# 属性交換における属性値保証

柿崎 淑郎 †　　前田 千徳 †　　岩村 惠市 †

† 東京理科大学
102-0073 東京都千代田区九段北 1-14-6
{kakizaki, maeda, iwamura}@sec.ee.kagu.tus.ac.jp

**あらまし** 属性情報は，名前，メールアドレスなどのほか，権限，職責，資格，地位などがあり，Web サービスなどにおける認可の根拠として用いられる．近年ではシングルサインオンにおける属性交換によって，利用者の属性情報を用いたサービスが増加している．しかし，属性交換によって得られた属性値が正確であるか，利用時点で有効であるかなどの保証については十分に検討されていない．本稿では，属性交換における属性値保証について考察し，属性情報を用いたサービスに対する影響と対策を検討する．

# Attributes Assurance in Attribute Exchange

KAKIZAKI Yoshio†　　MAEDA Kazunari†　　IWAMURA Keiichi†

†Tokyo University of Science
1-14-6 Kudankita, Chiyoda-ku, Tokyo 102-0073 JAPAN
{kakizaki, maeda, iwamura}@sec.ee.kagu.tus.ac.jp

**Abstract** Attribute information is that shows the authority, the responsibility, the qualification, and the position, etc., and is used as grounds of authorization in Web services. Recently, service that uses user's attribute information increases by attribute exchange in single sign-on. However, it is not paid attention enough whether the attributes, obtained by attribute exchange, are accurate from the view point of the assurance at the time of use. In this paper, we discuss about the attributes assurance in attribute exchange, and consider the influence and measures against services that use attributes.

## 1　はじめに

Web サービスの増加に伴い，利用者が記憶すべき ID・パスワード対も増加の一途を辿っており，アイデンティティ管理の必要性が高まっている．利用者がアイデンティティ管理の恩恵を受ける技術として，シングルサインオン（以下 SSO とする）がある．SSO は 1 つの ID・パスワード対で，複数の Web サービスへのログインを可能とする技術であり，OpenID[1] や Shibboleth[2] などがある．

SSO によって認証の問題は解決されるが，アクセス制御のためには権限などによる認可が必要となる．SSO 環境下で，権限や資格などの属性情報をやり取りすることを属性交換（Attribute Exchange）という．OpenID などでは，SSO 時に属性交換も合わせて行い，利用者に対して適切なアクセス制御を行うことが可能である．

一般的に，属性交換に備えて，SSO 認証サーバは利用者の属性情報を集約して管理する．しかし，SSO 認証サーバは利用者を認証することはできても，権限や資格などの多様な属性情報

を全て認証・検証することは困難である．属性情報によっては，SSO 認証サーバが認証・検証できる場合もあるが，それ以外の属性情報においては，現状の属性値を確認できずに利用している場合や利用者の主張を単純に受け入れている場合などが考えられる．そのため，SSO 認証サーバから属性交換される属性値の保証が問題となる．

本稿では，属性交換における属性値保証について考察する．属性情報は多様であるため，ある 1 つの主体が全ての属性値を保証することはできない．そのため，属性値を認証・検証することができる主体との連携が必要となる．そこで，属性値保証のレベルと属性値確認の方法を挙げ，それぞれが属性情報を用いたサービスに与える影響とその対策について検討する．

## 2 関連研究

保証レベルについては NIST の SP 800-63[3] がある．SP 800-63 では，4 段階の保証レベルとその技術的要求事項を示し，アイデンティティの登録と発行，トークン，トークンとクレデンシャルの管理，認証プロトコル，アサーションの各分野における保証レベルを示している．保証レベルはレベル 1〜4 まで規定されており，レベル 1 が最も低く，レベル 4 が最も高い保証レベルである．

また，カンターライニシアティブでは SP 800-63 を基にして，Identity Assurance Framework v2.0 (IAF) [4] を策定している．IAF では保証レベル毎に事業者が満たすべき要件，事業者に対する監査要件を規定することで，シングルサインオンを行う事業者間における，信頼性の相互確認を簡素化している．

文献 [5, 6] では属性の保証レベルについて触れ，アクセス制御の決定に必要な選択肢の提供を目的として，基本的なトラストレベルだけでなく，評価に必要なメタデータを提供する方式を提案している．また，SAML2.0 を拡張して，レイヤー化されたトラストモデルでこれを実現している．

文献 [7] では，連携アイデンティティ管理の問題点として，以下の 4 つが挙げられている．

1. Limitation to web services
2. Persistent data storage
3. Federation security and privacy control
4. Syntax and semantics of attributes

2 番目では，アイデンティティプロバイダとサービスプロバイダ間での属性交換において，サービスプロバイダは，受け取った SAML アサーションが一定時間経過後に，まだ有効であるかどうかの再保証をアイデンティティプロバイダに要求することができない点を指摘している．

## 3 プレイヤ

本稿で用いるプレイヤを以下に説明する．

**AA**　属性認証局（Attribute Authority）であり，ユーザの属性情報を認証し，登録して，電子的に利用可能な状態にする．AA は信頼できる第三者機関（TTP）とする．

**AP**　属性プロバイダ（Attribute Provider）であり，ユーザに関する属性情報を集約し，一元管理する主体である．AA とは異なり，ユーザの属性情報を認証したり，検証したりすることはできない．AP は TTP とする．

**IdP**　ID プロバイダ（Identity Provider）であり，ユーザを認証する主体であり，シングルサインオンにおける認証サーバと同等である．IdP は TTP とする．IdP と AP は異なる主体も想定できるが，本稿では説明簡略化のために，AP は IdP の機能を持つ同一主体とする．

**SP**　サービス提供者（Service Provider）であり，AP からユーザの属性情報を取得し，ユーザに対してサービスを提供する主体である．

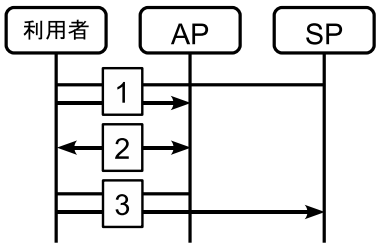


図 1: 属性交換の流れ

**利用者**　利用者は IdP で認証され，その認証結果をもって SP のサービスを利用する主体である．利用者の属性情報は AA が認証し，その属性値は AP に集約され，管理されている．

## 4　属性交換

属性交換とは，利用者の属性情報を主体間で安全に交換することをいう．属性情報には，名前，性別，メールアドレス，住所，電話番号などのほか，権限，職責，資格，地位などがあり，Web サービスなどにおける認可の根拠として用いられる．

属性交換を実現する既存技術として，OpenID では，AX (Attribute eXchange)[8]，Connect UserInfo[9] などがある．また，Shibboleth[2] では，SAML による属性交換が行われている．

図 1 に本稿における属性交換の流れを示す．ここでは説明簡略化のために，利用者は IdP によって認証済みとする．

1. SP は取得したい属性情報のリストをつけて，利用者を AP に転送する．
2. AP は SP から要求された属性情報を抽出し，利用者に対して提供可否を確認する．
3. 利用者が提供を許可した場合，AP は利用者を介して提供する属性情報を SP に送る．

SP は上記手順によって，利用者の属性情報を AP から取得する．SP は取得した属性情報によって，利用者を認可して，アクセス制御を行う．

## 5　属性値の保証

本稿では，AA で登録された属性情報を AP に集約し，SP は AP から属性交換を行い，利用することを想定する．このとき，問題とするのは，AP が持つ属性値の保証である．

AP は AA とは異なり，ユーザの属性情報を認証したり，検証したりすることはできない．そのため，AP が管理する属性値は，AA から取得した時点においては保証されているが，それ以降の任意の時点においても同様に保証されているとは限らない．例えば，引っ越しによって住所が変わった場合，AP はその変更を知らない限り，以前の住所を属性値として管理し続ける．この問題は以下の 2 つによって引き起こされる．

**問題点 1**　AP は属性値が正しいかを検証することができない．

**問題点 2**　属性値の変更に際して，AA から AP へ通知する手段がない [7]．

**問題点 1** は属性情報の多様性が原因となる．例えば，資産残高証明と在籍証明を同時に行える主体は存在しない．これは資産を管理する主体と勤務先が異なる主体であるからである．このように，多様な属性情報に対して，AP がその属性値を検証することは不可能である．

そうであれば，**問題点 2** を解決することで，属性値保証の問題を解決することとなる．**問題点 2** では，情報開示手段とその許可が問題となる．一般的に，属性交換は利用者が許可した範囲でのみ行われる．例えば，OpenID AX の場合，AP と SP の間での属性交換には利用者が介在し，利用者の許可と監視の中で行われる．また，SP が AP の管理する属性情報を取得する場合，属性交換は PULL 型で行われるため，AP は SP に対して PUSH する手段を持たない．そのため，属性値が変化した場合には，AP が SP に対して PUSH するか，または SP が AP に対して必要時に PULL しなくてはならない．

属性値の保証は以下の分類ができる．

**保証 1**　利用者の主張した属性値

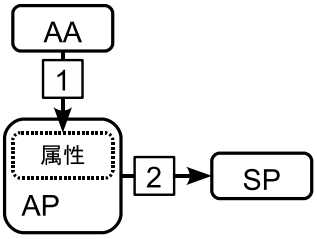


図 2: AP 集約

**保証 2** AA から属性交換した属性値

**保証 3** AA による属性証明書

**保証 4** AA による要求毎の属性値確認

このとき，**保証 1** が最も保証レベルが低く，**保証 4** が最も保証レベルが高い．**保証 1** は利用者の主張に基づいており，属性値は無保証である．**保証 2** は属性交換時点においては AA によって保証されているが，それ以降の任意時点における属性値は保証されない．**保証 3** は属性証明書の有効性を検証することで，属性値が保証される．この場合，属性証明書に限らず，有効期限内であれば属性値が保証される仕組みであれば良い．**保証 4** は常に属性値が保証されるが，AA ならびに利用者の手続きが煩雑であり，利便性に問題がある．

# 6 考察および検討

本章では属性値を保証する手法を挙げ，その正確性，保守性，処理コストについて考察を行う，

## 6.1 属性値保証手法

**AP 集約** 図 2 に属性情報を AP に集約する場合を示す．この方法では，AA が AP にアクセスして属性値を更新する．そのため，AP は AA に対して属性値の更新を許可する必要がある．AP は AA に対する更新許可のために，AP が管理する属性情報のエンドポイントとアクセスのための情報（トークン）を AA に渡す．AA は AP からのトークンを利用して，属性値に変更があった場合に，AP の属性値を更新する．AP

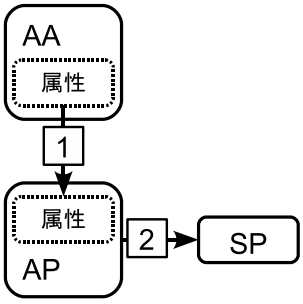


図 3: AP 中継

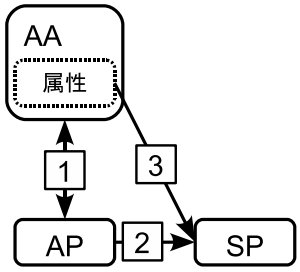


図 4: AA 集約

は自らが管理する属性情報を属性交換で SP に渡す．AA が適切に AP の属性値を更新していれば，SP が受け取る属性値は保証されている．

**AP 中継** 図 3 に属性情報を AP が中継する場合を示す．この方法では，AP が AA から属性値を得て AP の属性値を更新する．AP は SP からの属性交換要求毎か，または定期的な間隔で，AA から属性値を取得して，AP の属性値を更新する．AP から SP への属性交換は AP 集約と同じである．

**AA 集約** 図 4 に属性情報を AA に集約する場合を示す．この方法では，AP は AA が管理する属性情報のエンドポイントとアクセスのための情報（トークン）を AA から取得する．AP はエンドポイントとトークンを SP に渡し，SP は AA のエンドポイントにトークンを用いてアクセスし，属性情報を取得する．

## 6.2 正確性

属性交換された属性値が最新の属性値を反映しているかどうかを正確性とする．

表 1: 正確性，保守性，処理コストの比較

| | 正確性 | 保守性 | 処理コスト |
|---|---|---|---|
| AP 集約 | ○ | ○ | AA |
| AP 中継 | AP の更新頻度による | ○ | AP |
| AA 集約 | ○ | × | AA, AP, SP |

AP 集約では属性値が変更されたタイミングで，AA が AP に対して，属性値の更新を行う．AA 集約では SP が AA から直接に属性値を取得する．そのため，AP 集約と AA 集約は属性値が変更されたタイミングで，最新の属性値になるので，属性交換される属性値は正確である．

AP 中継では，SP からの属性交換要求毎に AA から属性値を取得すれば，正確性は高い．しかし，定期的な間隔で更新を行う場合，正確性は必ずしも高くない．

### 6.3 保守性

属性交換を行う権限が変更された場合の手間を保守性とする．

AP 集約では AP は AA のアクセス権限を変更することで，対応できる．

AP 中継では全役割を AP が担っているため，属性交換の権限が変更された場合においても，集中的に即時対応できる．

AA 集約では，SP が AA にアクセスするためのトークンを失効させる必要がある．しかし，トークンを発行しているのは AA であるが，SP への属性交換を指示するのは AP であるため，管理の統一性がなく，保守性に問題がある．

### 6.4 処理コスト

属性値が変更された場合における更新の手間を処理コストとする．

AP 集約では属性値の更新は AA が行うため，AP および SP に追加の処理コストは発生しない．

AA 集約では適切なタイミングで AP が AA から属性値を取得する必要がある．そのため，追加の処理コストは AP のみに発生する．また，SP の属性交換要求毎に取得すれば，属性値の保証は高いが，AP の処理コストは高くなる．

AA 集約では 3 者全てに処理コストが発生する．AA は SP がアクセスするためのトークンを発行・管理する必要があり，AP は AP からトークンを受け取り，SP へ渡す役割があり，SP は AP からトークンを受け取り，AA から属性値を取得する処理コストが発生する．また，6.3 節で述べたように，アクセス権限が変更された場合における処理コストも発生する．

### 6.5 まとめ

正確性，保守性，処理コストの結果を表 1 にまとめた．表 1 より，3 手法の比較では，AA 集約には解決すべき問題が多くあり，AA が処理コストを負担できれば，AP 集約が良い結果となる．

文献 [7] でも指摘されているように，属性値保証の問題は，最新の属性値に更新できない点にある．AP 集約手法は AP から AA に更新のためのトークンを渡すことで，AA が AP の属性値を更新できるようにすることで，この問題を解決している．

また，適切な有効期限の設定と失効管理が必要になるが，属性証明書などの証明書を利用することで，ある一定期間は属性値を保証することができる．

## 7 おわりに

本稿では，属性交換における属性値保証の問題について考察した．属性情報は多様であるため，ある 1 つの主体が全ての属性値を保証することはできない．そのため，属性交換によって得られた属性値を用いるサービスにおいては，

属性交換される属性値の保証が重要となる．属性値保証のレベルと属性値確認の方法を挙げ，それぞれが属性情報を用いたサービスに与える影響とその対策について検討した．

## 参考文献


[1] David Recordon and Drummond Reed. OpenID 2.0: a platform for user-centric identity management. In *Proc. of the second ACM workshop on Digital identity management (DIM '06)*, pp. 11–16, ACM, doi:10.1145/1179529.1179532, 2006.

[2] Shibboleth. `http://shibboleth.internet2.edu/`.

[3] William E Burr, Donna F Dodson, Elaine M Newton, Ray A Perlner, and W Timothy Polk. Electronic Authentication Guideline, SP 800-63-1 (draft3), 2011.

[4] Kantara Initiative. Identity Assurance Framework v2.0, `http://kantarainitiative.org/confluence/display/GI/Identity+Assurance+Framework+v2.0`, 2010.

[5] Ivonne Thomas and Christoph Meinel. Enhancing Claim-Based Identity Management by Adding a Credibility Level to the Notion of Claims. In *Proc. of 2009 IEEE International Conference on Services Computing*, pp. 243–250, IEEE, doi:10.1109/SCC.2009.66, 2009.

[6] Ivonne Thomas and Christoph Meinel. An Identity Provider to manage Reliable Digital Identities for SOA and the Web. In *Proc. of the 9th Symposium on Identity and Trust on the Internet - IDTRUST '10*, pp. 26–36, ACM, doi:10.1145/1750389.1750393, 2010.

[7] Wolfgang Hommel and Helmut Reiser. Federated Identity Management: Shortcomings of existing standards. In *Proc. of 9th IFIP/IEEE International Symposium on Integrated Network Management (IM2005)*, doi:10.1.1.147.2469, 2005.

[8] OpenID Attribute Exchange 1.0, `http://openid.net/specs/openid-attribute-exchange-1_0.html`, 2007.

[9] OpenID Connect UserInfo 1.0, `http://openid.net/specs/openid-connect-userinfo-1_0.html`, 2011.