L-036

分散認証基盤を活用したプライベート情報交換アーキテクチャの提案

今村理* 半井明大* 大澤由憲* 武田敦志† 北形元‡* 白鳥則郎‡ 橋本和夫*
*東北大学情報科学研究科 †東北学院大学教養学部情報科学科 ‡東北大学電気通信研究所

1 はじめに

近年，サービスの展開に PI(Private Information:プライベート情報) を用いることが一般的になっている．従来は Web サービスで PI が用いられることが多かったが，今後のユビキタス環境の発展により，あらゆる場所のコンピュータで PI を用いるサービスが提供されると想定される．

例えば，レストランを利用する時に PI の活用が考えられる．レストランで行う意思決定には，座席の決定と注文するメニューの決定がある．レストランに入店した際の座席の決定では，喫煙の有無や，窓際の席・奥の席・カウンター席・テーブル席・静かな席・にぎやかな席といった所への嗜好が影響する．メニューの注文の際には，好き嫌い・アレルギー・宗教・健康への配慮の必要性等が影響する．これらの意思決定を PI によってサポートすることで，より良いサービスの提供が可能になる．

ユビキタス環境で PI の利用を実現するには，レストランの備え付け端末のようなユーザが初めて用いる端末でも利用可能で，かつ細かな嗜好情報等のむやみには公開したくない情報も取り扱える PI 交換システムが必要である．さらに，レストランの端末のように本人の端末ほどの信頼性のない端末でも利用可能である必要がある．

そして，地域のレストランのように小規模なサービスプロバイダ (SP) でも PI が利用可能にならなければならない．従来の PI の取得・管理方法ではシステムの導入コストが高いことが多かったが，地域限定サービスや医療福祉サービス等でも利用できるようにする必要がある．

一方で，人権の一つとして認められている PI コントロール権は保護する必要がある．PI は本人の意思に従って運用されなければならない．

本論文では，PI コントロール権を保護しつつ，様々なコンピュータから利用可能で，小規模 SP でも利用できる PI 交換アーキテクチャについて論じる．これらの観点から既存研究を整理した上で，新たな PI 交換アーキテクチャを提案する．

2 関連研究

2.1 シングル・サインオンと PI 交換システム

ユーザ認証が必要なサービスが増えるに伴ない，ユーザが自身の ID・パスワードを適切に管理することが困難となり，利便性やセキュリティ面で問題となってきた．そのため，一組の ID・パスワードによって全てのサービスを利用できるようにするシングル・サインオンの研究が盛んとなった．

Shibboleth[1] は，統一された運用ポリシーのもとで複数の IdP(Identity Provider) がフェデレーションを作り，SP に対して単一の IdP のように振舞うことでシングル・サインオンを実現した．既存の認証基盤を連携させることで実現できるため SP に大きな変更が必要ないという利点があるが，参加する IdP や SP の密接な協力が必要という問題がある．そのため，参加 IdP 間の運用ポリシーが近く，IdP や SP の信頼性の確認も容易な大学間等でしか用いることができない．

Web のように，IdP 間の運用ポリシーや IdP・SP の信頼性が異なる状況で用いられるシングル・サインオンのシステムとしては，OpenID[2] が実用化されている．OpenID では，ユーザ端末が SP と IdP のインタフェースとなって認証情報を提供することで IdP 間の連携を省略している．しかし，IdP の信頼性の確認ができないという問題があり，普及はあまり進んでいない．

そして近年になって，認証情報のみならず PI が SP のサービス展開に重要となり，PI を交換するシステム

Private information exchange architecture based on distributed authentication infrastructure
Satoru IMAMURA*, Akihiro NAKARAI*, Yoshinori OSAWA*, Atsushi TAKEDA†, Gen KITAGATA*, Norio SHIRATORI‡, and Kazuo HASHIMOTO*
*Graduate School of Information Sciences, Tohoku University
†Department of Information Science, Tohoku Gakuin University
‡Research Institute of Electrical Communication, Tohoku University

が必要とされてきた．その要求を満たすため，OpenID Attribute Exchange[3] と呼ばれるプロトコルが提案された．それにより，IdP がプライベート情報を OpenID の認証情報とともに提供できるようになった．しかし，OpenID は IdP や SP の信頼性に対する考慮があまりなされておらず，PI を提供することを嫌がるユーザが多いため普及は進んでいない．

それに対して，個人で簡易に使えるシステムとして，Web フォームを自動的に埋める Sxipper[4] 等のソフトウェアがブラウザ・プラグインとして提供されている．しかし，プライバシーポリシーの考慮等 SP との連携が不可能で，便利なソフトウェアの枠を出ることができない．

一方で，Microsoft は.NET Framework3.0 の機能として Windows CardSpace[5] を提供した．CardSpace では，ユーザが財布の中に所属の異なる複数の名刺や社員証・クレジットカード等を入れておくように各 PI セットを管理し，ユーザクライアント中心の認証を提供する．さらにこのサービスでは，ユーザクライアント内に情報を保持して SP に提供する Personal Card と，IdP に認証を要求する Managed Card の 2 種類のカードを使い分けることで，PI を提供したいだけの場面とその証明を行いたい場面の両方に対応できるようになっている．

また，研究機関主導の次世代の PI 交換システムとして PRIME Architecture[6] が提案されている．PRIME プロジェクトでは，PI をユーザが安心して利用できるようにするためにはユーザの PI 開示ポリシーと SP の PI 利用ポリシーを定義し，PI 交換時にそれらのネゴシエーションを行うことが必要と認識された．PRIME Architecture では，SP とユーザクライアント双方に PRIME middleware を導入することで高度な PI 交換のネゴシエーションを可能にし，SP が取得した PI にラベルを付けて管理することでネゴシエーションで決められた利用ポリシーに従った運用が強制できる．

2.2 既存アーキテクチャの比較

既存の PI 交換アーキテクチャは，PI を保持している場所から外部 IdP 型とユーザ端末型に分類することができる．外部 IdP 型では，IdP が PI を保持し，ユーザもしくは SP が PI 利用時に IdP に PI を要求する．ユーザ端末型では，PI はユーザ本人のユーザ端末 (UT) に格納され，ユーザ端末が SP からの要求を受けて直接 PI を提供する．

外部 IdP 型のシステムとしては OpenID が挙げられ，OpenID は図 1a の構成をとる．SP から IdP に対して，ユーザ端末を介して PI 要求が行われる．この方式の利点は，PI の保管場所がサーバに固定であるため，信頼できる端末であれば利用するユーザ端末を変えても PI を利用できるという点である．しかし，ユーザが完全に IdP を信頼して PI を預けなければならないためユーザ不安の解消が難しく，また IdP が情報を漏洩した時のリスクが高い．

表 1: アーキテクチャの比較

アーキテクチャ	PI の管理者	PI の保管場所	ネゴシエーション機能
Shibboleth	× IdP	○固定	×なし
OpenID	× IdP	○固定	×なし
Sxipper	○ユーザ	×移動	×なし
CardSpace	○ユーザ	×移動	×なし
PRIME	○ユーザ	×移動	○あり
提案手法	○ユーザ	○固定	○あり

一方，ユーザ端末型のシステムとしては PRIME Architecture があり，図 1b の構成をとる．ユーザ端末型では，外部の IdP は存在せず，PI は自身のユーザ端末に保持される．そのため，プライバシー保護の観点では優れている．しかし，PI を利用できる環境は PI を保持している端末に限られてしまうという欠点がある．PI をユビキタス環境で用いるためにはモバイル端末等で持ち歩かなければならず，物理的な紛失の危険性や利便性の観点から好ましくない．

ここまで紹介したアーキテクチャを，PI の管理者と PI の保管場所，ネゴシエーション機能の有無についてまとめると表 1 となる．PI の管理者はプライバシー保護の観点からユーザ本人が望ましく，PI の保管場所は物理的な紛失の危険性や利便性の観点から固定である方が望ましい．IdP 型のアーキテクチャでは保管場所については優れているが PI の管理者については望ましくなく，ユーザ端末型のアーキテクチャでは逆の特徴を示している．また，ネゴシエーション機能については，PI コントロール権を保護しつつ，自律的な PI 交換を行うために必要な機能であると考えられる．

表 1 のように，従来のアーキテクチャでは PI の管理者をユーザとしたまま PI の保管場所を固定とすることができず，理想的な PI システムが実現できていなかった．本研究においてはそれを可能とし，さらにネゴシエーション機能によって PI のコントロールも可能にするアーキテクチャを提案する．

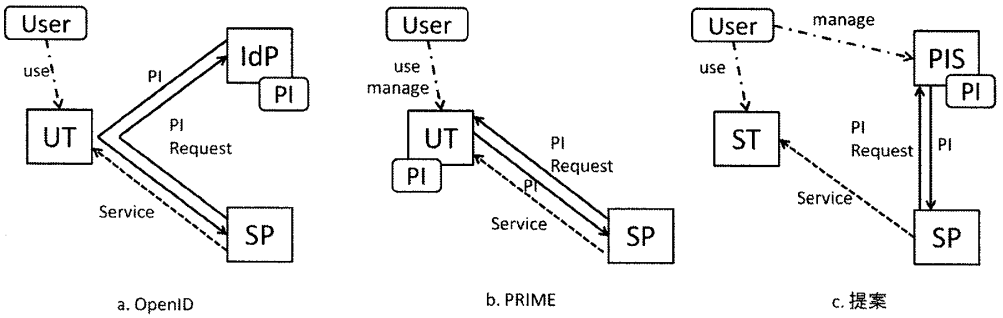

図 1: 各アーキテクチャの構成

3 提案手法

3.1 PIS(PI Server) の導入

本研究では，PI の管理者をユーザ本人としつつ PI の保管場所を固定とする PI 交換アーキテクチャを提案する．そのためにまず，その要求を満たす PI の保管場所として PIS(PI Server) を導入する．

PIS は，例えば家庭のブロードバンドルータに組み込むなどユーザサイドに設置するサーバで，SP にユーザ認証や PI を提供するパーソナル IdP の役割を行う．図 1c のように，ユーザ端末である ST(Service Terminal) を介さず SP に PI を提供する．ユーザ端末から離れた場所に PIS があることで，ひとつのユーザ端末に縛られることなく様々な端末から PI を利用することができる．また，PIS は，PRIME が備えるような PI ネゴシエーション機能を持ち，全ユーザが自身でこれを管理することで PI コントロール権が保護される．

しかし PIS を導入すると，いかにしてなりすまし攻撃を防ぎ認証を可能にするかという課題と，どのように PIS と SP や ST との間で連携するかという課題が生じる．前者の課題については，全 PIS に PKI による鍵認証を行えばこの課題は解決できるが，非常に多くのコストがかかり非現実的である．そのため，本研究では分散認証基盤を用いることで解決する．我々の開発した分散認証基盤 [7] を用いることにより，全 PIS の識別子 (PIS-ID) の一意性が保証できる．そして後者の課題について，次節で提案を行う．

3.2 提案アーキテクチャ

本研究では，図 2 のアーキテクチャを提案する．提案アーキテクチャは，大きく分けて，DAI(Distributed Authentication Infrastracture:分散認証基盤)，PIS，PIASP(PI Application Service Provider)，ST(Service Terminal) から構成される．

DAI は PIS と PIASP の Communicator に対して，ID の一意性を示す認証機能と公開鍵を用いた暗号化通信を提供する．PIS はユーザの PI とその開示条件 (PI

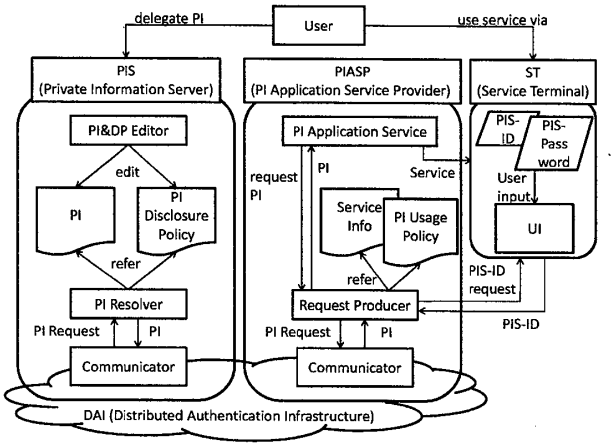

図 2: 提案アーキテクチャ

Disclosure Policy) を保持し，PIASP との PI 開示ネゴシエーションを行う PI Resolver を持つ．PIASP は，ユーザに提供されるサービスの本体である PI Application Service とそのサービス情報 (Service Info)，取得した PI の利用方法等を記述した PI Usage Policy，そして Service Info や PI Usage Policy を用いて PIS への PI の要求を PI Request としてまとめる Request Producer から構成される．ST は，PIASP からサービスを受ける機能とユーザが PIS-ID を入力するインタフェース (UI) を持つ．ユーザ (User) は，PIS に PI を委譲し，ST を介して PI を用いたサービスを受ける．

3.3 各モジュール間の連携

本節では，ユーザがレストランでメニューの推薦を受ける時を例として，提案アーキテクチャの各モジュール間の連携を説明する．

ユーザは，事前に PI&DP Editor を用いて，食べ物の嗜好等の PI と，その PI Disclosure Policy を “サービスタイプがレストランで仙台市の商店街に属するなら開示” というように登録する．レストラン (SP) 側は，“仙台市の商店街に属するレストラン” というような自身の情報を Service Info に登録する．PI Usage Policy

には，“PI はメニューの推薦のみに用い，アウトソーシングせず，取得後一日以内に削除する” というような PI 利用条件を記述する．

メニュー推薦サービス（PI Application Service）で PI が必要になると，ST からユーザの入力した PIS-ID が取得される．PIASP は Service Info と PI Usage Policy から PI Request をまとめ，PIS に PI を要求する．PIS では受け取った Service Info や PI Usage Policy を PI Disclosure Policy と比較し，一致すれば PI が PIASP に転送される．そして，PIASP から PI を利用したメニュー推薦サービスがユーザに提供される．

3.4 提案手法の特徴

提案手法の一番の特徴は，利用する ST を選ばないということである．PI の保管場所が固定であるため，どの ST からでもアクセスでき，ユビキタス環境での利用に適している．さらに，ST の信頼性の要求も従来のアーキテクチャと比較すると低い．PI の交換は ST を介さず PIS と PIASP の間でネゴシエーションに基づいて行われ，本人認証のために PIS-ID と PIS-Password の入力を行うが，それに関してもワンタイムパスワード等を用いれば保護できるためである．このような特徴を有しているにもかかわらず，PI の管理は完全にユーザ本人が行えるため PI コントロール権は強く保護されている．

さらに，PI 交換が PIS と PIASP 間で行われるため ST の行う処理や通信量は比較的少なく，バッテリ消費量が問題になるモバイルデバイスや太陽光発電で動作するデバイス等での利用にも適している．この点でも，あらゆる場所にコンピュータが設置されるユビキタス環境に適している．

また，IdP の分散と分散認証基盤の採用により，スケーラブルなアーキテクチャとなっていることも特徴である．理論的には全世界の人々が PIS を持ったとしても利用可能で，ユビキタス社会のインフラとしての機能を十分に果たすことができる．

3.5 実装

我々は，提案アーキテクチャの実現に向けた実装を進めている．DAI については [8] において，PI Resolver については [9] において実装を示した．今後は，その他のモジュールについても実装を行う予定である．

4 まとめ

本論文では，ユビキタス環境に適した PI 交換アーキテクチャについて論じた．

従来提案されていた PI 交換アーキテクチャでは，利用できるユーザ端末が限定されてしまい，シームレスに様々なコンピュータを使い分けるユビキタス環境に適していなかった．そのため，本論文では様々なユーザ端末においても利用できる PI 交換アーキテクチャを提案した．

PI コントロール権を保護するため，PI をユーザサイドに置きつつユーザ端末に信頼性を要求しないアーキテクチャを目指し，本研究ではこれをユーザの近くに PIS(PI Server) を導入することで実現した．PIS の導入に伴う外部からのセキュアなアクセス方法の課題やなりすましの問題は分散認証基盤の利用によって解決した．そして，PI 開示ポリシーによる PI の運用を可能にして PI 利用時のユーザ操作を低減し，様々なコンピュータからシームレスに利用できる PI 交換基盤を構築した．

謝辞 本研究の一部は，情報通信研究機構 (NICT) の委託研究「ダイナミックネットワーク技術の研究開発」の助成を受けて実施したものである．

参考文献

[1] Shibboleth, http://shibboleth.internet2.edu/

[2] OpenID, http://openid.net/

[3] OpenID Foundation, “OpenID Attribute Exchange 1.0,” http://openid.net/specs/openid-attribute-exchange-1_0.html

[4] Sxipper, http://www.sxipper.com/

[5] Microsoft, “Introducing Windows CardSpace,” http://msdn.microsoft.com/en-us/library/aa480189.aspx

[6] D. Sommer, et al., “PRIME Architecture V3,” https://www.prime-project.eu/prime_products/reports/arch/pub_del_D14.2.d_ec_WP14.2_v3_Final.pdf

[7] A. Takeda, et al., “A New Scalable Distributed Authentication for P2P Network and its Performance Evaluation,” WSEAS Transactions on COMPUTERS, vol.7, pp.1628-1637, 2008.

[8] A. Nakarai, et al., “An Overlay Authentication Network for Active Utilization of Private Information,” International Symposium on Applications and the Internet, 2010.

[9] Y. Osawa, et al., “A Proposal of Privacy Management Architecture,” International Symposium on Applications and the Internet, 2010.