B-002

IC カードによるユーザ認証システムにおける負荷テスト方式の検討

渡名喜 元史† 八木 礼佳† 山下優人†
株式会社日立製作所‡

1. はじめに

ユーザ認証においては導入が容易なパスワード認証が主流であるが、セキュリティ面で課題が多く近年では多要素認証の採用が増えている[1]。多要素認証ではパスワードのような知識情報の他に、所持情報や生体情報を組み合わせて使用することでセキュリティの向上を実現しているが、処理が複雑になるため負荷テストにおいて認証機能をシミュレートすることが困難になるといった課題がある。

筆者らが負荷テストを実施したシステムは、ユーザ認証に暗証番号（知識情報）と IC カード（所持情報）の 2 要素認証を採用している。本システムは広く利用されるため、大規模（約 40 トランザクション/秒）な負荷をかけ、性能要件を満たしているかを確認する必要があった。

大規模な負荷が必要な場合は負荷テストツールを利用するのが一般的である。テスト対象システムがユーザ認証機能を有する場合、パスワード認証であれば負荷テストツールの適用は容易だが、IC カードによる認証のような物理デバイスを必要とする場合は適用が難しい。これは、物理デバイスを多数備えた環境を構築するのが現実的ではないためである。このため、これまでは負荷テスト時に対象システムの認証機能を一時的に無効化するケースもあった[2]。

本論文では、オープンソースの負荷テストツールに IC カード認証の処理を組み込み、対象システムに手を加えることなく負荷テストを実施した事例を報告する。

2. システム概要

負荷テスト対象のシステムは IC カードに搭載された秘密鍵と電子証明書を利用してユーザ認証を行う機能を有している。

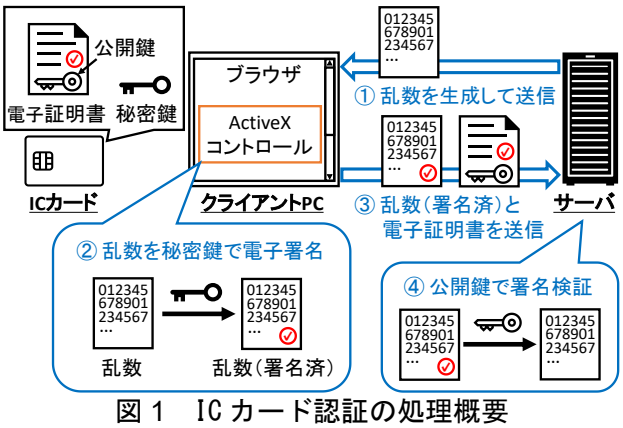

図 1 IC カード認証の処理概要

利用者がパソコンに接続されたカードリーダに IC カードをかざし暗証番号を入力すると、IC カード内の秘密鍵を用いてサーバから送信された乱数が署名される。署名済の乱数は公開鍵として機能する電子証明書と共にサーバへ返送される。サーバ側では署名された乱数を復号・検証することで利用者を認証する。

3. 課題

負荷テストを実施するには大量のテストデータが必要となる。具体的には、限界テストは約 40 万件、耐久テストは約 1,000 万件のテストデータを用意する必要があった。実際に物理デバイスを利用して負荷テストを実施する方法は、テストデータ数分の IC カードを用意し、それらをカードリーダで読み取り認証する必要があるため現実的ではない。

カードリーダのような物理デバイスを必要とするシステムの負荷テストでは、物理デバイスが介在する機能を無効化あるいは簡略化することで、負荷テストツールにより負荷をかけられるようにするのが一般的である。しかし、本システムにおいて上記機能は主要な位置づけであり、当該機能を無効化/簡略化すると負荷テストの目的を達成できない恐れがある。そのため、本システムの負荷テストにおいては、物理デバイスが介在する署名機能（秘密鍵を用いてサーバから送信された乱数に署名する機能）を有効にした状態で、負荷テストツールにより負荷をかける方法を検討した。

物理デバイス無しで負荷テストツールにより IC カード認証をシミュレートするためには、IC カードに搭載されている秘密鍵と電子証明書を用意する必要がある。しかし、IC カードは耐タンパー性を有しており、IC カードの実物を用意してそこから秘密鍵を取り出すという方法は不可能であった。そのため、秘密鍵と電子証明書を別の方法で用意する必要がある。また、クライアント（ブラウザ）側の拡張機能として実現されている署名機能を負荷テストツールに組み込む必要がある。

上記課題の解決に加えて、本システムの開発以降、テストスクリプト作成者以外のプロジェクトメンバが今後主体的にテストを推進できるよう、署名機能について詳細な知識がない人であっても負荷テストを実施できる仕組みの構築をめざした。

4. 施策

サーバ側プログラムに手を加えずに負荷テストツールを実行するためには、クライアント（ブラウザ）側の拡張機能として実現されている署名機能を負荷テストツールに組み込む必要がある。市販およびオープンソースの負荷テストツールでの実現可否を検討した結果、オープンソースの負荷テストツールである Apache JMeter に署名機能を実装する方針とした。Apache JMeter は Java と同等の文法を扱う BeanShell というスクリプト言語で機能拡張できる仕組

A study on load test method for user authentication system using IC (Integrated Circuit) card
† Motofumi Tonaki, Reika Yagi, Yuto Yamashita
‡ Hitachi, Ltd.

みを有しており、サーバ側プログラムが Java で実装されていることから機能的な親和性が高いと判断したためである。

署名機能を Apache JMeter に実装するにあたって、必要となる秘密鍵、電子証明書はバイナリ形式では扱いにくい。通常、Apache JMeter でユーザ ID やパスワードを取り扱う場合はテキスト形式の CSV フォーマットを利用するため、本システムのテストでは事前に PEM (Privacy Enhanced Mail) 相当のテキストデータとして Apache JMeter で扱いやすい書式に変換する方針とした。形式変換には様々な方法が考えられるが、1 つのツールで負荷テスト作業を完結するため、Apache JMeter で変換専用のスクリプトを用意して変換する方式を採った。

テストデータとしての電子証明書は、テスト数分必要となる。このため、テスト用の認証局を立ち上げて負荷テストに必要な数の電子証明書を発行する方針を採った。電子証明書の発行はシェルスクリプトで実装し、必要な件数を任意のタイミングで発行できる仕組みとした。本システムのテストにおいては、サーバ側の署名復号・検証処理の性能確認が目的であるため、電子証明書の発行に使用する秘密鍵はユーザごとに異なるものを用意する必要はない。このことから同一の秘密鍵を用いて、異なる基本情報/詳細情報を持つ電子証明書を発行することで効率化を図った。

テストデータとしての秘密鍵は、システムの特徴から同一のものを使用する方針とした。BeanShell で実装した署名機能では、この秘密鍵を用いて署名用インスタンス（以下署名器と呼称）を生成する。各ユーザで署名処理を行うたびに署名器を生成するのではなく、負荷をかける直前に一度だけ署名器を生成し、以降の処理では生成した署名器を再利用する方針とすることで、効率化を図った。これは Apache JMeter が標準で有する「一度だけ実行されるコントローラ」という機能で実現した。コントローラとはいくつかの処理単位（HTTP リクエスト等）を束ねたものであり、「一度だけ実行されるコントローラ」はその名の通り属する処理を 1 度だけ実行する。「一度だけ実行されるコントローラ」に署名器の生成処理を実装することで、それ以降の処理（ログインからログアウトまで）を複数回実行する場合にも署名器の生成処理は 1 度実行するだけでよい。

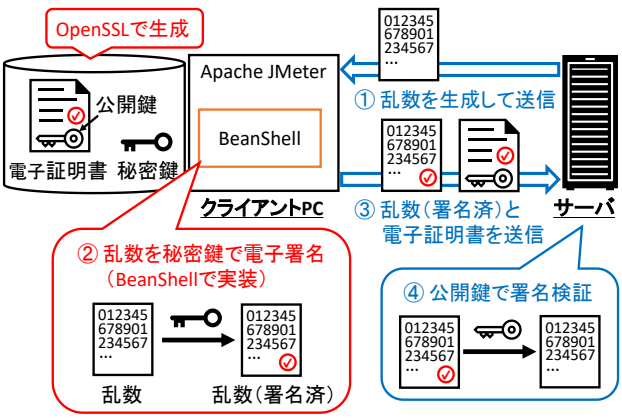

図 2　負荷テストツールに実装した署名機能

5. 評価

署名機能を Apache JMeter に実装することで、テスト対象であるサーバ側プログラムに手を入れず（改修規模：0step）、物理デバイスを使用しない（カードリーダ：0 台、IC カード：0 枚）で、暗証番号と IC カードを用いる 2 要素認証の負荷テストを実現した。限界テストでは 10 分間に約 8 万トランザクションの負荷をかけるテストを実施し、秒間約 135 回の認証を実現、耐久テストにおいては 24 時間に約 400 万トランザクションの負荷をかけるテストを実施し、秒間約 45 回の認証を実現した。

負荷テスト用のテストデータ準備作業においては、秘密鍵は単一、電子証明書はシェルスクリプトで発行できる仕組みとしたことで、大量のテストデータ作成が必要になった際にも、テストデータ作成に必要な証明書の発行をスムーズに行うことができた。また、共通の秘密鍵を利用するものの、基本情報/詳細情報はテストデータごとに一意な値とすることで、エラー発生時の原因特定を容易化した。

負荷テストの実施手順は、以下の 3 ステップとなる。

① シェルスクリプトでのテストデータ用電子証明書発行
② Apache JMeter での電子証明書の CSV フォーマットへの変換
③ Apache JMeter でのテストスクリプトの実行

テストごとにスクリプトの中には手を加えずに署名機能を使用できるような仕組みとし、テストデータの用意は電子証明書の発行のみで良いため、Apache JMeter の操作手順のみを理解できれば、本システムの署名機能について知識がない担当者でも負荷テストを実施できる。難易度の高い操作や知識を必要としないシンプルな仕組みとしたことで、テストスクリプト作成者以外のプロジェクトメンバにおいても負荷テストの実施が可能となった。

6. まとめ

本システムは運用開始以降、大きな問題もなく稼働し続けている。シンプルな実装、仕組みとしたことで、本番稼働以降も、Apache JMeter での物理デバイスなしの負荷テストを何度も実施しており、安定稼働に大きく貢献している。

従来、IC カードのような物理デバイスを必要とするシステムの負荷テストでは、物理的な環境の構築が現実的ではないため一部機能を無効にするといった対応が必要であった。本システムでは負荷テストツールである Apache JMeter に物理デバイスの処理を作りこむことで、テスト対象プログラムに手を入れずに負荷テストを実現する方式を明確にした。

システムのセキュリティを担保する上で認証は重要な要素の一つであり、本論文において複雑な認証機能の性能をテストしたことは非常に有意義であると考える。今後は、近年増加している生体認証[1]など多様な認証機能に対しても同様に、対象システムに手を加えることなく負荷テストを実現する方式を検討する。

参考文献

[1] Ometov, A., Bezzateev, S., Mäkitalo, N., Andreev, S., Mikkonen, T., Koucheryavy, Y., “Multi-Factor Authentication: A Survey”, Cryptography, Vol.2, No.1 (2018).
[2] NTT DATA INTRAMART CORPORATION, “intra-mart Accel Platform セットアップガイド”, Vol.40, https://document.intra-mart.jp/library/iap/public/setup/iap_setup_guide/index.html, 11.13.2.2.1 (参照 2023-5-26).