[ポスター発表] 研究報告

公開鍵認証における所有者検証に基づいた複数認証器のシームレスな登録

畠山 昂大[1,a)]　小谷 大祐[1,b)]　岡部 寿男[1,c)]

Seamless Registration of Multiple Authenticators Based on Authenticator Ownership Verification in Public Key Authentication

1. はじめに

今日多くのサービスにはパスワード認証が欠かせないが、パスワード認証はフィッシングやパスワードの使い回しなど多くの問題を抱えている。そこで、公開鍵暗号を利用した認証（公開鍵認証と呼ぶ）が提案されている。この認証では、ユーザはサービスへの登録時にデバイス上で鍵ペアを生成し、秘密鍵をそのデバイスに残しつつ、公開鍵をサービスに登録する。このような鍵管理機能を持ったデバイスを認証器と呼ぶ。この認証の一つとして FIDO がある。FIDO にはパスワード認証の持つ問題を解決するだけでない利点がある。例えば、鍵ペアを生成した認証器を検証できるアテステーション [*1] という仕組みがあり、サービスは秘密鍵が認証器上の露出できない形で保存されているかなどを検証できる。

しかし、公開鍵認証にはユーザがサービスへのアクセス時に登録済みの公開鍵に対応する秘密鍵を保持する認証器しか使用できない問題がある。スマートフォンやパソコンなどデバイスの複数所持が容易に想定できること、買い替えなどでデバイス更新が起こり得ることを踏まえると、単純に全ての認証器のそれぞれで登録を行うことはユーザが煩わしさを感じてしまう。

そこで、本稿ではユーザが所有する認証器であればどれを用いても公開鍵認証を用いてアクセスできる環境を実現するために、認証器の所有者を検証することで未登録認証器をシームレスに登録できる仕組みを提案する。

2. 関連研究

Sakimura ら [4] の認証連携技術を用いることで登録するサービス数を削減する方法がある。しかし、登録数を削減できるが 1 つになることはなく根本的な解決には至らない。加えて、認証を行う Identity Provider がユーザの利用するサービスを把握できるプライバシー上の問題もある。

Nishimura ら [3] はユーザの所有するデバイス間で秘密鍵を共有することを提案している。デバイスの所有者を検証するために、信頼できる第三者が所有者を識別できる証明書をデバイスに発行する。しかし、公開鍵認証は秘密鍵がデバイスから露出しないことで強い強度を保証できるため認証強度を下げてしまうことや、アテステーションを利用できない問題がある。さらに、信頼できる第三者は WebPKI における認証局のように管理コストが大きく、信頼できなくなった場合の影響が甚大になるなど問題も多い。

Frymann ら [1] や Lundberg ら [2] のアカウントリカバリーに関する研究を紹介する。バックアップデバイスを金庫など安全な場所に保管しつつ、普段使用するメインデバイスがバックアップデバイスの公開鍵を代わりに登録することで、メインデバイスを紛失した場合にバックアップデバイスを使ったサービスへの認証を可能にする。事前にバックアップデバイスの公開鍵のシードとなる情報をメインデバイスが受け取ることで、メインデバイスはバックアップデバイスのサービスごとに異なる公開鍵をシードから導出し登録する。この結果、複数サービスが結託してもバックアップデバイスの公開鍵の同一さを用いた名寄せを防げている。しかし、メインデバイスがバックアップデバイスの代わりに登録するためバックアップデバイスのアテステーションをサービスはリカバリー時にしか検証できない、バックアップデバイスが盗難された場合などのユースケースは想定されていない、などの問題がある。

3. 提案

ユーザが所有する認証器であればどれを用いても公開鍵認証を用いたサービスへのアクセスができる環境を実現するために、未登録認証器を使ったサービスへの初回アクセ

[1] 京都大学
[a)] hatakeyama@net.ist.i.kyoto-u.ac.jp
[b)] kotani@media.kyoto-u.ac.jp
[c)] okabe@i.kyoto-u.ac.jp

[*1] https://fidoalliance.org/fido-technotes-the-truth-about-attestation/

© 2021 Information Processing Society of Japan

ス時に所有者を検証することでクレデンシャル (認証用の鍵) を登録する仕組みを提案する。前節から認証器の所有者検証に信頼できる第三者の存在を仮定しないこと、所有者の証明を悪用し複数のサービスが結託してユーザの名寄せを行えないこと、クレデンシャルは直接その認証器が登録しアテステーションできることを要件とした。

具体的な仕組みとして、1 台目の認証器登録時にクレデンシャルと共に所有者検証用の公開鍵を登録し、2 台目以降の認証器登録時にはクレデンシャルに対してその所有者検証用の秘密鍵で署名する仕組みを提案する。これを図 1 を用いて説明する。サービス α に対してまず認証器 A を用いて新規登録を行い、その後別の認証器 B を用いてアクセスする時を表現している。

まず、認証器間で (1) シード s を共有する。認証器 A でサービスに新規登録する時、まずは通常の登録フローと同じく (2) クレデンシャルを生成し、そのアテステーションを作成する。そして、(3) 乱数 R を生成し、 s を秘密情報としながら鍵共有アルゴリズムを用いて (4) 所有者検証用の秘密鍵 c を計算し、対応する公開鍵 C を導出する。さらに、(5) クレデンシャルに c で署名し、C をメタデータと共に登録する。メタデータは、乱数 R と、サービス識別子 sid_α と R に対して c で作成したメッセージ認証コード MAC からなる。

認証器 B でサービスにアクセスするとき、サービスからメタデータを受け取る。(6) 受け取った R を使って同じように所有者検証用の鍵ペアを導出する。(7) 導出した秘密鍵 c を使って MAC を検証し R が今アクセスしているサービスのために別の所有する認証器上で作られたものだと検証できれば、(8) クレデンシャルの生成とアテステーション及び、(9) クレデンシャルに所有者検証用の秘密鍵 c で署名する。サービスは、クレデンシャルのアテステーションと所有者としての署名を検証し、有効と判断すればそれを登録する。

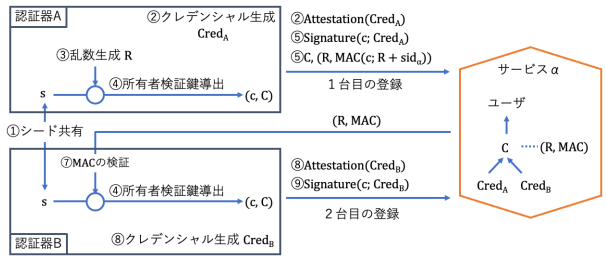

図 1　提案手法

4. 考察

前節に書いた要件が達成できているかを確認する。まず、1 台目の登録時を信頼の起点として所有者検証鍵を登録することで、2 台目以降の所有者検証を信頼できる第三者なしに行うことができている。さらに、所有者検証用の公開鍵 C は乱数 R が異なれば違う値になり C と R からシードを導くことが不可能なため、悪意あるサービス間で所有者検証鍵を共有しても名寄せできない。加えて、受け取ったメタデータから所有者検証用の鍵を導出する際は MAC を確認することで、悪意あるサービス間でメタデータを共有した場合に生じる復号できた事実を元にした名寄せを防げている。また、クレデンシャルの生成は認証器ごとに行うため、クレデンシャルのアテステーションが行えている。

シードについて、鍵共有アルゴリズムを用いて所有する認証器以外では計算できない値にすることが望ましいが、シードの共有と保存方法が安全だとサービスが信頼することは難しい。というのも、サービスは共有に参加した全ての認証器からシードの保管方法を尋ねなければならないからだ。故に、シードが漏洩した場合のサービスやユーザへの被害を軽減するために、サービスは 1 台目の認証器登録時にシード共有を行なった認証器の数を把握すること、この仕組みを使って認証器が登録された時にユーザへ通知を送ること、が考えられる。認証器の数を把握しておけば、全ての認証器が登録された後でシードが漏洩した場合に攻撃者の認証器登録を防ぐことができる。そして、セキュリティ通知を認証器の新規登録時に送ることでユーザは攻撃者が攻撃者自身の認証器を登録した事実を知ることができ、また認証器の新規登録時にのみ送信されるので頻繁に送られることはなくユーザを煩わせることもない。

5. まとめと今後の課題

本稿では、複数認証器のシームレスな登録を行えるように、事前に認証器間でシードを共有することで認証器所有者の証明を第三者なしに、またサービスごとに異なる証拠を使って行える仕組みを提案した。今後の課題として、買い替えや盗難などの認証器のライフサイクルに応じてシードの更新どのように行うべきか、シードが安全に保管されていることをサービスが信頼するためどのような情報を認証器が送信すべきか、がある。

参考文献

[1] Frymann, N. et al.: Asynchronous Remote Key Generation: An Analysis of Yubico's Proposal for W3C WebAuthn, *ACM CCS 2020*, p. 939–954 (2020).
[2] Lundberg, E. and Nilsson, D.: Asynchronous delegated key generation without shared secrets (DRAFT), `https://github.com/Yubico/webauthn-recovery-extension`.
[3] Nishimura, H. et al.: Secure Authentication Key Sharing between Personal Mobile Devices Based on Owner Identity, *Journal of Information Processing*, Vol. 28 (2020).
[4] Sakimura, N. et al.: Final: OpenID Connect Core 1.0 incorporating errata set 1, OpenID Foundation.

© 2021 Information Processing Society of Japan