クライアント側におけるセッション管理上の脆弱性検査手法

内海正貴[†1] 小菅祐史[†1] 河野健二[†1,†2]

1. はじめに

ショッピングサイト等のウェブアプリケーションの多くは，セッション管理を行うことで利用者ごとに専用のコンテンツを配信している．セッション管理とは，利用者にアクセス毎にセッション ID と呼ばれる変数を送らせることによって，ウェブアプリケーション側で利用者の認識を行う仕組みである．

セッション管理は適切に行う必要がある．適切に行われていない場合，攻撃者はウェブアプリケーションの正規の利用者のユーザ名やパスワードを知らなくても，その利用者がログインした後にしか行えない操作を実行できてしまう．このような攻撃に対し，適切な対策手法が存在している．ところが，対策が複雑なため実装を誤っていたり，間違った対策手法をとっていたりといったウェブアプリケーションが多く存在している．WhiteHat Security[1)] によると，ウェブアプリケーションの約 14%が Session Fixation に対する脆弱性を，約 24%が Cross-Site Request Forgery(CSRF) に対する脆弱性を持っている．このように多くのウェブアプリケーションが脆弱性を含んでいるため，ユーザがウェブアプリケーションを利用する際，セッション管理上の脆弱性が無いことを確認する手段が有用であると考えられる．

2. 提　　案

本研究では，ウェブアプリケーションのセッション管理上の脆弱性の有無を，クライアント側から検査する手法を提案する．ウェブアプリケーションの利用者が提案機構を用いて検査することで，利用者が自衛を行うための判断材料となることを目的とする．提案機構の概要を図 1 に示す．提案機構は HTML や変数の取得を行うリクエスト・レスポンスの処理部分と，各脆弱性検査を行う部分に分かれる．本研究では Session Fixaton，Session Prediction，CSRF という 3 つの攻撃を対象とし，脆弱性の有無を検査する．

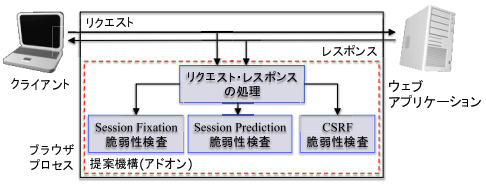

図 1　提案機構の概要

2.1 Session Fixation 脆弱性の検査手法

Session Fixation とは，攻撃者が用意したセッション ID を被害者に使わせることで被害者のアカウントを乗っ取る攻撃である．攻撃者は脆弱性のあるウェブアプリケーションから入手したセッション ID を被害者に送りつけ，被害者は攻撃者の誘導によってそのセッション ID を用いて脆弱性のあるウェブアプリケーションにログインしてしまう．これにより，攻撃者は自分が得たセッション ID を用いて脆弱性のあるウェブアプリケーションにアクセスすることで，被害者の専用ページを閲覧できるようになる．

Session Fixation への対策には，ログイン前のセッション ID は用いずに，ログイン後にはセッション ID を付け替える手法が有効である．新たにログインしたユーザのセッション ID は，攻撃者によって用意されたセッション ID と異なるため，攻撃者はそのユーザのセッションにアクセスできない．そこで，対策がとられているかを検査するために，ログイン前とログイン後のセッション ID の値を比較する．値が異なっている場合，対策がとられていると考えることができる．この検査を行うためには，1) ログイン前後のリクエストの判別，2) リクエストに含まれる複数の変数の内，どれがセッション ID なのかの判別が必要がある．

2.1.1 ログインページの判別

ログインページを判別することができれば，リクエストがログイン前のものか，ログイン後のものかがわかる．一般的に，ログインを行うためにはユーザ名とパスワードを入力する必要がある．そこで，提案機構は HTML の解析を行い，ウェブページが入力フォー

†1 慶應義塾大学
Keio University
†2 科学技術振興機構 戦略的創造研究推進事業
JST CREST

ムを含み，かつその入力フォームにテキスト入力欄とパスワード入力欄がある場合，そのページをログインページであると判断する．

2.1.2 セッション ID の抽出

セッション ID を抽出することができれば，実際にログイン前とログイン後で値が変わっているかを判断することができる．そこで，提案機構はセッション ID が利用者を識別するための変数であることに着目し，アクセスする度に異なる値が割り振られ，かつログイン前・後の両方に存在する変数をセッション ID であると判断する．

2.2 Session Prediction 脆弱性の検査手法

Session Prediction とは，攻撃者が脆弱性のあるウェブアプリケーションで使用されているセッション ID を推測し，被害者のアカウントを乗っ取る攻撃である．攻撃者は脆弱性のあるウェブアプリケーションにアクセスすることでセッション ID を入手し，そこからセッション ID の規則性を見つけ出し，アクセスを試みることで被害者の専用ページを閲覧できるようになる．

Session Prediction への対策には，セッション ID を十分に長いランダムな文字列にすることが有効である．この手法を用いることで，攻撃者は容易にセッション ID を推測することができなくなる．そこで，対策がとられているかを検査するために，複数回ログインを行い，その度に発行されるセッション ID を比較する．取得したセッション ID の値に規則性がなく，十分な長さの文字列であれば，対策がとられていると考えることができる．

2.3 CSRF 脆弱性の検査手法

CSRF とは，脆弱性のあるウェブアプリケーションにおける被害者のログイン状態を利用し，被害者の意図しない操作を行わせる攻撃である．攻撃者は悪意のあるスクリプトを含むウェブページを用意し，被害者をこのページに誘導する．被害者がこのページにアクセスすると，悪意のあるスクリプトが自動的に脆弱性のあるウェブアプリケーションにリクエストを送る．リクエストを受け取った脆弱性のあるウェブアプリケーションでは，被害者が直接リクエストを送ってきたと判断し，処理を行ってしまう．

CSRF への対策には，Referer ヘッダやウェブアプリケーション側で用意した変数の確認によって，正しいページから送られたリクエストであるかを判断する方法が有効である．この手法を用いることで，攻撃者が用意したウェブページからのリクエストは全て遮断することができる．そこで，対策がとられているかを検査するために，これらの値を除いてリクエストを発行する．発行したリクエストに対して期待したレスポンスが帰ってこない場合，対策がとられていると考えることができる．

3. 関 連 研 究

セッション管理上の脆弱性対策手法として BASS[2] や Origin ヘッダ[3] がある．BASS は，Session Fixation に対する脆弱性対策をブラックボックス化し，自動で対策を行うサーバサイド・スクリプト言語である．Origin ヘッダは，CSRF 対策として有用である Referer ヘッダにおけるプライバシー上の問題を解決したヘッダである．セッション管理における脆弱性に対する研究はサーバ側での対策が主流となっている．

一方，クライアント側で脆弱性の検査を行う研究として SecuBat[4] がある．SecuBat は入力フォームに既知の攻撃を入力してリクエストを送り，そのレスポンスを解析することで脆弱性検査を行う．しかし，対象の攻撃は SQL Injection と Cross-Site Scripting となっており，セッション管理上の脆弱性は検査していない．

4. 現状と今後の予定

各脆弱性に対する検査機構を設計し，リクエスト・レスポンスの処理部分及び Session Fixation に対する検査機構を実装した．今後は Session Prediction, CSRF に関しても検査機構の実装を行い，実際に運用されているウェブアプリケーションに対して実験を行うことで，提案手法の有用性を示す予定である．

参 考 文 献

1) WhiteHat Security, Inc.: WhiteHat Website Security Statistic Report Winter 2011, 11th Edition, `http://www.whitehatsec.com/resource/stats.html`.

2) Yu, D., Chander, A., Inamura, H. and Serikov, I.: Better abstractions for secure server-side scripting, *Proceeding of the 17th international conference on World Wide Web (WWW '08)*, pp.507–516 (2008).

3) Barth, A., Jackson, C. and Mitchell, J.C.: Robust Defenses for Cross-Site Request Forgery, *Proceedings of the 15th ACM conference on Computer and Communications Security (CCS '08)*, pp.75–88 (2008).

4) Kals, S., Kirda, E., Kruegel, C. and Jovanovic, N.: SecuBat: a web vulnerability scanner, *Proceedings of the 15th international conference on World Wide Web (WWW '06)*, pp.247–256 (2006).