7ZA-02

OAuth/OpenID Connect実装におけるセキュリティ状況の調査

菊田　翼†　　真木　康太郎†　　細谷　竜平‡　　八代　哲‡　　齋藤　孝道†
明治大学†　　明治大学大学院‡

1　はじめに

Webアプリケーションを利用する際の認証において利用されるソーシャルログインにはOAuthやOpenID Connectが用いられることが多い．OAuthにはバージョンが複数存在し，推奨されている最新バージョンはOAuth2.0[1][2]である．しかし，ソーシャルログインはWebアプリケーションの実装によっては，プライバシー上の問題を引き起こすことや攻撃に対し脆弱となることがある[3]．本論文では，Webサイトのログインページを巡回し，認証フローを辿ることで，そのWebサイトにおける，Google，FacebookおよびTwitterを利用したソーシャルログインの実装状況を，セキュリティの観点で調査した．その結果，SNSへのアクセス権限を必要以上に取得しているWebサイトや，実装上の欠陥によりCSRF脆弱性が存在している可能性を持つWebサイトが確認された．

2　関連知識

2.1　OAuth/OpenID Connectにおける Cross-Site Request Forgery

OAuth/OpenID ConnectにおけるCross-Site Request Forgery（CSRF）では，攻撃者が自らの環境で認証のフローを中断し，攻撃者が用意した意図しないリクエストを送信させるWebサイトに標的を何らかの形でアクセスさせることで，中断した処理を標的に続行させる，攻撃者のアカウントと標的を連携させる手法が挙げられる．その対策として，認証を開始するセッションと認可コードを要求するセッションが同一かをクライアントが判定するために，リダイレクトURLのクエリにstateパラメータ[3]を付与して認証を進める方法がある．

3　調査方法

3.1　調査対象

本論文では，Top Sites in Japan - Alexa[4]における上位500サイト（以降，上位500サイトと呼ぶ）を調査対象とした．

3.2　ソーシャルログインの実装状況の調査

調査を行うにあたって，Webブラウザの処理を自動化できるライブラリであるSelenium[5]を用いて，Webサイトをクローリングするためのシステム（以降，クローリングシステムと呼ぶ）を構築した．クローリングシステムを用いて上位500サイトを巡回し，Google，Facebook，およびTwitterを利用したソーシャルログインが実装されているかを調査した．

クローリングシステムは，事前に用意した上位500サイトのログインページのURLを保存したファイルを読み込んで，そのファイルに存在するURL1つ1つに対してアクセスを行う．アクセス先でGoogle，Facebook，Twitterのいずれかの認証画面が表示されていた場合，あらかじめ用意した各サービスのIDとパスワードを入力しログインを実行する．ログインに成功した時に，そのWebサイトにソーシャルログインが実装されているとした．

3.3　必要以上に取得している権限の調査

3.2節の実験で得られた結果を用いて，クライアントが利用者からどの程度のアクセス権を取得しているか，またその中にサービスを利用する上で必要以上に取得しているアクセス権はないかを調査した．なお，連携に用いたGoogle，Facebook，Twitterそれぞれのアカウントの設定ページから，連携されているWebアプリケーション名とそのアプリケーションに委譲しているアクセス権を確認することができる．

3.4　CSRF脆弱性の有無の調査

2.1節で示したCSRF脆弱性について調査を実施する．ソーシャルログインを認可するための認証における通信を，パケットキャプチャツールであるFiddler for Windows[6]を用いてキャプチャした．その結果から，認可コードを送信するリダイレクトURLにstateパラメータが付与されているかを調査した．認可コードとは，クライアントがリソースサーバへアクセスすることを，利用者が許可したことの証明として，リソースサーバが利用者へ受け渡す文字列のことである．stateパラメータが付与されていない場合はCSRF脆弱性を含み，付与されている場合はCSRF脆弱性を含まないとした．ただし，Twitterにおいては OAuth1.0を拡張したTwitterOAuthという独自のフレームワークが用いられているので，この調査の対象とはしなかった．

Measurement Study of Security Implementation in OAuth/OpenID Connect for Web Application
†Tsubasa KIKUTA†Kotaro MAKI‡Ryohei HOSOYA
‡Satoshi YASHIRO †Takamichi SAITO
†Meiji University
‡Graduate School of Meiji University

4 調査結果

4.1 ソーシャルログインの実装状況

Google，Facebook，Twitter のいずれかを用いたソーシャルログインが，上位 500 サイトにおいて，重複も含めてどの程度実装されているかを図 1 に示す．

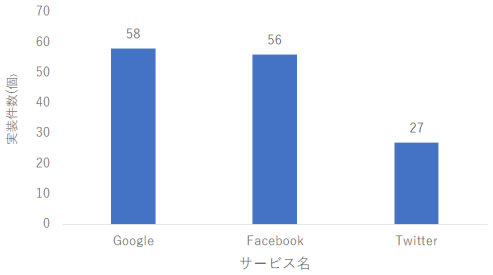

図 1: ソーシャルログインの実装状況

Google と Facebook が実装されているサービスの数は同程度であった．さらに，Google，Facebook，Twitter のいずれかを用いたソーシャルログインが実装されている Web アプリケーションは上位 500 サイト中 79 件存在することが分かった．

4.2 利用者へ要求しているアクセス権

Google，Facebook および Twitter において，クライアントに委譲しているアクセス権を分類した結果を表 1，表 2，および表 3 に示す．分類において，要求されていた数が多かった権限を項目として設定し，少なかった権限はその他に分類した．

表 1: Google におけるアクセス権の分類と件数

付与されていたアクセス権	件数
メールアドレス	43 件
基本プロフィール	40 件
おおよその年齢	6 件
その他	10 件

表 2: Facebook におけるアクセス権の分類と件数

付与されていたアクセス権	件数
公開プロフィール	50 件
メールアドレス	41 件
生年月日	6 件
その他	12 件

調査の結果，Google において，連絡先への全ての編集権限を与えられているサービスが 58 件中 1 件存在した．このサービスは，利用者にオンラインのストレージを提供するサービスであることから，連絡先の全ての編集権限は不要であるとみなせるので，必要以上にアクセス権を取得しているサービスがあると言える．

4.3 state パラメータの調査

3.4 節で示した CSRF 脆弱性について調査した結果を示す．state パラメータが実装されているサービスは Google では 58 件中 26 件，Facebook では 56 件中 38 件，state パラメータが実装されていないサービスは Google では 58 件中 15 件，Facebook では 56 件中 14 件であった．また，Google では 58 件中 17 件，Facebook では 56 件中 4 件のサービスでの通信において，認可コードを含むパケットを取得できなかったので，state パラメータが実装されているかどうか判断できなかった．

表 3: Twiter におけるアクセス権の分類と件数

付与されていたアクセス権	件数
読み取り	28 件
書き込み	20 件
メールアドレス	4 件
ダイレクトメッセージ	2 件

4.3.1 state パラメータの実装差異

state パラメータを表す文字列は，以下の 3 点に分類できることがわかった．

1. 数字の 0 から 9 のみで表された文字列
2. 数字の 0 から 9，および小文字の a から f のみで表された文字列
3. 数字の 0 から 9，および全てのアルファベットで表された文字列

なお，state パラメータの文字数に規則性は見られなかった．これらの結果から state パラメータの実装は全てクライアントに依存していることが考えられる．

5 研究倫理

本論文では，調査対象のサービスに対して悪影響を及ぼさないように注意して実験を行った．また，論文を執筆する際は上位 500 サイト中の特定のサービス名を明示しないように配慮した．

6 まとめ

本論文では，クローリングシステムを用いて上位 500 サイトを巡回し，ソーシャルログインの実装状況を調査した．その結果上位 500 サイト中，79 件の Web サイトでソーシャルログインが実装されており，その中に必要以上に権限を要求していると考えられるサービスが存在していた．また，仕様通りに state パラメータが実装されていないために CSRF 脆弱性が存在している可能性のあるサービスが Google では 15 件，Facebook では 14 件存在した．

参考文献

[1] RFC6749 The OAuth 2.0 Authorization Framework. https://tools.ietf.org/html/rfc6749.

[2] RFC6750 The OAuth 2.0 Authorization Framework: Bearer Token Usage. https://tools.ietf.org/html/rfc6750.

[3] RFC6819 OAuth 2.0 Threat Model and Security Considerations. https://tools.ietf.org/html/rfc6819.

[4] Top Sites in Japan - Alexa. https://www.alexa.com/topsites/countries/JP.

[5] Selenium - Web Browser Automation. https://www.seleniumhq.org/.

[6] Fiddler Web Debugging Tool for Free by Telerik. https://www.telerik.com/fiddler.