA[†1]　Kazutsuna YAMAJI[†1]
Hiroyuki SATO[†2]　Yasuo OKABE[†3]

A framework of cooperation on user authentication among organizations has begun to spread widely by utilizing Single-Sign-On mechanisms. OpenID is mainly used for commercial field, and SAML (Security Assertion Markup Language) is mainly used for academic field. Major countries are constructing an academic federation each. These mechanisms have possibilities for advanced cooperation since both have a mechanism to provide attribute information on authenticated user for service sites. This manuscript reports on a trial to implement a gateway between SAML and OpenID to support academic/student discount service by providing affiliation information of users, such as student, faculty, staff, etc., which is maintained by universities, by cooperating authentication server operated by universities and service sites operated by commercial providers.

1. はじめに

最近のクラウドサービスの充実により，自組織の外で提供されるサービスの認証に，自組織の認証機構を利用する認証フェデレーションの利用が急速に広まっている．このベースにあるものは，いわゆる認証・認可の分離に基づいたシングルサインオン技術の活用であり，国際標準である SAML (Security Assertion Markup Language) [1]や OpenID [2]に基づいた機構が主として利用されている．学術分野においては，SAML に基づいた認証フェデレーションの構築が主流となっており，国を単位として欧米を中心に構築が進んでいる[3]．日本においても，国立情報学研究所を中心に 2009 年より構築を開始し, 2010 年より学術認証フェデレーション「学認」として実運用を行っている[4][5]．一方，商用サービス提供授業者においては, OpenID の活用が進んでおり，Aol，Facebook，Google，mixi，Yahoo!などに登録されたアカウントを利用して，他のサービスにログインできるような環境の展開が進んでいる[6].

この 2 つの認証フェデレーションは，上述のようにそれぞれ異なる機構を利用しているが，民間で提供される商用サービスの充実と高度化により，学術分野においても，これらを研究教育に採用する例が増加してきている[7][8][9]. 他方，民間サービスにおいては，福利厚生的利用を含め，学術関係者の利用に対して割引を提供する慣例が古くからあり，割引をオンラインサービスにおいても同様に実現しユーザを確保したいという要望がある．基本機構の異なる２つのフェデレーションを技術的に接続し，学術機関側から認証とユーザの身分を示す情報を提供することにより，割引を考慮したサービスの提供を実現することが可能となる．

そこで，このような仕組みを実現するために必要となる機能について検討し試作を行ったので報告する．

2. 認証フェデレーション

2.1 シングルサインオン

サービス毎に異なる ID およびパスワードを用いて認証を行うことは，システム管理者とユーザの双方にとって煩雑である．複数のサービスで共通の ID とパスワードを利用するためには，まず LDAP[10]等を用いて ID およびパスワードを保持する認証データベースを統合し，各サービスから同一の認証データベースを参照する方法が考えられる. さらにそこから，各サービスから認証機能を分離し，サービス共通の認証サーバを作り，ユーザは ID とパスワードをそのサーバに入力して認証結果を各サービスに伝える，という形態に発展させることにより，シングルサインオン

†1 国立情報学研究所
National Institute of Informatics
†2 東京大学
The University of Tokyo
†3 京都大学
Kyoto University

が実現される．認証サーバが認証状態をしばらくの間保持し，その間に他のサービスからの認証要求が来た際に，ユーザに対する再認証（ID とパスワードの再入力）を求めない仕組みを持たせることにより，いわゆるシングルサインオンとしての機能が実現される．このシングルサインオンの機能を，一つの組織内で閉じて利用するだけでなく，他の組織と連携させる形で活用する形態は「認証フェデレーション」あるいは単に「フェデレーション」と呼ばれる．

2.2 基本アーキテクチャ

SAML や OpenID は，認証フェデレーションを実現するための枠組みを提供するものである．サービスで共通の認証サーバのことを, SAML では IdP (Identity Provider)と呼び, OpenID では OP (OpenID Provider)と呼ぶ．一方，この認証サーバにおける認証処理の結果に基づいて実際にサービスを提供するサーバのことを，SAML では SP (Service Provider)と呼び，OpenID では RP (Relying Party)と呼ぶ．名称は異なるが，基本的には同様の機能を提供するものである．一組織の中で閉じたシングルサインオンでは，IdP/OP は一つだけ存在するのが一般的であるが，認証フェデレーションの場合は，フェデレーションに参加しサービスを利用する組織毎に IdP/OP を構築する分散型の構造をとることになる．

学術分野において広く用いられている SAML をサポートしたプラットフォームとしては，Shibboleth [11]や SimpleSAMLphp [12]などがある. 一方, OpenID についても，いくつかのライブラリが提供されている．

2.3 属性送信と送信同意

SAML や OpenID によって提供されるシングルサインオン機構では，IdP/OP は認証結果として認証の可否を SP/RP に伝えるだけでなく，併せて認証されたユーザに関する情報を伝達する機能を持つ．このような情報は属性情報と呼ばれる．例えば，学認では，システム運用基準[13]において 15 種類の属性情報を定義している．これらの多くは Internet2 で定義されている eduPerson オブジェクトクラス[14]のものをベースとしているが，例えば，その中の eduPersonAffiliation はユーザの身分を示す属性情報であり, student (学生)，faculty (教員)，staff (職員)などの値を持つ．OpenID では，OpenID 2.0[15]の拡張仕様である Attribute Exchange[16]において属性情報の交換が規定されているが, 最近では OAuth 2.0[17]をベースとして設計された OpenID Connect[18]が策定され，そちらへの対応が始まっている[19].

認証フェデレーションでは，他の組織に対して属性情報を送信する場合が生じるが，このような情報の送信は業務委託によるサービス利用でない限り第三者提供にあたる．属性情報のうちの一部は個人情報に相当するものであり，プライバシー保護のための考慮が求められる．日本においては, プライバシー保護に関する法律[20][21]が定められており, 一般には事前の同意 (オプトイン; Opt-In)あるいは事後の求めによる提供停止 (オプトアウト; Opt-Out)に対応することが求められる．特に国立大学については，独立行政法人に準じる扱いが求められるため，オプトインをサポートする必要がある．公立大学については地方公共団体が定める条例が定めているが，大半は独立行政法人に準じた扱いとなっている．オプトインを実現するには，ユーザから情報提供を受ける際に事前の同意を得ておく方法もあるが，情報の提供先が頻繁に追加されることを考えると，IdP において，情報を送信する際に同意を得る仕組みを提供することが望ましいと考えられる．個人情報の送信同意のためには，スイスのフェデレーション SWITCHaai [22] を提供する SWITCH が開発した，Shibboleth IdP において利用可能なプラグインである uApprove[23]を利用することができる．学認では，日本語に対応するとともに，よりきめ細かな制御ができるようにするために改良した uApprove.jp を提供している[24].

3. フェデレーション連携によるサービス提供

3.1 Student Identity Trust Framework

学術分野における認証フェデレーションと，民間商用サービスにおける認証フェデレーションは，それぞれ異なる機構をしながら独立に発展してきたものである．しかしながら，民間で提供される商用サービスの充実と高度化はめざましく，また，クラウド環境の普及も相まって，学術分野においても，各種サービス向けのシステムを自力で構築せずに，アウトソーシングした方が安価かつよりよいサービスが実現できるようになってきたことから，民間商用サービスを研究教育に採用する例が増加してきている. 他方，民間サービスにおいては，福利厚生的利用を含め，学術関係者の利用に対して割引を提供する慣例が古くからあり，割引をオンラインサービスにおいても同様に実現しユーザを確保したいという要望がある．このような背景から，基本機構の異なる２つのフェデレーションを技術的に接続し，学術機関側から認証とユーザの身分を示す情報を提供することにより，割引を考慮したサービスの提供を実現することについての検討を開始した[25]．認証フェデレーションは異なる組織が提供する IdP と SP とが連携して実現されるものであり，相互の信頼関係が重要であることから，トラストフレームワーク(Trust Framework)とも呼ばれる．そこで, 学生の身分であることを大学が責任をもって証明し，その情報に基づいて学割サービスを提供する試みを，Student Identity Trust Framework (SITF)と呼んでいる．

3.2 SITF における認証連携アーキテクチャ

SITF における認証連携アーキテクチャにおいては, 大学が IdP を提供し，民間サービス側が RP を提供するモデルについて考える．大学側は SAML ベースの認証連携に基づく IdP を提供し，民間サービス側が OpenID Connect ベース

の認証連携に基づく RP を提供することになるため，その間を橋渡しするためのプロトコルゲートウェイが必要となる（図 1）．このプロトコルゲートウェイは，SAML 側においては SP としての役割を果たし，OpenID 側においては OP としての役割を果たす．また，ユーザに関する属性情報の送信について，OpenID 側の RP ごとに制御できるべきという観点から，このプロトコルゲートウェイは，サービス（すなわち RP）ごとに用意することを想定する．

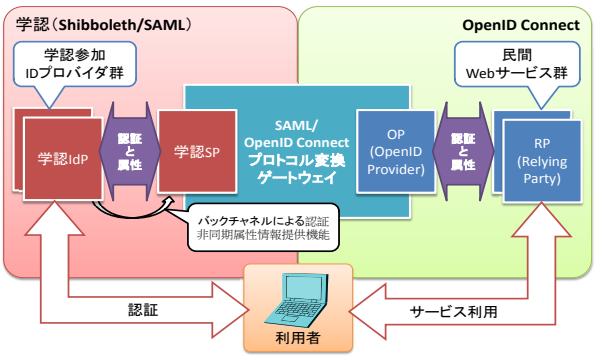

図 1　SAML/OpenID プロトコルゲートウェイのデザイン
Figure 1　A design of SAML/OpenID Protocol Gateway

学割が適用されるサービスにおける課金方法としては，次の 2 通りの形態が考えられる．

1. 一回のサービス利用ごとに対価を支払うもの（チケット等の購入など）
2. 継続的なサービスの利用権を取得し，定期的に対価を支払うもの（月額料金が設定されたアクセスサービスなど）

後者としては，例えば UQ コミュニケーションズが提供する「モバイル WiMAX キャンパスネットワーク接続サービス」[26]のようなものが該当する（実際には，このサービスは直接 SAML に対応する形で，割引に対応している）．

このような形態において，継続的な割引を適用するためには，ユーザの在籍確認を定期的に行う必要がある．毎回の在籍確認のタイミングで，ユーザに認証を求めることは煩雑であることから，ユーザへの対応を求めることなく，IdP に対して在籍確認が可能となる仕組みが望まれる．SAML では，バックチャネルと呼ばれる，ユーザのブラウザを介さず，SP が IdP から直接属性情報を取得するための機能が備わっているため，これを活用する方法が考えられる．しかし，この機能はユーザの認証を受けて属性情報を取得する形態を想定しており，有効期間が非常に短い（1 時間以下）セッション識別子を用いる方法を採っている（以下，この形態を同期型とする）．定期的な課金での在学確認のためには，有効期間の長い識別子を用いた属性情報の取得に対応する必要がある．そこで，SP ごとに生成される永続的なユーザ識別子である eduPersonTargetedID (ePTID)を利用した属性情報の取得に対応することとした．

4. プロトコルゲートウェイの設計と試作

4.1 概要

3 章で述べた，SAML の IdP と OpenID Connect の RP の連携を実現するために必要となるプロトコルゲートウェイを含めた全体の処理フローを図 2 に示す．SAML をベースとする学術の認証フェデレーションである学認では，Shibboleth が広く用いられていることから，IdP は Shibboleth によるものを前提とする．しかし，オリジナルの Shibboleth は，バックチャネルを用いた非同期の属性情報送信に対応していないため，プロトコルゲートウェイの構築とは別に，SP に対する拡張も行う．また，非同期の属性情報送信は，ユーザの関知しないところで行われるため，ユーザが随時，バックチャネルによる属性情報の送信状況を確認するための機能を IdP に追加する．さらに，その機能を用いて，その後の属性送信を拒否することも可能とする（図 3）．

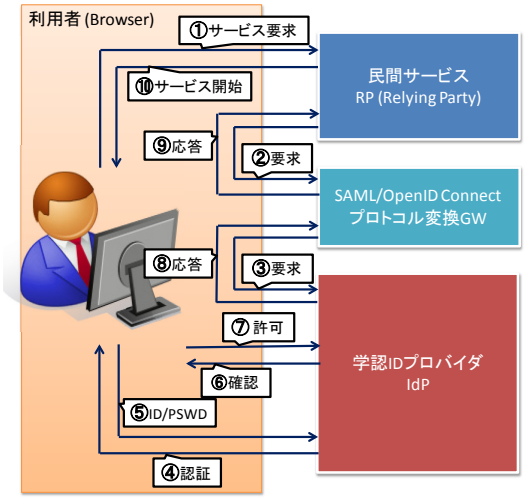

図 2　プロトコルゲートウェイを使用した処理フロー
Figure 2　Whole Flow with Proposed Protocol Gateway

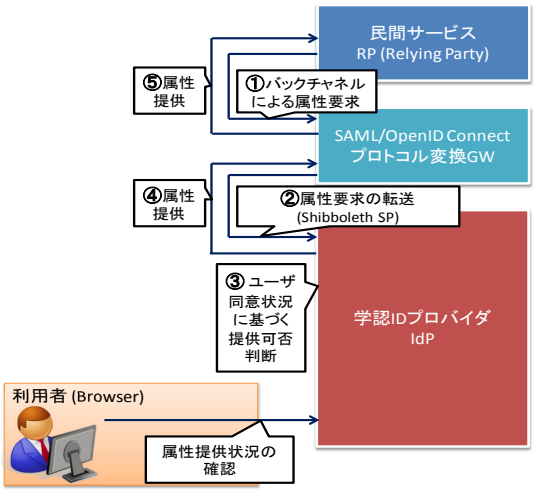

図 3　バックチャネルによる属性送信状況の確認
Figure 3　Checking Status of Attribute Release with Backchannel

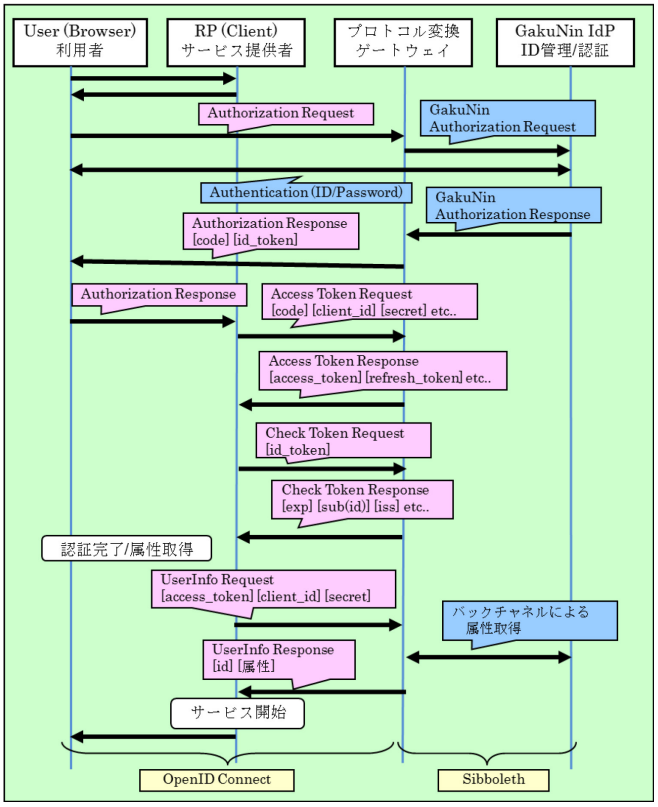

図 4　ゲートウェイにおける処理のフロー
Figure 4　Process flow at the Protocol Gateway

表 1　ゲートウェイに実装した OpenID Connect の機能
Table 1　Implemented Functions in OpenID Connect Specification

手順	引数
Authorization	client_id: require redirect_uri: require response_type: support = “token”, “code”, “id_token” scope: require = “openid” support = “eduPersonAffiliation”, “PPID” state: optional nonce: optional
Access Token (“code”)	client_id: require client_secret: require redirect_uri: require code: require grant_type: require = “authorization_code”
Access Token (“refresh_token”)	client_id: require client_secret: require redirect_uri: require refresh_token: require grant_type: require = “refresh_token”
Check Session	id_token: require
Userinfo	access_token: require

4.2 ゲートウェイ

ゲートウェイでは，OpenID Connect による RP からの要求を受け付け，それを SAML に変換して IdP に要求を伝える形でプロトコル変換を実現する．ゲートウェイは，SAML IdP に対して，同期した属性情報送信と，非同期の属性情報送信の両方に対応する．

OpenID Connect における属性情報の取得は，いわゆるバックチャネル的方式によって行われる．OpenID Connect (OAuth 2.0) において属性情報の取得を行うには認証の結果返される access_token が必要となる．また，access_token と共に返される refresh_token を用いることで access_token の再取得が可能となっている．access_token の有効期間は比較的短いが，継続的なアクセスを実現するために，refresh_token を用いた更新機能が提供されている．プロトコルゲートウェイには，RP からのアクセス頻度に応じて，適当な有効期限を設定したこれらのトークンを発行する機能を持たせる．

まず，同期した属性情報提供時の具体的な認証フローを説明する（図 4）．最初に，プロトコルゲートウェイでは受け取った RP からの OpenID Connect による Authrization 要求を SAML の Authrization 要求にプロトコル変換を行い，SAML の IdP に対する認証を要求する．SAML の認証が完了しアサーションを受け取ると，それに含まれる属性情報を保持した上で，RP には OpenID Connect の Authrization 応答として code と id_token がユーザのブラウザを経由して返される（今回の実装では response_type として”code id_token” を指定している）．RP は，受け取った code と登録済みのクライアント情報に含まれる Token エンドポイントを用いて access_token と refresh_token, および ePTID に対応するユーザ識別子である PPID (Pairwise Pseudonymous Identifier)を取得する．さらに RP では，OpenID Connect の Check Token エンドポイント（OpenID Connect の現仕様には含まれない独自機能）により id_token を検証することで正しい接続である事を確認する．RP が access_token を正しく取得できたならば，後は OpenID Connect の UserInfo エンドポイントより属性を取得することで処理が完了する．UserInfo エンドポイントを用いた操作以降は OpenID Connect（正確には OAuth 2.0）の標準的な操作である．

一度，ゲートウェイが IdP に対して認証を行った後は，ゲートウェイにおいて取得したトークンと対応する ePTID をデータベースに保存しておくことで，その後の非同期な属性情報取得に備える．RP より access_token（さらに必要に応じて refresh_token による access_token の再取得）を用いた非同期の属性情報取得要求が来れば，ユーザ認証なしに属性情報の取得が可能になる．

IdP においては，非同期の属性情報取得を実現するために，ePTID を識別子とした属性情報の取得に対応するよう設定を行う．これについては，既存の Shibboleth IdP において設定の調整のみで対応可能である．

今回の実装では，access_token の有効期限は 1 時間としている．一方，refresh_token の有効期限はある程度長くとる必要があるため，今回はデフォルトで 32 日間としている．これは，サービス登録時に指定することで変更可能となっている．

今回のゲートウェイ実装は，Ruby 1.9.2p290，Rails 3.2.12，SQLite 3.6.20 を用いて構築した．OpenID Connect の仕様のうち実際に実装したものを表 1 に示す．

4.3 バックチャネル属性要求ハンドラ

プロトコルゲートウェイは，SAML における SP の機能を持ち，Shibboleth SP を利用して構築しているが，Shibboleth 標準の SP では，バックチャネルによる非同期な属性情報取得のためのハンドラ（インタフェース）を持たないため，プロトコルゲートウェイの一部として呼び出すことが可能な，属性情報要求ハンドラを新たに用意した．属性情報要求ハンドラは既存の他の Shibboleth SP のハンドラと同様に，以下に示すような URL をハンドラのパスとしてアクセスすることで呼び出される．このパスは，設定ファイル shibboleth2.xml の<Handler>要素内の Location 属性にて変更可能である．

https://sp.example.ac.jp/Shibboleth.sso/AttributeQuery?entityID=https%3A%2F%2Fidp.example.ac.jp%2Fidp%2Fshibboleth&nameId=XXXXXXXXXXXXXXXXXXXXXXXXXXXX%3D

属性情報要求ハンドラは，IdP に対する AttributeQuery を行うにあたり，表 2 に示すパラメータを受け取る．

表 2　属性情報要求ハンドラが受け取るパラメータ
Table 2　Parameters of handler for AttributeQuery

パラメータ	内容
protocol	メタデータの IdP Role で設定されている protocolSupportEnumerationの値
entityID	IdPのentityID（必須）
format	SAMLのユーザ識別子のフォーマット
nameId	SAMLのユーザ識別子（ePTID，必須）

属性情報要求ハンドラは，以下に示す要領で処理を行い応答を返す．

1. entityID で指定された IdP に対して nameId に紐付けられたユーザに関する AttributeQuery を行い，SAML Response を受け取る．
2. SAML Response に含まれる AttributeStatement を JSON 形式に変換する．JSON オブジェクトのキーおよび値は， attribute-map.xml および attribute-policy.xml の定義に従う．もし，AttributeStatement が含まれていない場合は，空の JSON({})とする．
3. 一つの属性が複数の属性値をもつ場合は，属性値をセミコロン(;)で連結する．
4. 属性値自身にセミコロン(;)が含まれる場合は，上述の区切りのセミコロンと区別するために，バックスラッシュ(¥)でエスケープする．
5. 得られた JSON オブジェクトをハンドラのレスポンスとして返す．

表 3　ユーザに対して求める同意の選択肢
Table 3　Choices of User Consent for Release of Attributes

種別，略称	メッセージ
(a) 毎回確認	サービスに送信する情報を毎回確認します. 今回は情報を送信することに同意します.
(b) 保存する	次回からこのサービスではこの画面を表示しません. 今後このサービスに対して同一の情報を自動的に送信することに同意します.
(c) 表示しない	この画面をもう表示しません. ユーザ情報を今後すべてのサービスに対して自動的に送信することに同意します. 送信する情報は表示以外のものを含む可能性があります.

表 4　非同期の属性情報送信を考慮した同意選択肢の表現
Table 4　Revised Choice of (b) to Support Asynchronous Back-channel Attribute Query

種別，略称	メッセージ
(b) 保存する	次回からこのサービスではこの画面を表示しません. 属性情報に変化がない限り，今後このサービスに対して今回と同一の情報を自動的に送信することに同意します. また，サービスからの問合せに対しても，今回と同一の情報を自動的に送信することに同意します.

4.4 バックチャネルによる属性送信を考慮したユーザ同意の取得

IdP における属性情報送信に関してユーザの同意を得るためには，IdP のプラグインである uApprove を利用することができる．しかし，このプラグインはユーザ認証と同期した属性情報の送信にしか対応していないため，非同期の属性情報送信を考慮した同意の選択肢を用意するとともに，ユーザによる選択に対応した属性情報の送信機能を提供する必要がある．既存の uApprove をそのまま利用すると，送信先として同意していない SP からの要求に対しても応答を返してしまうという問題がある．

ユーザに対して提示する選択肢は，既存の uApprove を日本語対応した uApprove.jp においては表 3 に示すような内容となっている．このうち(b)について，非同期の属性情報送信を考慮した表現に改めた（表 4）．

ユーザが(a) を選択した場合，ユーザが同意した属性情報は同期したログイン時にだけ送信され，AttributeQuery への応答では一切の属性を送信しない．従って，非同期の

属性情報送信を用いるサービスは利用できない．(b) を選択した場合，送信に同意した属性情報は将来の照合のために，ストレージに保存される．送信を同意した SP からの属性情報要求があった際に，保存してある属性情報と比較を行い，一致しているものについてのみ，SP に応答する．これは，取得した最新の属性値がユーザの同意した時点の属性値と異なっている場合，ユーザが送信に同意した情報とは見なせないためである．このような場合は，ユーザが次回のログイン時に再同意しない限り，変更のあった属性の属性情報を提供することができない．IdP 側における非同期の属性情報送信では，このような形でプライバシー保護に配慮する．

4.5 属性情報送信状況の確認機能

非同期での属性情報送信では，ユーザに属性情報が送信されるタイミングが分からないことから，ユーザが定期的に属性情報の送信状況について確認できる手段を別途提供することが望ましい．

本試作では，属性情報の提供状況の確認のために，IdP 上にユーザが参照可能な属性送信済み SP 一覧ページを用意した．属性送信済み SP 一覧ページは，IdP 上に，SP としての機能として実現しているため，ユーザが IdP において認証を要求されるタイミングではなく，直接当該ページにアクセスする形で参照することができるようになっている．実装としては，IdP のプラグインではなく，JSP による独立したアプリケーションとなっている．このページは，この SP には当該 IdP にアカウントを持つユーザのみしかアクセスしないため，認証フェデレーションには参加する必要はない．

ユーザに提示される情報を表 5 に示す．サービス名は entityID やメタデータに定義されたサービスの表示名，属性送出同意日時はユーザが属性を自動的に送信することに同意して「送信」ボタンを押したときの時刻，バックチャネルによる最新属性取得日時および属性取得回数は，SP が IdP の AttributeQuery プロファイル用いて属性情報を取得したときの時刻と，その通算回数を示す．表示対象となる SP は，前述のユーザ認証時の属性情報送信同意の選択肢において，「(b) 保存する」または「(c) 表示しない」を選択したものとなる．「(c) 表示しない」を選択した場合は，個別の SP 名ではなく「すべてのサービス」としてまとめられた 1 件だけが表示される．「(a) 毎回確認」を選択した場合は，バックチャネルによる属性送信には同意していないことになるため，一覧には表示されない．IdP では，これらの情報をユーザに提供するために，データベースにアクセス記録を格納する．

属性送信済みSP 一覧ページには，表示したSP ごとに，同意を取り消すための削除ボタンが用意されており，個々のSPごとに同意を取消すことが可能となっている．また，すべての同意を一括で取消すためのボタンも用意している．

表 5　属性情報送信状況としてユーザに提示される情報
Table 5　Information Shown as Status of Released Attributes

項目	値
サービス名	文字列
エンティティID	文字列
属性送出同意日時	日時
バックチャネルによる最新属性取得日時	日時
バックチャネルによる属性取得回数	数値

5. おわりに

本報告では，SAML を用いて認証連携を行う学術フェデレーションと，OpenID を用いて認証連携を行う民間のフェデレーションとが連携し，学術機関が提供するユーザに関する属性情報，その中でも特に身分（学生，教員，職員等）の情報を活用して，民間による学割サービスを提供するための仕組みを設計し試験実装を行った．認証フェデレーションの枠組みを活用することで，ユーザの自己申告によらない確実な情報を提供することができ，サービス提供側も安心して割引等の措置を実施することが可能となる．提供される情報はプライバシーにもかかわるものとなる可能性もあるため，このような連携を実現する際に義務づけられるポリシーについても平行して検討し，実サービスの実現につないでいきたいと考えている．

謝辞　本報告の内容は，総務省「戦略的国際連携型研究開発推進事業」（平成 24 年度，情報セキュリティに関する研究開発課題の委託）による支援を受けて「情報流通連携のためのオープンな ID 連携プラットフォームにおけるプライバシー保護機能の高度化の研究開発」として実施したものの一部である．

参考文献

1) S. Cantor, J. Kemp, R. Philpott, and E. Maler ed., "Security Assertion Markup Language (SAML) V2.0," http://saml.xml.org/saml-specifications, March 2005.
2) OpenID Foundation, “OpenID Foundation website,” http://openid.net/, last visited Apr. 1, 2013.
3) REFEDS (Research and Education Federations), “REFEDS Federation Survey”, https://refeds.terena.org/index.php/Federations, last visited Apr. 1, 2013.
4) 西村健, 中村素典, 山地一禎, 大谷誠, 岡部寿男, 曽根原登: 日本における学術認証フェデレーションとその役割および効果, 信学技法, Vol. 111 No. 375, IA2011-55 pp.5-8, 2012.
5) 島岡政基, 西村健, 古村隆明, 中村素典, 佐藤周行, 岡部寿男, 曽根原登: 学術機関のためのサーバ証明書発行フレームワーク (ネットワーク管理・オペレーション, <特集>若手研究者のためのフロンティア論文), 電子情報通信学会論文誌, Vol.J54-B, No.7,

pp.871-882, 2012.
6) OpenID Foundation, “Surprise! You may already have an OpenID”, http://openid.net/get-an-openid/, last visited Apr. 1, 2013.
7) Google Apps for Education, http://www.google.com/intl/ja/enterprise/apps/education/, last visited Apr. 1, 2013.
8) マイクロソフト, “Office 365 導入事例”, http://www.microsoft.com/ja-jp/office/365/showcase.aspx, last visited Apr. 1, 2013.
9) Yahoo! Japan, “Yahoo!メール Academic Edition”, http://business.yahoo.co.jp/ymacademic/, last visited Apr. 1, 2013.
10) M. Wahl, T. Howes, S. Kille, “Lightweight Directory Access Protocol (v3),” The Internet Society, RFC2251, 1997.
11) Shibboleth Consortium, http://shibboleth.net/, last visited Apr. 1, 2013.
12) SimpleSAMLphp, http://simplesamlphp.org/, last visited Apr. 1, 2013.
13) 国立情報学研究所, 学術認証フェデレーション システム運用基準 (Ver.1.2), 2011.
14) Internet2, eduPerson & eduOrg Object Classes, http://middleware.internet2.edu/eduperson/,
15) OpenID Foundation, “OpenID Authentication 2.0 - Final,” 2007.
16) OpenID Foundation, “OpenID Attribute Exchange 1.0 – Final,” 2007.
17) D. Hardt, Ed., “The OAuth 2.0 Authorization Framework,” RFC6749, Internet Engineering Task Force (IETF), 2012.
18) OpenID Foundation, “Welcome to OpenID Connect,” http://openid.net/connect/, last vidited Apr. 1, 2013.
19) Yahoo! Japan, “YConnect(OAuth2.0/OpenID Connect)をリリースしました！,” http://techblog.yahoo.co.jp/web/auth/yconnect/, 2012.
20) 総務省, 個人情報の保護に関する法律, 法令データ提供システム, http://law.e-gov.go.jp/htmldata/H15/H15HO057.html, 2003.
21) 総務省, 独立行政法人等の保有する個人情報の保護に関する法律, 法令データ提供システム, http://law.e-gov.go.jp/htmldata/H15/H15HO059.html, 2003.
22) L. Hämmerle, "SWITCHaai: shibboleth-based federated identity management in Switzerland," Proceedings of CESNET 2006 Conference, 2006.
23) The SWITCH Foundation, "uApprove," http://www.switch.ch/aai/support/tools/uApprove.html, last visited Apr. 1,2013.
24) Tananun Orawiwattanakul, Kazutsuna Yamaji, Motonori Nakamura, Toshiyuki Kataoka , Noboru Sonehara: User Consent Acquisition System for Japanese Shibboleth-based Academic Federation (GakuNin), International Journal of Grid and Utility Computing (IJGUC), Vol. 2, No. 4, pp. 284-294, 10.1504/IJGUC.2011.042944, 2011.
25) 国立情報学研究所, “産学の ID をつなぐ世界初のトラストフレームワークの研究に着手 ～利用者情報の安全な流通を目指し，学生向けサービスの提供を支援～”, http://www.nii.ac.jp/news/2011/0305/, 2012.
26) UQ コミュニケーションズ, “モバイル WiMAX キャンパスネットワーク接続サービス”, http://www.uqwimax.jp/service/corporate/campusconnect.html, last visited Apr. 1, 2013.