Users' Acceptance Study of OAuth Manager Module for Social Login in Mobile Environment

Lee Kah Ho and Norliza Katuk
School of Computing, Universiti Utara Malaysia
06010 UUM Sintok, Kedah, Malaysia
k.norliza@uum.edu.my

***Abstract*—Social login is a way that allows social network users to use their credential to log in to other applications. Currently, many developers make use of Open Authorization (OAuth) protocol to implement social login (SL). The design of OAuth protocol works well on workstations and desktops as they uniformly use web browsers to access web applications. However, it is exposed to security issues when it is moved to the mobile environment. Although native mobile applications are installed on the mobile devices, this protocol will call system browsers to complete the task; hence, exposing users to token redirection attacks. In overcoming the issue, this study attempts to evaluate a method called OAuth Manager Module (OMM) that aims to improve the security of this protocol in a mobile environment. It provides client isolation to prevent malicious actions during the social login process. A controlled experiment was conducted to evaluate user acceptance towards OMM. A within-subject design was conducted on thirty participants who participated in this study on a voluntary basis. The results show that users perceived OMM useful and easy-to-use compared to social login with system browser. However, in overall, users are still worried about the security of using social logins on mobile devices. This study can further serve as a foundation for various research on the security aspect of social login.**

***Index Terms*—Mobile Applications; Single Sign-On; Authentication Protocols; Usable Security.**

I. Introduction

It is known that the emergent of online social media applications has changed the way people communicate and socialize [1]. The impact of social media application can be seen at all level of users; regardless of their age. Not only individual users, but businesses are also making the most of the technology to get connected to their customers for marketing purposes [2]. Apart from providing corporate social media sites, organizations can access the social media users' profile by embedding social login (SL) [3-6] facility in their applications. It is an authentication mechanism that allows users to use their social network credential to access third-party applications without the need to register themselves to the application providers. SL is a variant of single sign-on (SSO) mechanism. It is a design of independent software systems that allows users to log in once to gain access to these systems without being asked for login credential again and again. SSO mechanism has been implemented by application providers as a way to reduce password fatigue [7].

Among various SSO implementations, Open Authorization (OAuth) provides a good standard interface for allowing third-party applications to request private resources on behalf of the users from a web server. This simplifies users' actions from making duplicated resources (e.g., photo) in different web services. Instead, users can share their resources from one web-based application to other platforms by using OAuth. However, in the current trend of mobile devices, OAuth cannot be implemented exactly how it used to be in a normal web application environment. OAuth highly relies on browser redirections of the access token. Simply said, users who would like to access third-party applications using their social network credential would be asked to provide their credential in a new separate mobile browser page. Then, when users supply their social network credential, the information will be redirected to the authentication server and back to the mobile browser page where the third-party applications started. The use of multiple mobile browser pages for communicating the users' social network credential could expose the communication to token redirection attacks [8].

To overcome this issue, Shehab and Mohsen [9] proposed OAuth Manager Module (OMM) programmed in mobile devices which aims to provide security of information exchange of native mobile applications by minimizing the browser redirections. In other words, the mobile devices will refrain from opening a new browser page when users would like to use their social network credential to access third-party applications. Hence, token redirection attacks through the use of multiple browser pages can be avoided which makes the communication of users' social network credential safe. The idea of using OMM is excellent and promising better choice for users in accessing third-party applications using their social network credential. However, the module should also be tested on the users to see how they perceived the way of communicating their social network credentials so that the usability of such module is confirmed.

To our knowledge, usability study of OMM tested on users has not yet been conducted yet. Measuring usability is an essential task to ensure the accuracy of the module [10] and as well as users' acceptance of the technology [11]. Hence, it is justifiable to measure user acceptance towards OMM for the SL implementation; which has been the main aim of this study. The study will focus on the use of OAuth for implementing authentication of SL for third-party native mobile applications. The next section presents an overview of the concepts that relevant to this study. Then, it is followed by the methodology section which explains the steps and procedures to carry out this study. Lastly, the results are presented in the following section, and finally, it is followed by a concluding remark to this research.

II. Related Works

A. SSO and Social Login

The use of username and password (i.e., credential) is a common way for authenticating users in today's Internet-based applications. An addition to the authentication mechanism, it also works as a mechanism for user personalization. However, the rapid increase of applications that asked for this user credential caused password fatigue [7] among users. They feel mental and emotional pressure to remember an excessive number of credentials for different applications as part of their daily routine. To avoid users from writing the credentials on the paper or replicating the same credentials for multiple applications; an identity management system is needed. Then, the SSO comes as a solution to password fatigue and identity management.

SSO allows users to use a single credential for authentication once to access multiple applications from different applications providers [12]. SSO is available in different forms, covering from enterprise solutions to individual needs. The recent development allows application providers to use social network credentials in implementing SSO. It is known as social login (SL) [3, 5, 6]. Users of many social network providers such as Facebook, Twitter, Google+, and Yahoo can use their credential to access third-party applications. This mechanism has been widely implemented in many web and mobile applications. Figure 1 shows an example of SL used in a popular online shopping application.

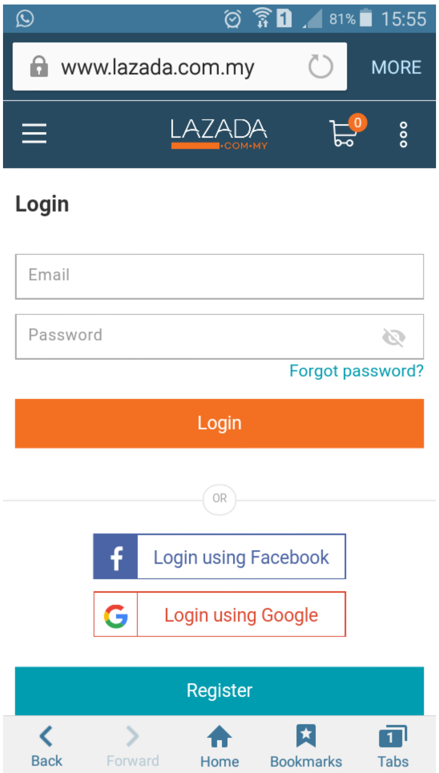

Figure 1: SL used in a popular online shopping application (A screenshot of www.lazada.com.my taken using a mobile device)

B. The Mechanism for Implementing SL and OAuth

SL can be implemented in third-party applications using open standards and protocols such as OpenID and OAuth [3]. Users authenticate to a third-party application can be implemented using credentials issued by supported OpenID identity providers such as Google or Yahoo. The providers supplied API that allows users to have a simplified sign-in process by eliminating the new-member registration process [13]. OpenID provides authentication services to the relying parties. On the other hand, OAuth allows users to authorize an application to act on behalf of the user on another application [14]. However, this protocol has been re-purposed by Facebook, Google, and Microsoft [15] for user authentication.

Communication using OAuth protocol involved four components (parties) [4, 16]; client, resource owner, the authorization server, and resource server. Table 1 defines the components. Figure 2 shows the communication flow among OAuth components.

Table 1
The Components of OAuth Protocol [4, 16]

Components	Description
Resource owner	An entity that allows the client to access protected resources or accounts.
Resource server	A server that validates access tokens presented by a client and serves the requested protected resources. It is the API server that stores users' information.
Client	An application that represents the resource owner and requests for the protected resource. It also referred to as a third-party application
Authorization server	A server that authenticates the resource owner and issues access tokens to the client.
Authorization grant	A resource owner's credential used by clients to obtain an access token from the authorization server.
Access token	A credential for accessing protected resources.

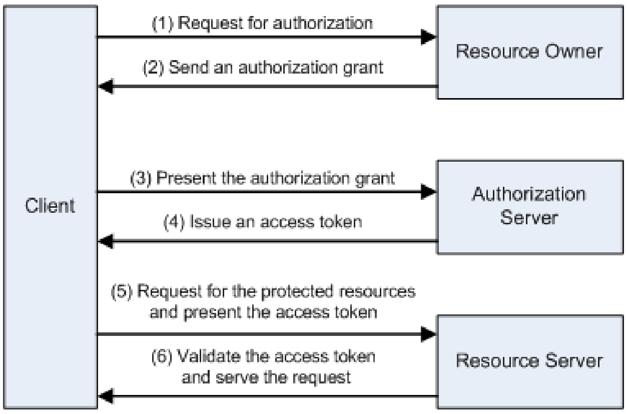
Client
(1) Request for authorization
(2) Send an authorization grant
Resource Owner
(3) Present the authorization grant
(4) Issue an access token
Authorization Server
(5) Request for the protected resources and present the access token
(6) Validate the access token and serve the request
Resource Server

Figure 2: OAuth authorization flow [4, 16]

OAuth uses browser redirection extensively for sending the tokens between the involved parties [15]. Redirections are the central mechanism of OAuth that could open a way for attackers to target such implementation [17].

C. OAuth Token Redirection Using System Browser

System browser refers to web browser applications that are installed on a mobile device. Some popular mobile web browsers are Chrome, Firefox, and Safari. In OAuth authorization flow, a user-agent is needed to perform redirection and isolate the authorization from interrupting other processes. In the mobile platform, some developers choose to use these mobile web browsers to be the role of user-agent in mobile OAuth authorization flow. In OAuth, the client application can send the OAuth authorization link to a browser application for authentication and authorization. The transmission of data and result can be achieved by mobile platform architecture such as Intent Manager in Android [9].

Although using a web browser in the mobile device can achieve an isolated user-agent in OAuth authorization flow, there are still vulnerabilities exist in this implementation. For example, attackers can create a malicious app and register itself to the platform architecture and intercept the OAuth response from a web browser. It will lead to leaking of the legitimate access token to malicious applications.

D. *Secured Centralized OAuth Manager in Mobile Device*

As discussed in the earlier section, the implementation of OAuth for native mobile applications is not straightforward as implementing it in web applications. To implement OAuth in native mobile applications, two problems must be solved:

1. No redirection of user agent: Native mobile apps do not have a web browser to perform redirection which is the core specification of OAuth. An alternative to this problem is either embedding a web view client in the native application or make use of system browser to carry out OAuth process.
2. No isolated user agent: Embedded web view and system browser on mobile devices are still under the control of developers. It increases the risk of credential theft.

In overcoming the above problems, Shehab and Mohsen [9] proposed a centralized OMM to be implemented in mobile devices. The module handles all OAuth authorization requests made by the mobile device and returns the access token to the requesting client after the authorization process completes. Figure 3 illustrates the conceptual design and the process involved in OMM. The flow of the communication is:

1. Application requests activation of OMM from Android Intent Manager (AIM).
2. AIM passes the required parameters to OMM.
3. OMM performs OAuth authentication and authorization process with the SL Provider through an embedded protected web view.
4. OMM returns an access token to AIM.
5. AIM passes the access token back to the application.
6. Application accesses API and resources of SL Provider using the access token.

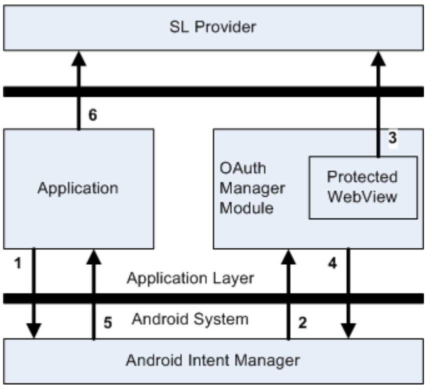

Figure 3: OMM Conceptual Design [4, 9]

Table 2 shows the similarities and differences between system browser and OMM.

Table 2
Comparison between System browser and OMM

Aspects	System Browser	OMM
Isolated user-agent	Yes	Yes
Response can be intercepted	Yes	Can be prevented [9]
User login cookie	Persisted after exit from browser	Erased after exit from module
Re-login required for another app?	No, if the cookie persisted	Yes

From the aspect of similarities, both system browser and OMM isolate user-agent where OAuth clients are unable to disturb the user-agent authentication and authorization process. Next, other application can intercept the OAuth response after authorization process if it is carried out using system browser. However, OMM can be programmed to give warnings if it detects any app trying to register itself to intercept the response.

Thirdly, system browser will automatically keep cookies, and it persists even after the browser exits. It makes the subsequent request skips the login process, and OAuth process automatically completes and may lead to information leak if other people possess the device. However, OMM store cookie only for one OAuth session. The cookie will be removed after the application exits the module.

Similar to the above aspect, system browser does not require users to re-login with the server since the cookie is available. Any subsequent OAuth request is not properly authenticated because it skips the login part. This is, however, will not happen in OMM because the cookie will not persist.

E. *Implementation of OMM*

A native mobile application was developed to demonstrate the implementation of OMM. The name of the application is Photo Tagger. This native mobile application allows users to take a photo and tag the photo from users' Facebook friend's list. The prototype was installed on Samsung smartphones running OS Android 4.4.2. Table 3 shows the hardware and software specification for developing Photo Tagger.

Table 3
Hardware and Software Specification

Hardware	Software
System Type : x64 CPU : Intel i5 2.4 GHz RAM : 4.0 GB GPU : ATI Mobility Radeon HD 5470 Disk Space : 500 GB	Windows 7 Eclipse IDE Android SDK 4.3 (API 18)

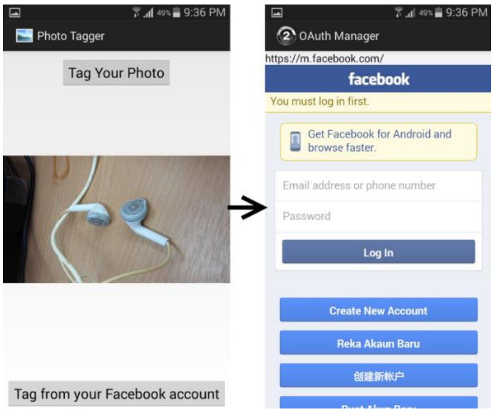

Figure 4: The process for tagging a photo using OMM in Photo Tagger

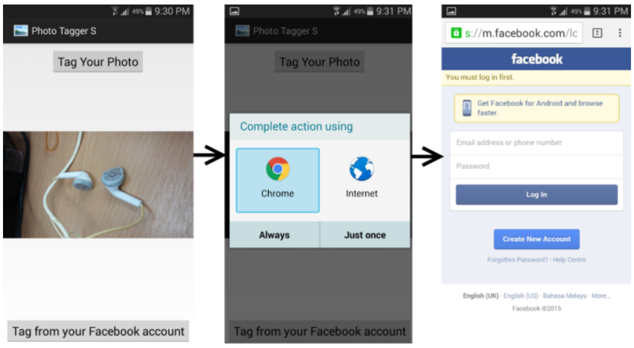

Figure 5: The process for tagging a photo using system browser in Photo Tagger

Figure 4 shows the interfaces of Photo Tagger application that implements OMM. Once a user snapped a photo using Photo Tagger, the user can touch “Tag from your Facebook account” button, and the OMM straight away call for Facebook login interface. Then, the user is required to provide their Facebook username and password to access his/her friend list. As a comparison, access to Facebook’s friend list using system browser was also developed in Photo Tagging application. Figure 5 shows the interface of the application when system browser is implemented. The obvious difference between both authentication methods is the elimination of system browser call.

In Shehab and Mohsen’s study, they evaluate the OMM from the aspect of performance, which included CPU and memory consumption. However, the usability and user acceptance of OMM are yet to be evaluated. Hence, this study aims to evaluate users’ acceptance of OMM using Photo Tagging as the main instrument.

III. Methodology

A controlled experiment was carried out to investigate users’ acceptance towards OMM. The independent variable was SL approach, which is using OMM and system browser. The dependent variable is user acceptance of the SL mechanism. The hypothesis is “users have a similar perception on the usefulness and ease-of-use of OAuth system browser and OMM.” 30 participants participated in this study on a voluntary basis. They comprised 15 males and 15 females, aged between 21 to 45 years. They were recruited through an advertisement on Facebook, and they were invited for a face-to-face meeting with the principal researcher at a few convenient venues for conducting the experiment.

Photo Tagger application was the main instrument used in this experimental study (as explained in the previous section). Photo Tagger has both OMM and system browser for SL. A Facebook account that contained a list of friends/contacts was created for this experiment. A post-task questionnaire was used in this study to measure user acceptance towards OMM. It was adapted from Davis [18]. It contained nineteen items (ten for perceived usefulness and nine for ease-of-use) and measured using a seven-point Likert scale (i.e., one represented ‘strongly disagree’ and seven represented ‘strongly agree’). The questions are listed in Table 4. The participants were also asked about their preference of SL mechanism at the end of the post-task questionnaire.

Table 4
The post-task questionnaire

Perceived usefulness	Perceived ease-of-use
• OMM enhances my effectiveness in accessing the Photo Tagger. • OMM increases my productivity. • OMM makes it easier to access Facebook resource. • OMM gives me greater control over my work. • OMM enables me to accomplish tasks more quickly. • OMM saves my time when I use it. • OMM meets my needs. • OMM does everything I would expect it to do. • OMM is useful in overall.	• OMM is easy to use. • OMM is user-friendly • OMM is flexible. • OMM requires fewer steps to accomplish what I want to do with accessing Facebook resource. • OMM is easy to learn how to use it. • OMM can be used without written instructions. • I can easily remember how to use it. • I don't notice any inconsistencies as I perform OMM in Photo Tagger. • I can recover from mistakes quickly and easily when accessing Photo Tagger using OMM. • I can use Photo Tagger with OMM successfully every time.

The procedure for carrying out the experiment is:

1. The participants read the information sheet.
2. The participants sign the consent form.
3. The participants fill up the background information.
4. The participants snap a photo using the Photo Tagger applications.
5. The participants touch the “Tag form your Facebook account” menu and tags friends from Facebook contact list (the sequence of OMM and system browser for each participant was assigned at random).
6. The participants answer the post-task questionnaire.

IV. Results

The Cronbach’s Alpha coefficients for the nineteen items are between 0.89 and 0.97, indicating that the data are internally consistent. A Wilcoxon Signed Rank test was conducted to evaluate the participants’ responses on the user acceptance of the two SL mechanisms. User acceptance was measured from the aspects of usefulness and ease-of-use. There is a statistically significant difference in terms of the user acceptance towards OMM and system browser.

By looking at the overall user acceptance response, the result yields a significant difference, with $z = -3.93$, $p < 0.001$, and with a large effect size ($r = 0.51$). From the aspect of usefulness, the result yields a significant difference with $z = -3.95$, $p < 0.001$. From the aspect of ease-of-use, the result also yield a significant difference with $z = -3.57$, $p < 0.001$.

Further, the participants’ responses on their preference of SL mechanism; it is revealed that 7 participants (23%) preferred to use system browser for the SL mechanism, while 23 participants (77%) preferred using OMM. From the analysis, 2 participants (9%) stated that the OMM does not cache the login credentials when logging into the Facebook account. There were 4 participants (17%) stated that OMM was easier to use as compared to Chrome browser. Another 3 participants (13%) stated that OMM was more convenient to be used as compared to the system browser. However, there were 14 participants (61%) did not specify any reason for choosing OMM as their preferred SL mechanism. On the other hand, there was only 1 participant (14%, out of 7 participants) who provided the reason for choosing system browser as his/her preferred SL mechanism. Hence the only reason for choosing system browser that can be observed is that the browser is developed by a more trustable party. The results suggested that most participants perceived usefulness and ease-of-use towards OMM over system browser; therefore the hypothesis of this study is accepted.

V. Discussion And Conclusion

The research suggested that OMM is useful and easy-to-use that is supported by the participants' feedback. The result is highly affected by an extra step required by the participants to choose the browser they wished to use for the SL mechanism. On the contrary, the participants did not have to choose the browser for SL mechanism when using OMM. Furthermore, the switching between the client application (Photo Tagging application) and OMM is faster because it is more lightweight as compared to the browser application. However many users think that SL is not secure as their login credential might be stolen after performing user authentication. It may be caused by two factors. Firstly, the client application resides on the users' mobile devices and it is possible to have access to the browser cache. Unlike desktop browsers, browser cache that contains login credential in mobile devices is more exposed to security threats. Secondly, users feel insecure to trust any intermediate applications or clients to carry out authentication process which involves their login credential. Although OMM is designed to be more secure in technical point of view, users still feel insecure because it is not famous, or widely admitted as secured as compared to browser applications.

The outcome of this study could be used as a reference for developers that implement OAuth protocol for SL. In future, a study on token redirection attack in OAuth should be carried out. Further, necessary actions to mitigate this attack can be another potential area of study for researchers to explore.

References

[1] N. J. Yuan, Y. Zhong, F. Zhang, X. Xie, C.-Y. Lin, and Y. Rui, "Who will reply to/retweet this tweet?: The dynamics of intimacy from online social interactions," in *Proc. of the 9th ACM Int. Conf. on Web Search and Data Mining*, 2016, pp. 3-12.

[2] D. L. King, "How to Connect with and Communicate with Customers," *Library Technology Reports,* vol. 51, pp. 16-21, 2015.

[3] R. Gafni and D. Nissim, "To social login or not login? Exploring factors affecting the decision," *Issues in Informing Science and Information Technology,* vol. 11, pp. 57-72, 2014.

[4] L. K. Ho and N. Katuk, "Social login with OAuth for mobile applications: User's view," in *Proc. of the 2016 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE),* 2016, pp. 89-94.

[5] G. Kontaxis, M. Polychronakis, and E. P. Markatos, "Minimizing information disclosure to third parties in social login platforms," *Int. Journal of Information Security,* pp. 1-12, 2012.

[6] L. K. Moey, N. Katuk, and M. H. Omar, "Social login privacy alert: Does it improve privacy awareness of Facebook users?," in *Proc. of the 2016 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, 2016, pp. 95-100.

[7] S.-T. Sun, Y. Boshmaf, K. Hawkey, and K. Beznosov, "A billion keys, but few locks: the crisis of web single sign-on," in *Proc. of the 2010 Workshop on New Security Paradigms*, 2010, pp. 61-72.

[8] F. Yang and S. Manoharan, "A security analysis of the OAuth protocol," in *Proc. of the 2013 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM),* 2013, pp. 271-276.

[9] M. Shehab and F. Mohsen, "Towards Enhancing the Security of OAuth Implementations in Smart Phones," in *Proc. of the 2014 IEEE International Conference on Mobile Services (MS),* 2014, pp. 39-46.

[10] A. Hussain, N. L. Hashim, N. Nordin, and H. M. Tahir, "A metric-based evaluation model for applications on mobile phones," *Journal of ICT,* vol. 12, pp. 55-71, 2013.

[11] H. Ibrahim and T. A. Al-Rawashdeh, "Acceptance of web-based training system among public sector employees," *Journal of Information & Communication Technology,* vol. 13, pp. 87-107, 2014.

[12] A. Pashalidis and C. J. Mitchell, "A taxonomy of single sign-on systems," in *Proc. of the Australasian Conf. on Information Security and Privacy*, 2003, pp. 249-264.

[13] M. N. Ko, G. P. Cheek, M. Shehab, and R. Sandhu, "Social-networks connect services," *Computer,* vol. 43, pp. 37-43, 2010.

[14] A. Vapen, N. Carlsson, A. Mahanti, and N. Shahmehri, "A look at the third-party identity management landscape," *IEEE Internet Computing,* vol. 20, pp. 18-25, 2016.

[15] E. Y. Chen, Y. Pei, S. Chen, Y. Tian, R. Kotcher, and P. Tague, "Oauth demystified for mobile application developers," in *Proc. of the 2014 ACM SIGSAC Conf. on Computer and Communications Security*, 2014, pp. 892-903.

[16] D. Hardt, "The OAuth 2.0 authorization framework," 2012.

[17] B. Leiba, "Oauth web authorization protocol," *IEEE Internet Computing,* vol. 16, pp. 74-77, 2012.

[18] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Quarterly,* pp. 319-340, 1989.