情報処理学会論文誌　コンシューマ・デバイス & システム　Vol.7　No.3　1–14 (Sep. 2017)

コンシューマ・サービス論文

個人端末のコンテキストを使った本人性の検証

大神 渉[1,a)]　五味 秀仁[1,b)]

受付日 2017年2月28日, 採録日 2017年7月3日

概要：登録者が自身の情報を登録した後，利用者が登録者と同一であることを確認する本人確認を必要とするサービスが増えた．ID やパスワードなどを使う本人確認の手法では，攻撃者による情報の窃取や，複数の人物が結託をした場合，なりすましを防ぐことができない．一方，生体認証や公的機関の発行した書類を用いた本人確認手法では，なりすましをより強力に防ぐことができるが，ユーザやサービス提供者が情報の管理を通じてその確認の負担を負わなくてはならないという課題がある．そこで，本稿では，ユーザが所持する個人端末の情報を使った本人性の検証を行うことで，結託や窃取によるなりすましに強く，ユーザやサービス提供者の負担を軽減する本人確認手法を提案する．提案手法の実装や従来手法との評価を通じて，これらの手法により上記の課題が解決できることを示す．

キーワード：本人確認，結託，窃取，情報の管理

Identity Verification Using Personal Device's Context Information

Wataru Oogami[1,a)]　Hidehito Gomi[1,b)]

Received: February 28, 2017, Accepted: July 3, 2017

Abstract: Services requiring identification verification has increased. It is verifying method whether a user is the same as the registrant or not. In the identity verification method using a pair of ID and password, it is impossible to prevent spoofing by identity theft or collusion. On the other hand, it is possible to more strongly prevent these attacks using biometrics or documents issued by public institutions. However, there is a problem that a user or a service provider must bear the burden of verification through information management. In this paper, we propose an identity verification method that is strong against spoofing by collusion or stealing using the information of the personal terminal possessed by the user. Our method can also reduce burden on users and service providers. Through the implementation of the proposed method and the evaluation with the conventional method, we show that these problems can be solved.

Keywords: identity verification, spoofing by collusion, identity theft, information management

1. はじめに

ユーザが実世界でサービス提供を受ける際に，ネットワークを介したオンラインで登録した人物と同一である確認が必要なサービスが増えている．たとえば，電子チケットの販売では，不特定多数による利益目的の転売行為を防ぐために，オンラインで購入したチケットに対して当日会場でチケットを提示した人と購入者が一致しているか確認する．また，オンラインで銀行口座開設を受け付けるサービスでは，口座の犯罪転用を防ぐために，登録情報に基づくユーザ本人の実在やその身元を確認する．このように，目前の人物がオンラインの登録時と同じ人物であることを確認することを本人確認と呼ぶ．

一方，従来の本人確認の方法では，攻撃者によるなりすましの可能性があり，適切なユーザにサービスを提供できないことが問題である．

これらのサービスにおける本人確認では，ユーザの知識を用いて認証した結果（知識認証）を用いる方法と，ユーザの本人らしさを確認できる情報で本人性の検証を行う方法の 2 つが用いられている．知識認証を用いる例として，

[1] ヤフー株式会社
Yahoo Japan Corporation, Chiyoda, Tokyo 102–8282, Japan
[a)] wogami@yahoo-corp.jp
[b)] hgomi@yahoo-corp.jp

電子チケット販売では，ユーザが購入時に使った ID とパスワードを使った認証を行い，それが成功した場合，入場に必要なバーコードなどの情報が表示される．サービス提供者は，表示された情報を会場で読み取ることで本人確認を行っている [1]．この方法を用いると，表示された情報を提示可能なことから，ユーザが登録時と同じ知識（ID とパスワード）を持つことを確認できる．しかし，ID やパスワードの窃取や，金銭の授受を通じて別のユーザと結託が行われる場合，なりすましの可能性があり，本人であることを確認することは難しい．一方，本人性の検証を行う方法の例として，電子チケット販売ではユーザはオンラインを通じて顔写真を登録しておき，入場時に会場で照合を受けることで本人確認が実施されている [2]．この方法は，知識認証を用いる場合と比べて，他人が再利用しにくい情報を検証するため，窃取や結託に対する耐性がある．一方，事前に正確な情報の登録が求められ，ユーザとサービス提供者双方に登録時に負担を強いてしまう．たとえば，顔写真を登録する際，ユーザは光度や表情・向きなど複数の条件を満たす写真を撮影してサービス提供者に預ける必要がある．また，サービス提供者は写真がユーザと照合可能であることを担保するための確認作業や，顔写真というユーザ自身では変更が難しい情報の厳重な保管など，運用の負担が大きい．

このように，本人確認には以下の 3 つの課題があり，それらを並立して解決する手法が求められている．

(1) 窃取や結託によるなりすましに脆弱
(2) ユーザが情報を登録する負担の大きさ
(3) サービス提供者が情報を管理/運営する負担の大きさ

本稿では，ユーザが所持する個人端末の情報を使った本人性の検証を行うことで，結託や窃取によるなりすましに強く，ユーザやサービス提供者の負担も軽減する本人確認手法を提案する．

本稿の構成を以下に示す．まず，本稿が解決すべきモデルケースを示してから議論に必要な用語や条件を提示する（2 章）．次に，本人確認の問題点をなりすましに対する脆弱性，ユーザやサービス提供者負担として整理し（3 章），それらを解決する本人確認手法を提案する（4 章）．また，提案手法の実装を行い（5 章），結託と窃取によるなりすましに対し，従来の本人確認手法との比較を通じて評価を行った（6 章）．さらに，提案手法の適用や限界に触れて本研究の今後の展開を考察し（7 章），最後に関連する研究とその差分を述べる（8 章）．

2. モデルケースとその分析

この章では，問題を整理する前段階として本稿が解決する具体的なモデルケースを提示し，その分析を通じて用語およびそれらが満たす前提条件を述べる．

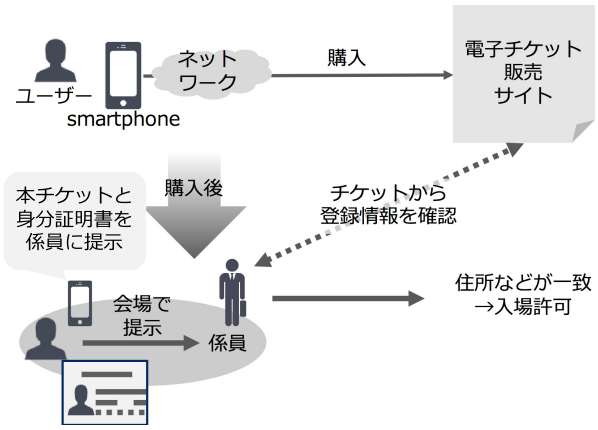


図 1　本稿におけるモデルケース
Fig. 1　The model case of this paper.

2.1 モデルケース

図 1 に本稿が解決すべきモデルケースとして以下の流れに沿って実施する電子チケットサービスを示す．

(1) ユーザは自身のスマートフォンを使ってオンライン上の電子チケット販売サイトへアクセスする．購入に必要な住所などの項目を Web フォームに記入し，あるイベントのチケット購入が完了する．
(2) あるイベントの当日，ユーザは電子チケットを提示するためにスマートフォンを持参して会場を訪れる．チケットの注意書きには，会場の入口にチケットの不正利用防止を目的とした係員が立っており，彼らにチケットと身分証明書を提示することが書かれている．
(3) チケットの提示を受けた係員は購入時にユーザが入力した住所情報などを参照したうえ，それらが身分証明書のものと一致するかどうかを逐一確認し，一致する場合にのみ会場への入場を許可する．

2.2 用語

モデルケースを分析し，本稿が扱う問題に関する用語を以下に列挙する．

SP　サービス提供者，Service Provider．ユーザが信頼する第三者機関．チケット販売サイトを提供し，販売したチケットへの不正利用を防止する．
登録者　オンラインでチケットを購入するユーザ．
利用者　会場でチケットを提示して入場するユーザ．登録者と同一であってもよい．
攻撃　利用者が別の登録者になりすましてチケットを利用する行為．
本人確認　3 章で後述する，SP が登録者と利用者の一致を確認する方法．
知識認証を用いた確認　ID とパスワードの組など登録者の知識を入力し認証した結果を用いる本人確認の方法．
本人性の検証を用いた確認　指紋情報など，あらかじめ登録者の本人らしさを確認できる情報を預っておき，利

用者が提示した情報からその検証を行う本人確認の方法．

デバイス　各ユーザが所持している端末のうち，チケットを提示するために会場へ持参するもの．

2.3　前提条件

問題を整理する前提として，下記の条件を示す．

(1) SP は利用者や登録者と結託しない．

(2) SP の web サーバには脆弱性が含まれず，不正アクセスなどによる攻撃は成立しない．

(3) デバイスは下記のセキュリティ機能を有している．

- 適切な管理のもと，PIN やパスコードなどによって端末の利用が制限されており，他人はこれらを知る以外に操作できない．
- 端末を紛失時，所有者自身の求めに応じてその位置情報などを知ることができる．

以降の章では，モデルケースをもとに問題を明らかにし，それらを解決する本人確認手法について提案する．

3.　問題

モデルケースで示したように，ネットワークを通じたオンラインでユーザが自身の情報を登録し，その後実世界でその情報をもとに提供を受けるサービスでは，SP が実世界で利用者と登録者が同一であるか確認が必要な場合がある．そこで，SP は本人確認を行うことで，この同一性を確認し，目前の利用者にサービスを提供してよいか判断する．ここでは，本人確認の実施形態を整理し，従来採用されている手法とその課題を整理する．

3.1　本人確認

本人確認とは，SP がサービスの利用を求められたとき，目前に存在する利用者が事前に登録した本人であることを確認するための処理であり，様々な場面でサービスの提供を行う対象が本人であることを確認するために実施されている．登録者はたとえば住所や顔写真などの自身の属性情報をあらかじめ SP に登録しておく．SP はサービスの提供を求める利用者が提示する情報を用いて，利用者とその属性が結び付くことを確認したうえで登録されている属性情報と一致するかどうかを確認する．たとえば，氏名や住所という属性情報は，運転免許証などの公的機関が発行した書類を所持していることと，そこに表記された属性情報が同一であることにより確認することができる．SP は，利用者と登録者が同一であることが確認できた場合にサービスを提供し，確認できなかった場合には別の方法を使って確認を行うか，利用を拒否することができる．

3.2　従来手法

本人確認は大きく**知識認証を用いた確認**と**本人性の検証を用いた確認**の 2 つの手法が用いられている．

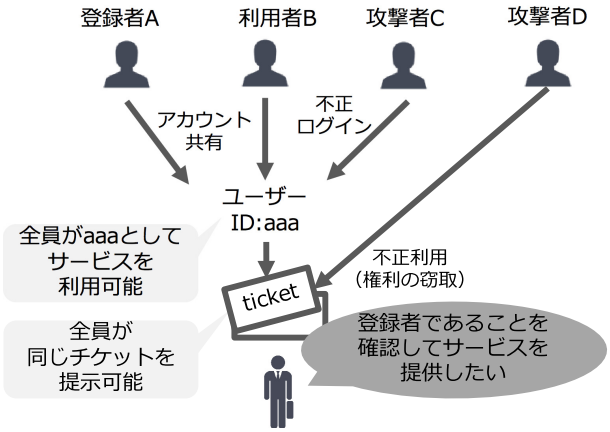


図 2　結託や窃取によるなりすましの例

Fig. 2　Example of spoofing by collusion or stealing.

知識認証を用いた確認は，デジタルチケット販売サービスのパスマーケット [1] や銀行のオンラインバンキングなどが採用している．この方法は，オンライン上で認証を行うため，登録者と利用者が同一の知識を記憶していることを確認可能である．しかし，ユーザが知識を窃取されることで攻撃者によるなりすましを許してしまう [3]．また，一般的な利用形態とは異なるものの，正規の利用者が金銭の授受などを通じて結託することによって，積極的にアカウント共有を行う利用者は登録者と同一であることを確認することが難しい．結託の例として，チケット販売における自身が購入したチケットを他人に売りつける転売行為や，金融サービスにおける口座の売買などの不正利用が，それぞれ社会的な問題として注目されている．モデルケースにおいて，住所などの情報確認の代わりに，知識認証を用いた確認をする場合に，結託や窃取によるなりすましの例を**図 2** に示す．図 2 に示すように，アカウントの共有（結託）(B)，不正ログイン（情報の窃取）(C) やチケットの不正利用（権利の窃取）(D) によるなりすましが行われたとき，SP は利用者と登録者 (A) の一致を確認したうえでサービス提供を行うことができない．

本人性の検証を用いた確認は，オンラインでの銀行口座開設サービスを提供しているみずほ銀行 [4] やデジタルチケット管理を行うテイパーズ [2] などが採用している．運転免許証などの公的機関が発行した書類を登録し，記載された住所へ必要書類を送達可能なことや，顔や指紋などの身体的特徴情報を登録し，イベント会場でそれらの特徴を持つ人であることを確認しようとする試みである．これらの手法は，知識認証を用いた確認と比べて，情報を窃取することが難しく，また窃取できたとしても攻撃者による再利用が ID やパスワードなどに比べて難しいため，より強固に登録者と利用者の一致を確認することができる．一方，登録を行うためにはユーザの心理的な負荷が大きい．つまり，ユーザは SP を信頼できるか否かにかかわらず，

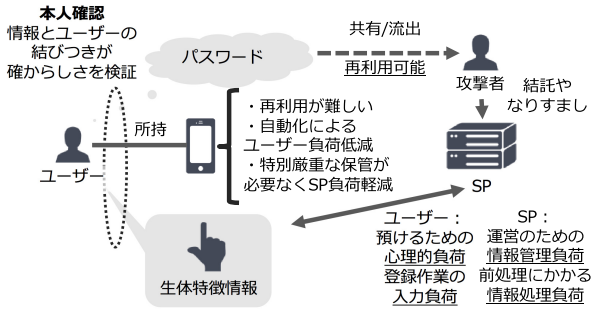


図 3　要件分析
Fig. 3　Requirements analysis.

本人が変更することの難しい情報を渡さなければ，サービス提供を受けることができない．また，登録する情報が正しく検証されなければサービス提供を受けられない可能性が高いため，たとえば，顔写真を登録する際には，光度や表情・向きなど複数の条件を満たす写真を撮影する必要があり，容易に登録することができない．一方，SP はユーザから預かる情報を保護するために知識認証を用いた確認で保存する情報に比べて厳重な保管・管理が求められ，負担が大きい．さらに，登録者が登録した情報が照合に使えるものか事前に判定しておかなければ正しい利用者が本人確認を受けても失敗してしまうことがある [5] ため，それらのチェックを行う手間もかかる．

3.3　要件

問題を分析した結果，以下の 3 つの要件を抽出した．

(1) ユーザ同士の結託やデバイス・情報の窃取による攻撃が試みられたとしてもそれを防ぐことが可能である．
(2) ユーザの登録や利用の負担を軽減する．
(3) SP が情報を管理/運営する負担を軽減する．

各要件を**図 3** のように分析した．つまり，従来の知識認証を用いた確認では，確認に用いる情報が他人によって再利用可能であるため，結託や窃取によるなりすましを許す．一方，従来の本人性の検証を用いた確認では，知識認証を用いた確認に比べてなりすましを抑制できる可能性が高い反面，情報を登録するユーザや運営する SP の負担が大きい．本稿では，これら 3 つの要件を並立して解決する本人確認手法を提案する．

4.　提案手法

前述の要件を満たすために，SP が所持物とユーザの関連付けを検証する手法を提案する．特に，ユーザが携帯して頻繁に利用するスマートフォンなどのデバイスに注目し，再利用が難しく，ユーザの明示的な入力が不要で，SP の保管・管理にかかる負担を軽減する本人確認手法を提案する．

4.1　設計

本人確認の手法として，デバイスの情報を含むユーザ情報の登録（登録フェーズ）とその情報をもとに検証を行う（検証フェーズ）の 2 つを提案し，設計する．**図 4** に提案手法における本人確認の設計を示す．ユーザは 2 つのフェーズにおいて，スマートフォンなど自身のデバイスを用いて SP の Web サイトで操作を行う．検証フェーズでは，リーダと呼ぶ情報読取装置を通じて利用者自身が登録者であることを示してサービスの利用を申請する．

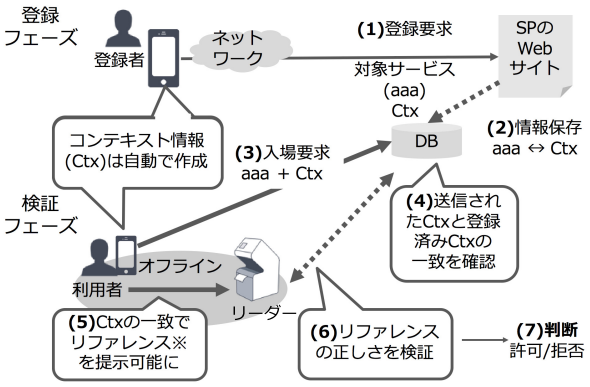


図 4　提案手法の設計
Fig. 4　Design of proposed system.

登録フェーズ　登録フェーズは，登録者が SP に対して対象のサービスとともに自身のデバイスに関する情報である，コンテキスト情報（Ctx）を登録する過程である．図 4 において **(1)** 登録者は SP の Web サイトで提供を受けるサービスを選択する．このとき，非同期処理を用いることで，登録者の利用デバイスに依存する Ctx を自動的に生成してサービス選択と同時に送信する．SP は指定されたサービスを登録者に提供してもよいか（たとえば決済の完了など）を確認し，**(2)** サービス aaa を登録者に提供可能であることを DB に保存し，同時に登録者の ID と対応する形式で Ctx を保存し，登録フェーズを終了する．

検証フェーズ　検証フェーズは，登録フェーズで登録された登録者の Ctx と利用者の Ctx との比較を通じて，登録者と利用者が一致していることを確認する過程である．図 4 において，利用者はサービス aaa が提供される日に会場に赴き，**(3)** サービス aaa に対する利用要求を行う．このとき，(1) と同様に Ctx' を自動で生成し，送信する．サービス aaa とともに Ctx' を受け取った SP は **(4)** Ctx' と aaa に登録者の ID で登録されている Ctx を比較する．その結果，一致するときには，提供サービスを指し示す（リファレンス）情報を利用者端末に送信する．利用者の端末にはたとえば二次元バーコードなどの形式に変換された **(5)** リファレンス情報が表示され，それをリーダに対して提示する旨が利用者に通知される．リーダは **(6)** リファレンス情報を読み取り，それを SP に問い合わせることによって正しく発行された情報であることを確認する．これによって **(7)** リファレンス情報を読み取ることができた端末を持つ利用者と登録者が同一であることが確認できたと

判断し，サービスを提供する．また，リーダが正しくないリファレンス情報を読み取った場合や，リファレンス情報を提示できない利用者へのサービス提供を拒否する．SP がサービス提供を行うか判断が終了したら検証フェーズを終了する．

4.2　識別と伝播

4.1 節では，非同期な自動処理によってユーザの登録にかかる入力の負担（要件 (2)）が軽減できることを述べた．ここでは，その他の要件 (1) と (3) を満たすため，オンライン上で行う Ctx による識別と，その結果をオフラインでのリーダとのインタラクションで伝播する手法を設計する．すなわち，Ctx は，識別に用いることで結託や窃取によるなりすましに対して頑健であり，SP により容易に管理可能な情報であるように設計する．また，識別結果はリファレンス情報としてオフラインに存在するリーダへ伝播するため，それを結託や窃取によって再利用されることによるなりすましを防ぐための設計を行う．

4.2.1　コンテキスト情報（Ctx）による識別

Ctx は，ユーザが頻繁に利用し携帯するスマートフォンなどの情報（デバイス情報）とその利用環境に関する情報（環境情報）の 2 つで構成する．デバイス情報は製品に由来する固有情報であり，環境情報はたとえば利用している言語などユーザの利用形態に依存する内部的な設定情報である．デバイス情報での識別だけでは，同一の端末を使用しているユーザを識別することはできず，環境情報での識別だけでは，同じ地域のユーザで大きな違いは出ないことが考えられるため，これらを合わせた Ctx を識別に用いることでユーザ 1 人に対して 1 つの識別子を付与する．

デバイス情報と環境情報を合わせてユーザの識別に用いることで，なりすましを行おうとする攻撃者は自身の手元でその情報を再現する環境を用意しなくては攻撃が成功しない（6.1 節で後述）．しかし，これらの情報が一致する確率は 0 ではなく，同じサービスを利用しようとする利用者が同じ Ctx を登録した場合，本人確認を誤ってしまう可能性がある．そこで，Ctx に登録フェーズで行う ID を確認した後に発行する ID 確認情報も加えることにより，これらの利用者も識別することを提案する．

デバイス情報と環境情報，ID 確認情報によって作成した Ctx は，多くの場合ユーザには変更が難しい，あるいは変更すると不便が生じるため，たとえばこれらの情報が流出した場合個人の特定やトラッキングなどの恐れがある．そのため，Ctx をそのまま SP へ送出することは，ユーザが SP を信頼できない場合に心理的な負荷が生じ，SP も情報を預かるためにより厳重な保管が求められるため負担となる．そこで，Ctx は送出する前にかならず一方向関数などを用いることで不可逆に変換して送信する．

4.2.2　識別結果の伝播

4.1 節で述べたように，SP は Ctx が一致した証拠として，サービスのリファレンス情報を利用者が持つデバイスへ送信し，リーダへ提示する．リファレンス情報を用いることで，オフラインでの利用者の Ctx の窃取や共有を防ぎ，他の目的に転用されることから保護する．

一方，リファレンス情報は，利用者あるいは登録者を示す情報が含まれていないため，Ctx が一致しない他人でも再利用可能である．たとえばリファレンス情報の形態として，二次元バーコードを用いた場合，画面をキャプチャすることで本人確認を受けていない利用者でもリーダに提示することができる．そこで，伝播の方法として，動的リファレンス情報生成と伝播ポリシの設定を提案する．

動的リファレンス情報生成は，リファレンス情報の中に時間情報や発行回数などを埋め込むことで，Ctx の検証ごとに異なるリファレンス情報を生成する．これにより，リファレンス情報の再利用の防止と検知を行う．

伝播ポリシとは，SP が検証フェーズで Ctx やリファレンス情報を検証する際の規則である．SP は利用者から送付された Ctx やリファレンス情報を検証する際，ポリシを確認し，その結果を利用者の端末やリーダに通知する．ポリシによって検証が失敗した場合，利用者はリファレンス情報の再取得を行わなくてはならない．たとえば，リファレンス情報に十分に短い有効期限を設けるポリシを設定することを考える．利用者が期限切れのリファレンス情報をリーダに提示したとき，リーダは利用者に対してその場で再取得を要求する．再取得には，再度 Ctx の照合を受ける必要があるため，利用者がその場で同じ Ctx が生成できない場合，なりすましとしてサービス提供を拒否することが可能である．このようにポリシをあらかじめ SP 側で設定しておくことによって，リーダは具体的なポリシや利用者の Ctx を把握しなくてもサービスの利用を許可あるいは拒否することができる．伝播ポリシを適用することで，リファレンス情報が再利用可能な状況を利用者に合わせて制限し，なりすましを防ぐ．

5.　実装

提案手法の実現性を示すため，ユーザがデバイス上のブラウザを用いて，SP の Web ページで提供を希望するサービスの登録を行い，オフラインでも同様の環境を使用することで本人確認を受けるシステムを構築した．ここでは，登録と検証の 2 つのフェーズでの処理を詳細に述べる．次に，SP のサーバが必要なデータ内容を明らかにし，最後に，コンテキスト情報の実装である Fingerprint の生成と照合方法について実装の詳細を述べ，適用可能なユースケースとそれを実現した実装の動作を述べる．

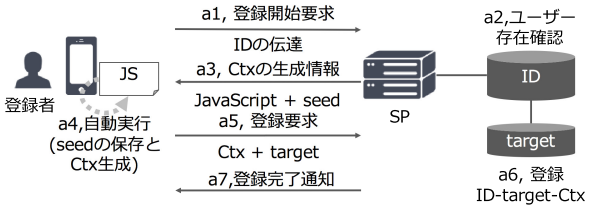


図 5　登録フェーズの実装

Fig. 5　Implementation of registration phase.

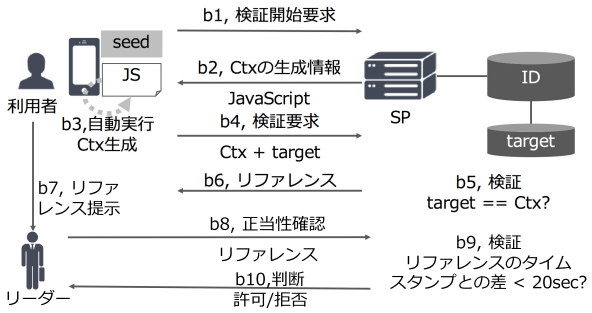


図 6　検証フェーズの実装

Fig. 6　Implementation of verification phase.

5.1　登録フェーズ

図 5 に登録フェーズ（手順 a1–a7）の実装を示す．

a1. 登録開始要求　登録者は提供を希望するサービスを選択するため，SP の Web サイトへアクセスする．SP は，登録者を個別に識別するための ID を発行する．すでに SP の ID がある場合，ログインを通じて ID を申告してもよい．

a2. ユーザ存在確認　**a1** を受けた SP は，ID の存在を確認する．新たに発行した ID であればこのとき登録する．

a3. コンテキスト情報を生成する情報を送信　コンテキスト情報（Ctx）を作成するため，ID 確認情報を作成し登録者の端末に送信する．ID 確認情報は，容易に推測させないために ID ごとにランダムな文字列 nonce を使って生成する文字列を用いる．本実装では，異なる ID 間で衝突しないように，nonce と ID を接続した文字列を SHA-256 でハッシュ化した情報を seed と呼び，ID 確認情報として用いる．nonce は初回だけ生成し，ID とともに保存しておく．Ctx を生成するための JavaScript を埋め込んだ，登録者がオフラインで提供を希望するサービス（target）を選択する画面（HTML）を表示するとともに登録者に seed を送信する．

a4. Ctx の自動生成　ブラウザのバックグラウンドでは，**a3** の HTML 内に埋め込まれた JavaScript が非同期で Fingerprint を算出する．Fingerprint は，デバイス情報と環境情報をあわせた文字列であり，後述する手順（5.4 節）に沿って算出する．本実装では，ID 確認情報（seed）を用いて作る Fingerprint を Ctx として算出する．また，ブラウザの cookie に seed を記憶させる．

a5. 登録要求　JavaScript により生成した Ctx を HTML 内のフォーム（hidden 属性）に埋め込む．この動作が完了するまではフォームの送信ボタンを無効化する．登録者が送信ボタンを押したとき，Ctx と target を SP へ送信する．

a6. 登録　SP は受け取った target と Ctx の組合せを ID に対して紐付けることで登録する．このとき，SP は必要に応じて登録するためのポリシを参照する．

a7. 登録完了通知　両方の情報の保存が成功したら，登録要求を受け付けた旨を登録者に通知する．

5.2　検証フェーズ

図 6 に検証フェーズ（手順 b1–b10）の実装を示す．

b1. 検証開始要求　利用者は提供を希望するサービスの本人確認のため，SP の Web サイトにアクセスする．

b2. コンテキスト情報を生成する情報を送信　提供を求めるサービス（target）を選択する画面（HTML）に Ctx を生成するための JavaScript を埋め込んで送信する．このとき，cookie を読み取り，seed が保存されていなかったら登録者の ID を利用者から申告してもらい，対応する seed を保存してある nonce から再計算し送信する．

b3. Ctx の自動生成　ブラウザのバックグラウンドでは，**b2** で埋め込まれた JavaScript が非同期で利用者のブラウザに保存されている cookie から seed を復元するか，あるいは SP から受け取った seed を使用し，利用者の Ctx’ を算出する．

b4. 検証要求　JavaScript は Ctx’ を作成したら，HTML 内のフォーム（hidden 属性）に埋め込む．この動作が完了するまではフォームの送信ボタンを無効化する．利用者が送信ボタンを押したとき，Ctx’ と target を SP へ送信する．

b5. Ctx の検証　target に対し，登録されている Ctx を DB から参照し，Ctx と Ctx’ の文字列の照合を行う．

b6. リファレンス情報の送信　照合が成功した場合，リファレンス情報を送信する．リファレンス情報は，target と作成時間のタイムスタンプ（Unix time）を連結した文字列に AES を用いて暗号化した文字列である．

b7. リファレンス情報の提示　利用者はリファレンス情報が表示されたらリーダに提示する．このとき，リーダが容易に読み取るためにリファレンス情報は HTML 上の画像ファイルとして，二次元バーコードに変換して利用者の端末に表示する．

b8. リファレンス情報の問合せ　リーダはリファレンス情報を読み取り，SP に問い合わせることで，発行されたリファレンス情報かどうかを検証する．

b9. リファレンス情報正当性検証　SP は **b6** で用いたものと同じ AES の鍵を使いリファレンス情報を復号する．復号した文字列が伝播ポリシに合致するか確認する．本実装における伝播ポリシは検証からリーダに提示するまでのタイムラグをできるだけ小さくすることで利用者と端末の所

表 1　Target テーブル
Table 1　Target table.

ターゲット識別子	タイトル	ユーザ識別子	Ctx	登録時間
trgt0001	TokyoMeeting	wogami	34c6cb8b...	1464706800

表 2　ID テーブル
Table 2　ID table.

ユーザ識別子	nonce
wogami	tawhrqn...

有者を関連付けるために，タイムスタンプが 20 秒以内のものであること，とした．

b10. 判断　**b9** の結果，正しいリファレンスを提示できた利用者にはサービスを提供（入場を許可）し，そうでない場合，利用者に再度リファレンスを提示させる．このサービスの利用可否判断には SP が設定するポリシが適用される．

5.3　データ管理

本実装では提供サービスに関する情報とユーザを関連付けて管理する Target と，ユーザの識別情報を管理する ID の 2 つのデータテーブルを実装した．ここでは，各テーブル内容の例を示し，情報の管理について述べる．

Target　Target テーブルの例を**表 1** に示す．Target テーブルは，サービスの内容とそれを所持する登録者の Ctx を管理する．提供サービスはターゲット識別子によって一意に特定できる．ターゲット識別子を指定すると，提供するサービスの内容（たとえばチケット販売では，開催する会場や開場時間など）が参照できる．これらはサービスを受ける際にユーザに提供する情報だが，本人確認には影響を与えないため，本実装では，サービスを代表する文字列として “タイトル” を実装した．Target テーブルは手順 **a6** で登録される登録者名（“ユーザ識別子”）とその Ctx，登録時間を保持し，検証フェーズで参照する．

ID　ID テーブルの例を**表 2** に示す．ID テーブルは，SP を利用した登録者の情報を管理する．登録者識別子と発行した nonce の情報を保存する．ID テーブルは，登録者がすでに登録されていることを確認し，登録者ごとに発行した nonce を特定するために用いる．

5.4　コンテキスト情報の生成と検証

コンテキスト情報（Ctx）は，デバイス情報と環境情報，ID 確認情報の組合せの情報であることはすでに述べた（4.2.1 項）．本実装では，セキュリティ向上を目的にユーザの端末情報を取得する技術として Fingerprint [6] に注目し，Web サイト上でユーザの追跡可能な特徴を活かし，各ユーザにユニークな情報を生成し，これを用いて Ctx を生成する．特に，デバイス情報として Canvas Fingerprint を，環境情報として User Agent（UA）をはじめとする複数の情報を用いた．ところで，デバイスに依存するユニークな識別子として，MAC アドレスが存在する．しかし，MAC アドレスはデバイスに対する固有情報であり，プライバシ上の懸念から多くのプラットフォーム上で取得が禁止されており，また，機種変更などによって変更しない限りユーザが自覚的に変更できず，本稿の要件 (2) にも反するため実装に不適であると判断した．本実装で用いた Fingerprint の手法は主要なブラウザ上で取得できることから Browser Fingerprint とも呼ばれる．各情報の生成方法とそれを用いた検証について述べる．

Canvas Fingerprint（CF）は，Mowery らが提案 [7] した．CF は，ウェブブラウザに搭載された WebGL を使って HTML5 の描画機能である canvas を使うと，同じ描画要素に対してハードウェア（グラフィックボード）やソフトウェア（OS やブラウザ）の違いが差分として計算できることを使ってユーザ環境を特定する手法である．JavaScript の API を使って実装することができ，画像データではなく，そのハッシュ値として扱うため，文字列の一致を確認することで特定の端末を追跡できるためデバイス情報として用いた．

環境情報には，UA などのユーザの利用環境を示す情報を用いた．本実装では，JavaScript の Navigator オブジェクトなどを用いて，以下の情報を取得した．

- 内容（UA / 言語 / 色の深度 / 解像度）
- 有無（セッションストレージ / ローカルストレージ / indexed DB / doNotTrack）
- 種類（open DB / CPU / プラットフォーム / インストールしているプラグイン）

このようにして算出したデバイス情報と環境情報だけを使ってユーザを完全に識別することは難しい．たとえば，Bursztein らによる Picasso [8] では，チャレンジに対してフォントテキストを含めて CF 算出を行うことで，スマートフォンを含めた OS とブラウザの組合せの識別に 100%成功することが報告されている．一方，同じ OS やブラウザであっても，バージョン間の識別に関しては限界があり，最も悪い識別率で 57.4%とされている．これはデバイス情報を主に使った識別であり，他の情報を使って識別可能性が上がることを示唆している．他方，高橋らによるスマートフォンにおける研究 [9] によると，UA と画面解像度のみを使った実験では，iOS 端末では約 27%，Android 端末では約 77%が識別可能である．これは環境情報単体でも少なくともこの数値以上に識別が可能であることを示している．

これらの識別には上記に提示した限界があることを理解したうえで，それらを組み合わせて本人確認を行う．つまり，まったく同じデバイスをデフォルト環境で使う場合でも，SP が登録者と利用者が同一であることを識別する必要がある．そこで，ID 確認情報を Ctx に追加する．

ID 確認情報は，登録フェーズでユーザ識別子を確認し

た後に，以下の手順 c1–c3 に沿って seed を作成し，登録者のブラウザに保存して用いる．

c1 ユーザ識別子をもとに nonce を取得する．

c2 c1 で nonce が取得できない場合，作成して ID テーブルに保存する．

c3 nonce とユーザ ID を結合し，SHA-256 でハッシュ化した文字列を seed とする．

CF と環境情報，および seed を用いた Ctx の生成方法を述べる．本実装では，環境情報と seed を使って canvas の描画内容を変更し，それをもとに CF を作成することでこれらの要素を含む文字列情報として Ctx を算出した．Ctx の算出は次の手順 d1–d3 に沿う．

d1 JavaScript は API を通じ環境情報と seed を取得する．

d2 d1 で取得した情報はすべて文字列であるため，これをすべて “#” で連結する．

d3 d2 を暗号化ライブラリである CryptoJS を用いて SHA-256 で変換する．

本実装の Browser Fingerprint の作成にあたり，GitHub で公開されている Anonymous browser fingerprint [10] の実装を手順 d1–d3 に合わせ改変して用いた．

5.5　ユースケース

モデルケースでは，知識認証 [1] や本人性の検証 [2] を用いた確認が数多く行われており，事例を通じた比較が行いやすい．従来のチケット販売における本人確認方法では，ユーザ同士が結託を行うことで転売行為が可能になり，チケットの希少性が上がった．そのため，需要のあるユーザがチケットを入手しにくい，あるいは高額で転売者から購入しなければ入手できないという不利益を受けるケースが増えた．SP は事業者としてこうしたユーザへの公平性提供のために本人確認を行うケースが増加している．実装では，ユーザがデバイスを使って SP の Web ページでチケットを購入し（登録フェーズ），会場を訪れた後にシステムを通じた本人確認を受ける（検証フェーズ）というユースケースを想定した．

ユースケースによって，各フェーズ内に適用されるポリシは変更される．たとえば，手順 **a6** では，在庫の有無や支払いの完了など登録するためのポリシを参照する．また，手順 **b10** では，5 回連続でリファレンスを表示できなかった場合，あるいはリファレンスを提示しても正当性の検証を成功できない場合，サービスの提供を拒否することができる，という制御など検証の実施に関するポリシを設定できる．これらは本人確認の結果には影響しないため，本実装では伝播ポリシ以外のポリシを排除して実装した．

5.6　実装環境

実装に用いた環境およびフレームワークを**表 3** に示す．実装した電子チケット販売システムの一部を**図 7** に示す．

表 3　実装環境
Table 3　Implementation environment.

言語	PHP 5.6.27
フレームワーク	CakePHP 2.4.10
DB	MySQL 5.5.50-38.0–log
二次元バーコード生成	PHP QR Code 1.1.4

図 7　本人確認手法の実装画面（一部）
Fig. 7　A screen shot implemented for identity verification.

(1) 登録者はチケットリストから所有者のいない任意のチケットを選択して購入することができる．その際，ID は事前にログインもしくは仮発行することでブラウザに保存された cookie から読み出す．

(2) (1) と同様にして ID が判明しているため，登録者は購入画面から購入ボタンを押すだけで容易に登録フェーズが完了する．

(3) 購入後，利用者は Web サイトを訪れ，(1)，(2) と同様にブラウザの cookie を用いて自身が所有するチケットの内容を確認することができる．

(4) 利用者は利用を要求するチケットの内容を確認した後，「表示」ボタンを押すと，Ctx を使った本人確認が実施され，本人確認結果がリファレンスの有無を通じて通知される．たとえば，図 7 の右下の検証結果画面では，どちらも同じ ID を申告して本人確認が行われたが，別の環境の利用を検知したため，右側の利用者にはリファレンス（2 次元バーコード）を表示していない．

6.　評価

従来手法である知識認証や本人性の検証を用いた確認との比較を通じて提案手法に対する定性評価を行う．従来手法による本人確認を実施しているサービスへの攻撃から，利用者 B が攻撃者となって登録者 A の情報や所有物を窃取するモデルと，登録者 A と利用者 B が信頼関係のもと結託するモデルにより抽象化を行った．ここでは，各攻撃モデルの具体的な攻撃手法を検討するとともにその耐性について評価を行う．

6.1　窃取モデル

利用者 B が攻撃者となり，別の登録者 A として振る舞うために必要な情報や端末を窃取する攻撃を窃取モデルと呼ぶ．従来の本人確認手法では，ID とパスワードなど登録者 A が記憶している情報や，印鑑や公的書類などの所有物を窃取することによりなりすましが行われている．ここから具体的手法を推測すると，窃取モデルを A と B の間でネットワークを介さず直接 A の所有物を窃取する直接接触攻撃と，ネットワークを介して A の情報を窃取する間接接触攻撃に分類でき，その具体的攻撃手法への耐性を評価する．5 章で述べた実装において，提案手法による本人確認に対して攻撃を成功させるために，B は A のデバイスで生成される Ctx を再現する必要がある．

6.1.1　直接接触攻撃

直接接触は，攻撃者 B が登録者 A とネットワークを介さずに直接接触を図ることで A になりすます攻撃である．本稿の実装による本人確認において，攻撃を行う B は，A のデバイスを窃取することで攻撃を試みる．ここで B が A になりすましてサービス提供を受ける際，以下の両方の条件を満たさなくてはならない．

- A が攻撃に気づかない．
- B が A のデバイスを自由に操作できる．

もし A が攻撃に気づいた場合，たとえば SP への電話連絡などにより，A に対する本人確認の実施を停止することができる．また，B がデバイスを手に入れたとしても，前提条件にある PIN や指紋認証などのロックを解除できなければ，提案手法による本人確認を行うことはできない．

提案手法に対する直接接触攻撃は以下の理由から失敗する．まず，デバイスは A が本人確認を受ける際に必要であり，A は自身が本人確認を受ける際に端末の紛失に気づく．また，本人確認はオフラインのサービス提供を行う場に A が赴いて実施されるため，B は直接接触攻撃を本人確認が実施されている場に地理的に近い場所で窃取を行う必要がある．このため，端末の窃取には成功したとしても，本人確認を A の申請によって停止される可能性が高い．また，デバイスを窃取できた場合でも，端末のロックを解除する情報も窃取しなくてはならず，A がデバイスの紛失に気づく前にそれらの情報も窃取したうえで B が本人確認を受けることは現実的ではない．さらに，提案手法では本人確認時にネットワークを介したアクセスを行わなくてはならず，B はそのアクセスによって窃取した端末のたとえば現在位置を A に知らせてしまう可能性もある．したがって，提案手法に対する直接接触攻撃は失敗する．

6.1.2　間接接触攻撃

間接接触は，攻撃者 B が登録者 A の端末にネットワークを介して不正プログラムなどを送り込み，間接的な手段で接触することで A になりすます攻撃である．提案手法に対して，B は自身の端末上に A の Ctx を再現するために，A の端末情報の窃取を試みる．このとき，B はたとえば見た目が SP とよく似ている Web サイトに誘導するフィッシング攻撃を用いる．Ctx は，SP が配布する JavaScript で cookie 内に保存してある seed を読み出して算出するため，ドメインの異なるフィッシングサイト経由で窃取することは難しい．

一方，別の方法として，たとえば管理者権限を持つマルウェアをフィッシングサイト経由で A のデバイスにインストールすることができる場合，A の同意なくメモリ上に載った Ctx を窃取することは可能であり，B の攻撃が成功しうる．ただし，一部の条件を満たせばこの間接接触攻撃を防ぐことが可能である．条件とは，対象となるサービス 1 つに対してユニークなターゲット識別子が発行されていることである．たとえば，モデルケースにおいては，席ごとにチケットが発行されることが多い．間接接触攻撃が行われたとき，A と B は双方が正しいリファレンスを提示するため，同じサービスの利用要求を行うという衝突が起こる．つまり，SP はこの状況で A にのみ許可を出すことができれば攻撃を防ぐことが可能である．

具体的に解決方法を述べる．リーダは，正しいリファレンスを提示した双方に環境情報を書き換えることを依頼する．具体的には，たとえば言語設定をふだん使っていないものにしてもらう，など Ctx の算出に利用している要素のうち，一般的な設定により変更できる項目を使う．書き換えた後，Ctx に変化が見られないものは，B であることが分かり，拒否することができる．これは，B が攻撃時に A が登録した Ctx は窃取したものの，その他の情報をオフラインの場においてすべて再現することが難しいためである．

6.2　結託モデル

結託とは，本来なりすましの被害者である登録者 A が，金銭のやりとりなど一定の信頼関係を築いた利用者 B とともに攻撃者となり，なりすましを行うモデルである．結託モデルを，親しい関係者同士での永続的な関係性の中で行われる永続結託攻撃と，金銭などのインセンティブにより一時的に結託する一時結託攻撃に分類し，その具体的攻撃手法への耐性について評価する．5 章で述べた実装において，提案手法による本人確認に対して攻撃を成功させるために，A が協力することで，A の Ctx を B に再利用させようと試みる．

6.2.1　永続結託攻撃

永続結託攻撃は，たとえば家族など親しい関係者同士の A と B が積極的に協力することで，A が登録したサービスを B に享受させようとする攻撃である．A と B の間には強い信頼関係が存在するため，A は B のなりすましに最大限協力する．

A は B に自身のデバイスすら共有することが可能であり，間接接触攻撃のように衝突が発生しないため，環境情

報の書き換えによるなりすましの発見を行うことができない．また，貸与によってAが登録したCtxを完全に再現できるため，正しいリファレンスをつねに提示可能であり，この攻撃を提案手法によって検出することはできない．永続結託攻撃は提案手法に対する脅威の1つであり，7章でその防御可能性について考察する．

6.2.2　一時結託攻撃

一時結託攻撃とは，金銭のやりとりなどを通じてAとBが一時的に結託する攻撃である．たとえば，電子チケット販売サービスでは，攻撃者Aが購入したチケットを利用者Bに売りつけることで，AはBから対価を受け取る代わりに攻撃に対して積極的な協力を行う．一方，永続結託攻撃に比べて弱い信頼関係しかないため，AがBに自身のデバイスを貸与するとそれを持ち去られてしまう危険性がある．そのため，一時結託攻撃では貸与が行われないと仮定する．このとき，Aは自身のCtxをBに共有することで攻撃を試みる．

この一時結託攻撃は防ぐことが可能である．なぜならば，Bは本人確認時に以下の行動をリーダの前で完了する必要があるためである．

- Aの Ctxを受け取る．
- Bの Ctxを書き換える．
- リファレンスを有効期限内に提示する．

本稿の実装では，Webサイト内のフォーム情報であるため，これを任意に書き換えることは難しくはないが，リーダと対面した状況において手動でCtxを書き換えることは難しい．しかし，AがCtxに関する情報を積極的に共有するため，たとえばBが自動化したソフトウェアで書き換える場合にはその検出が難しくなる．後述の7.2節ではこのような状況にも耐性のある情報を用いる手法について考察する．

6.3　従来の認証方法との比較

知識認証を用いた確認　この本人確認手法は，なりすましに対して脆弱である．まず，窃取モデルでは，盗み見やマルウェアなどにより認証に必要な情報が窃取可能である．同じサービスに対する衝突の解消はできず，間接接触攻撃による攻撃を防ぐことができない点も本提案手法に比べて脆弱である．また，結託モデルに対しても，必要な情報を共有され，再利用可能なため脆弱である．一方，ユーザが登録時にSPに情報を預ける負担や，登録する作業による負担は後述する本人性の検証を用いた確認に比べて比較的軽い．しかし，なりすましを避けるためには文字列長が十分に長く，多くの種類を使った無作為なパスワードを使うなどの方法が推奨されており[3]，負担が大きくなる可能性がある．最後に，SPの登録負荷は従来的な認証基盤を活用できるため，提案手法に比べて容易に実装や運用が可能である．

表4　各手法に対する評価
Table 4　Evaluation on each method.

評価軸	提案手法	Web passwords	Fingerprint
Memorywise-Effortless	○	×	○
Scalable-for-Users	○	×	○
Nothing-to-Carry	×	○	○
Efficient-to-Use	○	○	×
Infrequent-Errors	○	○	×
Resilient-to-Physical-Observation	○	×	○
Resilient-to-Targeted-Impersonation	△	×	○
Resilient-to-Internal-Observation	△	×	○
Resilient-to-Phishing	△	×	○
Resilient-to-Theft	△	×	×
Requiring-Explicit-Consent	△	○	○
Unlinkable	○	○	×
SPの情報管理負荷※	△	○	×

本人性の検証を用いた確認　この手法は利用者が登録している属性をその場で再現できることを確認する手法であるため，攻撃モデルに対して，その検証方法に依存した頑健性を持つ．一方，すでに3章で述べたとおり，ユーザが情報を登録する負担やSPがそれらの情報を適切に管理する負荷は大きい．

6.4　総合評価

Bonneauらによるパスワードとその他のWeb認証方法を比較評価した研究[11]を参考に，従来手法と提案手法における評価を行った．本稿では，Bonneauらの評価した手法の中から，従来手法の知識認証を用いた確認としてWeb passwordsの項目を，本人性の検証を用いた確認の1つとしてBiometricカテゴリのFingerprintの項目を参照し，提案手法と比較する．本人確認と関連する項目として，Bonneauらが評価した25の軸のうち，4章で述べた要件にあてはまる軸を選択し，**表4**に示す．評価は，各軸の性質を満たせば○を，そうでなければ×を，何らかの条件下で満たすことができれば△として，表4に整理した．

提案手法は，デバイスの所持以外の要素ではユーザの負担を軽減しており，なりすましに対しても従来手法に比べて衝突などの条件が合わされば防御が可能な点が増えており，セキュリティ面では頑健である．Requiring-Explicit-ConsentとUnlinkableはプライバシに関わる項目であり，後述する7.3節で述べるとおり，提案手法はUnlinkableな

手法である．一方，ユーザに ID を申告してもらうため，Requiring-Explicit-Consent は守られているものの，Ctx の計算は非同期で行われるため，それらの情報を用いることをユーザに同意をとってから行う必要があることから，△とした．

SP が情報を管理・運用する負荷に関する要件を評価するため，元の論文のスコープに入っていない [11] 項目として，表 4 の「SP の情報管理負荷」について別途評価した．この評価は，SP が指紋情報という本人を表す情報を預かるうえで厳重な保管を求められるため，Fingerprint は Web passwords に比べると低い．提案手法では，Ctx の性質を検討し，たとえばデバイス情報が機種変更によって変更でき，環境情報を自身の設定によって変更することが可能であるため，指紋情報などに比べて SP が情報を管理する負担も減る．一方，提案手法と Web passwords を比べると，双方とも文字列の一致を照合するという点では類似しているが，Ctx はユーザの操作環境などから生じる情報であるため，パスワードに比べるとやや扱いに慎重になるべきである．したがって SP の情報管理負荷については従来手法の中間（△）という評価にした．

7. 考察

デバイスの Ctx を用いて本人確認を行う手法を提案した．ここでは，6 章の評価のうち，提案手法による本人確認が難しいと評価した永続結託攻撃に対する防御と，Ctx の継続性を向上させる方法について考察を行う．また，Fingerprint を用いることで一般的に懸念されるプライバシの侵害可能性についても考察を行う．

7.1 永続結託攻撃に対する防御

提案手法は，登録者と利用者が同じ端末を利用してサービスを利用していることを担保にその本人性を検証している．そのため，従来の本人性の検証を用いた確認に比べて，強い信頼関係をもとにした永続結託攻撃に対する耐性が得られない．この原因は，Ctx が端末や操作環境に依存するものの，提案手法では利用者が端末を頻繁に利用していることを間接的に検証する点にある．本稿の実装のみでは，この点を補うことは難しいが，端末の所持を強力に検証できる実装方法について 2 点考察する．

端末を利用者に強く関連づける方法の 1 つとして，生体認証など種々の認証結果をオンライン上で活用する技術，たとえば標準技術として提案されている Fast IDentity Online（FIDO [*1]）などを使うことが考えられる．FIDO 認証では，ユーザの生体情報など認証に用いる情報をサーバに送信しない代わりに，認証結果として端末内で Trusted Execution Environment（TEE）などの安全な領域に保存された秘密鍵を使った署名付きのメッセージをやりとりする．このメッセージを Ctx として扱うことで本提案手法による本人確認の方法として実装することが可能である．FIDO などの非対称鍵をベースにした認証方法を用いた場合，登録した本人の生体情報をデバイスの中で照合するため，関連づけられた秘密鍵による署名を検証することで所持情報も確認することが可能である．また，ユーザの生体情報をサーバに送信しないため，ユーザや SP の負担も従来の本人性の検証を用いる確認に比べて小さい．一方，この方法を用いた場合生体情報の登録や認証ごとにかかるユーザの負担が増える．しかし，知識認証を用いた確認に比べてその負担は軽微であることが多い．たとえば，多くのスマートフォンに搭載されている指紋認証機能を使う場合，ID とパスワードを入力する時間に比べてはるかに短い時間で認識ができる．

利用者と端末の所持を強く確認するもう 1 つの方法として，Ctx の算出にユーザの活動や存在を要素として取り入れたうえでハッシュ化して送信することが考えられる．たとえば，周囲の音，光，利用者の動きによって端末の物理的位置を判別しようとする研究も報告されている [12]．デバイスを用いて，登録時の自然音やふだんの歩幅や加速度などのユーザと端末が連動して計測される活動に関する情報を使えば，永続結託を行う関係であっても容易に再現することは難しい．たとえば，親子関係のユーザ同士が永続結託を行う場合，親子が日中の時間帯で継続的に同じ位置や行動をとることは難しい場合が多い．一方，これらの情報を取得するためには，利用者のプライバシに触れる可能性もあり，取得情報への同意や理解を得るとともに，操作の煩雑性をなくすなどユーザの負担をできるだけ少なくする工夫が必要である．

7.2 コンテキスト情報の継続性

一時結託攻撃において自動化されたソフトウェアが登録者より積極的に提供を受けて Ctx を書き換えることで攻撃が成功することはすでに述べた．この攻撃は，そのときどきにおいて Ctx を算出する提案手法だけでは防ぐことができない．そこで，Ctx の生成に継続的な情報を含めることで，一時結託や間接接触攻撃で利用サービスの衝突が発生しない場合でも，Ctx 自体の再現が難しくなることで攻撃を防ぐことが可能になる．

また，OS やブラウザなどのバージョンアップやプラグインの有効性など 1–2 週間の短い期間であっても環境情報が変わる可能性がある．つまり，登録者本人が利用している同一の端末であっても，時間の経過によって Ctx が変化する可能性があり，その継続性を上げることでさらに本人確認の精度を上げることが可能になる．

Ctx の継続性を高める 1 つの方法として，Ctx のほかに追加の情報も受け入れることが考えられる．登録フェーズ

[*1] 入手先 ⟨https://fidoalliance.org⟩

で他の端末を同時に登録しておき，利用者が本人確認を受ける際，他の端末でも提案手法による本人確認を受けることで間接接触攻撃を防止できる．これは電子チケット販売サービスのように複数人で同時にサービスの利用を受けるときに都合がよい．たとえば，ともに同じ内容のサービスを受ける同行者の端末を登録しておくことで，ユーザは同時に本人確認を受ける場に来るという自然な動作によって，窃取モデルによるなりすましを防ぐことができるという利点を得ることができる．

Ctx の継続性を高めるもう 1 つの方法として，Ctx の各要素を継続的に更新していくことも可能である．たとえば，Ctx の算出に使う要素を 1 日ごとに観測すると，すべての要素が変化することは非常に稀である．そこで，たとえば Ctx の要素情報を一方向関数などで不可逆変換した情報を持っておき，一部の変化であれば新しい情報に更新するなど，短いスパンでの小さな変化を受容することによって継続的に登録者の Ctx を把握することが可能である．これらの変化量の受け入れに関する研究も存在する [13]．

7.3　プライバシの侵害可能性

Ctx を取得する実装として Fingerprint を用いたことは述べたが，ここでは，トラッキングによるプライバシの侵害の可能性を否定する．これは，もし，本提案手法によるトラッキングが可能であれば，日常的に使用しているブラウザ上でのユーザ行動が必要以上に SP に明らかにされることで，4 章で述べたユーザの心理的負荷が大きくなるためである．

特に，Browser Fingerprint を指して，情報の取得がプライバシの脅威となりうることは指摘されている [14]．これは，Fingerprint が多くのブラウザでユーザの利便性を高める目的で用いられている JavaScript を用いて実装されており，回避することが難しい [15] ためである．特に，一般的には cookie では行えないドメインを越えた追跡（ユニバーサルトラッキングと呼ぶ）が可能であり，利用履歴の名寄せなどを行うことができる点がプライバシの侵害を行っているといえる．ユニバーサルトラッキングは，Canvas Fingerprint による同一の描画内容を共有する Web サイト間で発生しうる．すなわち，同じ環境を使っているとき，ユーザの環境情報は同一であるため，Fingerprint の値はデバイス情報（Canvas Fingerprint）に依存する．もし複数の SP がその描画内容をすべてのユーザで同一にすると，デバイス情報は複数の SP のサイトで同一にすることができるためユニバーサルトラッキングが発生する．

しかし，本稿の実装では，Fingerprint 単体ではなく ID 確認情報をその要素に持つ Ctx を識別に用いており，ユニバーサルトラッキングは発生しない．認証を行う代わりに，ID を確認した後に発行する cookie の中に含められた seed が Fingerprint に含まれるため，cookie を通じて seed が他の SP へ伝えられることがないことはブラウザの機能が保証している．また，SP 間で描画内容を共通にしようとして 5.3 節で述べた ID テーブルを共有したとしても，両方のサイトで共通の ID が特定できない限り，同じ seed をユーザに配布することはできない．このように，ID 確認情報は本人確認を確実に行うためだけでなく，副次的にユーザのプライバシ保護にも役立つ．

8.　関連研究

関連研究として，個人の所有物やユーザのコンテキストを用いることで，従来の認証方法を強化し，本人確認を試みる研究について述べる．

Hayashi ら [16] は紙に印刷したり，端末に保存したりすることの可能な WebTicket を使い，パスワードと併用することでユーザに使いやすい認証方法を提案した．紙に印刷するなどしたトークンをカメラに提示することでユーザのログイン負担を軽減するアプローチは，認証を行う代わりにコンテキスト情報の検証結果をリファレンス情報としてリーダに伝播する提案手法と類似している．一方，提示されたトークンは再利用可能で，結託や窃取によるなりすましが可能なため，リファレンス情報を変化させてリーダの目前で検証を行うことによりなりすましを防ぐ提案手法とは異なる．

また，Tamrakar ら [17] は，Near Field Communication（NFC）を使って，個人端末の所持を確認することで少額の支払いを行う公的交通機関のチケットを提案した．この手法は，デバイス内のハードウェア的に隔離された領域である TEE 内に格納した秘密鍵を使って，リーダが生成したランダムな文字列（チャレンジ）に署名を行い，それをリーダで検証することで登録者のデバイスであることを確認することができる．認証を行う代わりに，端末に強く紐づけた秘密鍵を用いるため一時結託攻撃や窃取モデルによるなりすましに従来手法より頑健な点が提案手法と類似している．しかし，NFC はつねに通信を行うことが多く，その場合ユーザのインタラクションがないため，間接接触攻撃を行ってチャレンジや署名を攻撃者の端末に転送するなりすましが可能である．提案手法とはコンテキスト情報を毎回確認してリファレンス情報を提示させるというリーダとのインタラクションが必要な点で異なる．

Alaca ら [6] は，パスワード強化のために，Device Fingerprint の要素を分類し，攻撃者モデルを定義してそれに対する耐性を分析した．この研究では，Device Fingerprint として，より多くの要素を使うことでユーザの識別が可能になると結論づけており，ブラウザ以外からも取得可能な種々の要素が提案されている．提案手法では，ブラウザの利用を想定しそこから取得可能な情報を利用したが，たとえばアプリケーションとして実装すれば，それらの要素を採用できるため，なりすましに対してさらに頑健にできる

可能性がある．また，ユーザとの間にインタラクションなく認証が行うことでその負担を軽減する点では提案手法と類似しているが，ユーザから情報を implicit に多種類収集することでユーザを識別するアプローチは，ID 確認情報を使ってプライバシを保護するという点でもユーザの負担軽減を行う本提案手法とは異なる．

Karapanos ら [18] は，2 段階認証にトークンや SMS を用いる代わりにふだん利用しているデバイスの近接を環境音の観測を用いて行う Sound-Proof という手法を提案した．この手法では 2 台のデバイスのマイク経由で周囲の環境を観測することで，同じコンテキストを共有していることを 2 段階認証の要素として用いることで，ユーザ負担の軽減を目的としている．マイクという比較的どのデバイスでも搭載しているセンサを用いる手法であることから，考察でも述べたように継続的に環境音を 2 つのデバイスから取得することでより端末と個人の関連性が強くなるため，永続結託攻撃に対応できる可能性がある．一方，2 つ以上のデバイスがなくては利用できず，ユーザがそれらを用意する負担をかけてしまうという点で，単一のデバイスで，リーダとのインタラクションを通じて比較的容易に本人確認を行うことのできる提案手法とは異なる．

9. おわりに

ユーザが頻繁に利用するデバイスのコンテキスト情報の一致を確認することで，結託や窃取によるなりすましに強く，ユーザや SP の負担を軽減する本人確認手法を提案した．登録フェーズでデバイス情報と環境情報，ID 確認情報を要素としたコンテキスト情報を登録し，オフラインでサービスの利用要求を行う検証フェーズでもそれを生成してリーダの目前で検証を行うことで，本人確認を実現した．提案手法は従来手法と比べてユーザと SP の負担を軽減し，窃取や結託によるなりすましを防ぐという要件を並立して満たすことができた．今後の課題として，永続結託攻撃への防御やコンテキスト情報の継続性向上について検討し，それらを利用可能な実装として実現したい．また，本稿の評価では想定される攻撃モデルとその防御手法を具体的に評価するために定性評価を行ったが，攻撃に対する耐性を数値的な指標を用いた定量評価と合わせてさらに詳しい評価を行いたい．

参考文献

[1] Pass Market（オンライン），入手先 ⟨http://passmarket.yahoo.co.jp/⟩（参照 2017-02-27）．

[2] TAPIRS（オンライン），入手先 ⟨https://www.tapirs.co.jp/face-authentication.html⟩（参照 2017-02-27）．

[3] Holt, L.: Increasing real-world security of user ids and passwords, *Proc. Information Security Curriculum Development Conference* (*InfoSecCD*), pp.34–41, ACM (2011).

[4] みずほ銀行（オンライン），入手先 ⟨https://www.mizuhobank.co.jp/start/step/index.html⟩（参照 2017-02-27）．

[5] Okumura, A., Hoshino, T., Handa, S. and Nishiyama, Y.: Identity confirmation to issue tickets using face recognition, Technical Report 2, NEC Informatec Systems Ltd. (2016).

[6] Alaca, F. and Oorshot, P.C. van: Device fingerprinting for augmenting web authentication: Classification and analysis of methods, *Proc. Annual Computer Security Applications Conference* (*ACSAC*), pp.289–301, ACM (2016).

[7] Mowery, K. and Shacham, H.: Pixel perfect fingerprinting canvas in HTML5, *Proc. W2SP*, IEEE Computer Society (2012).

[8] Bursztein, E., Malyshey, A., Pietraszek, T. and Thomas, K.: Picasso: Lightweight device class fingerprinting for web clients, *Proc. Security and Privacy in Smartphones and Mobile Devices* (*SPSM*), pp.93–102, ACM (2016).

[9] 高橋和司，石川貴之，細井理央，安田昂樹，齋藤孝道：スマートフォンにおける Browser Fingerprinting，コンピュータセキュリティシンポジウム 2016 論文集, Vol.2016, pp.1087–1094 (2016).

[10] GitHub, Anonymous browser fingerprint（オンライン），入手先 ⟨https://github.com/Valve/fingerprintjs⟩（参照 2017-02-27）．

[11] Bonneau, J., Herley, C., Oorshot, P.C. van and Stajano, F.: The quest to replace passwords: A framework for comparative evaluation of web authentication schemes, *Proc. IEEE Symposium on Security and Privacy* (*SP*), pp.553–567, IEEE Computer Society (2012).

[12] Azizyan, M., Constandache, I. and Choudhury, R.R.: Surroundsense: Mobile phone localization via ambience fingerprinting, *Proc. Annual International Conference on Mobile Computing and Networking* (*MobiCom*), pp.261–272, ACM (2009).

[13] Yamada, T., Saito, T., Takasu, K. and Takei, N.: Robust identification of browser fingerprint comparison using edit distance, *Broadband and Wireless Computing, Communication and Applications* (*BWCCA*), pp.107–113 (2015).

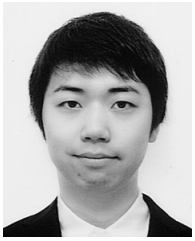
[14] Laperdrix, P., Rudametkin, W. and Baudry, B.: Beauty and the Beast: Diverting modern web browsers to build unique browser fingerprints, *Proc. IEEE Symposium on Security and Privacy* (*SP*) (2016).

[15] Nikiforakis, N., Kapravelos, A., Joosen, W., Kruegel, C., Piessens, F. and Vigna, G.: Cookieless monster: Exploring the ecosystem of web-based device fingerprinting, *Proc. IEEE Symposium on Security and Privacy* (*SP*), pp.541–555, IEEE Computer Society (2013).

[16] Hayashi, E., Pendleton, B., Ozenc, F. and Hong, J.: Webticket: Account management using printable tokens, *Proc. SIGCHI Conference on Human Factors in Computing Systems*, pp.997–1006, ACM (2012).

[17] Tamrakar, S., Ekberg, J.-E. and Asokan, N.: Identity verification schemes for public transport ticketing with nfc phones, *Proc. Scalable Trusted Computing* (*STC*), pp.37–48, ACM (2011).

[18] Karapanos, N., Marforio, C., Soriente, C. and Čapkun, S.: Sound-proof: Usable two-factor authentication based on ambient sound, *USENIX Security Symposium*, pp.483–498, USENIX (2015).



大神 渉　（正会員）

2012 年京都大学大学院情報学研究科知能情報学専攻修士課程修了．同年よりヤフー株式会社．アイデンティティ管理やアクセス制御を通じてユーザビリティとセキュリティに関するコンテキストアウェア技術の研究開発に従事．

五味 秀仁　（正会員）

1996 年京都大学大学院工学研究科応用システム科学専攻修士課程修了．同年日本電気株式会社入社，中央研究所配属．2001～2003 年スタンフォード大学客員研究員．2007 年よりヤフー株式会社．アイデンティティ管理，アクセス制御，プライバシ保護，トラスト管理，コンテキストアウェア技術の研究開発に従事．IEEE，ACM，電子情報通信学会各会員．博士（情報学）．