Classification of Third-Party Applications on Facebook to Mitigate Users' Information Leakage

Sanaz Kavianpour(✉), Zuraini Ismail, and Bharanidharan Shanmugam

Advanced Informatics School, Universiti Teknologi Malaysia, Kuala Lumpur, Malaysia
ksanaz3@live.utm.my, zurainiismail.kl@utm.my,
Bharanidharan.Shanmugam@cdu.edu.au

Abstract. Facebook is significant platform for third-party developers to run written applications in order to provide users extra functionality and services. Third-party applications (TPAs) access to user's profile and exchange their information. In doing so, this may lead to information leakage and privacy risks. Although Facebook has control over third-party applications, it still lacks control in the existing mechanisms. The aim of this paper is to investigate how to hinder TPAs from accessing user's private information while still sustaining the functionality of the applications. To address privacy and functionality simultaneously, this study suggests a classification framework providing mechanism in controlling TPAs access to the users' data residing on Facebook. The improved framework allows TPAs to utilize some of users' data according to their classification authority to mitigate users' information leakage.

Keywords: Classification · Online social networks · Privacy · Third-party applications

1 Introduction

Online social networks (OSNs) are a significant target for marketers, government agencies, and online predators as they accumulate terabytes of users' data [1]. OSNs have a popular feature which is known as third-party applications (TPAs) that are used by hundred millions of users per day. TPAs were initiated by presenting the "Facebook Platform" in May 2007 [2]. Facebook platform is based on graph application programming interface (API) that provides an exclusive software environment which able developers to create and run their applications. TPAs need users to grant access to their personal data in order to offer better functionality. Most OSNs offer to accept all or nothing mechanism for managing TPAs permissions to access a user's private data. This means that a user has no control over sharing only a subset of information unless not installing and using the application. On Facebook, a user can select what type of data he/she is eager to share with TPAs but he/she can approve all TPAs requests or deny all. Felt et al. have shown that 91% of the 150 top Facebook applications have unnecessary access to user's private data which can violate the principle of least privilege [3].

Á. Rocha et al. (eds.), *Recent Advances in Information Systems and Technologies*,
Advances in Intelligent Systems and Computing 569, DOI 10.1007/978-3-319-56535-4_15

Facebook uses permission-based platform security that decides what permissions can be granted to a given application while there are some sort of platforms that are based on the user-consent permission systems. User-consent permission systems have main security issue as users may get used to permission queries and grant access due to their carelessness or unreliable signals that manipulated by malicious applications developers [4]. TPAs are hosted on external servers which is beyond the Facebook control [5]. Hence, once a TPA gains access to users' data, there is no control on the usage and propagation of users' data. Limiting TPAs from a user data completely or providing access to bogus data in reply to their request may harm functionality or even their own business models. Thus, Facebook need to consider efficient techniques deals with TPAs in order to preclude users' privacy leakage.

In this paper in order to limit and control the TPAs access to user data, a novel classification framework for Facebook applications is proposed. The proposed framework analyzes sample of TPAs on Facebook and the information exchange between Facebook and TPAs. Then, it automatically classifies TPAs based on their features such as category, rating, required permissions set, external link to post ratio and website reputation scores (WOT) [6]. The findings depict that the proposed framework is feasible and useful in preventing sensitive information leakage to third-parties, as well as retaining TPAs functionality by providing explicit required access to users' data based on their class authority automatically. Proposed framework provides accurate control over exchanging information from Facebook to TPAs without users' interference.

This section introduces the paper while the following Sect. 2, describes the background problem with current OSNs platforms. Privacy issues of inappropriate exposure of user's information are presented in Sect. 3. Section 4, provides an overview of Facebook platform and presents privacy setting options on Facebook respectively. TPAs Classification Framework is illustrated in Sect. 5 in details. Section 6, specifies the experimental results. The paper concludes with a discussion on directions for future work in Sect. 7.

2 Background Problem with Current OSNs Platforms

In application-to-user interactions, OSNs apply only an all-or-nothing mechanism which is different from user-to-user interactions [7]. This mechanism may not let the users to control TPAs access to data according to their privacy preferences. Although some OSNs have extra privacy setting options deal with TPAs that let users to opt-in or opt-out some categories of profile data, privacy issues are still remain.

Privacy control for TPAs on Facebook are based on coarse-grain granularity of permissions, so TPAs can request for unnecessary data [5]. The real type of actions which TPAs can exercise on data are not specified entirely. Also, TPAs can receive permissions by user's friend who installed the application while user is not aware of. All these issues can violate users' privacy, consequently some mechanism is required to mediate TPAs access upon their requests by Facebook provider to mitigate exposure of unnecessary private data.

3 Privacy Issues of Inappropriate Exposure of User's Information

Facebook is one of the most significant challenging channel of information leakage [8]. Facebook allows third-parties to run their applications on this platform for their business needs to attract more users daily and magnify their networks. TPAs require access to users' information to run their services. Mostly, TPAs have access to users' public information by default, and they can access to private information by users grant as well. Granting access to TPAs may lead to privacy breaches as there are some malicious applications which are not following the privacy policies [9].

Users' demand to have more contacts, share as much as information on Facebook while being private as well arise privacy challenging issues. The main privacy issue occurs when TPAs commence to abuse their access to users' private data against users' expectations including disclosure of personal information to advertisers or sell users' private data to marketers [10]. Users' privacy can be threaten by vast information exchange which may lead to unintentional information disclosure, damaged reputation and image, unwanted stalking, and reconstruction of users' identities [11]. Thus, our principal motivation is to propose mechanism in order to protect users' private data from being leaked by TPAs even intentionally or unintentionally while keeping TPAs functionality. Our mechanism will grant access to TPAs according to the classification authority and will control information flow from Facebook to TPAs to protect users' data. The details of the TPAs classification and access levels are described in Sect. 5.

4 Overview of Facebook

In this section, the Facebook platform and privacy setting options on Facebook are described respectively. Facebook is selected as it is the most popular OSNs which offers TPAs providing games and entertainment possibilities [12].

4.1 Facebook Platform

Facebook is a huge repository of data which encompasses users' personal data and users' logs interaction information with friends as well as their activities. It also comprises of TPAs that extracting identifiable user data in order to share or sell it to advertisers or marketers [13]. The information interaction flow between TPAs and Facebook users is shown in Fig. 1.

Facebook applications developers will proxied their homepage on Facebook or in an iframe. Users can download these applications from the Facebook App center [14] and grant access to the applications requests in order to use them. Each application requires a valid user session key and application secret in order to query the Facebook server. Facebook will provide access token to TPAs developers after users' grant authorization to the application. Once the access token is given, application can collect the user's personal data and share or sell it out of user's control.

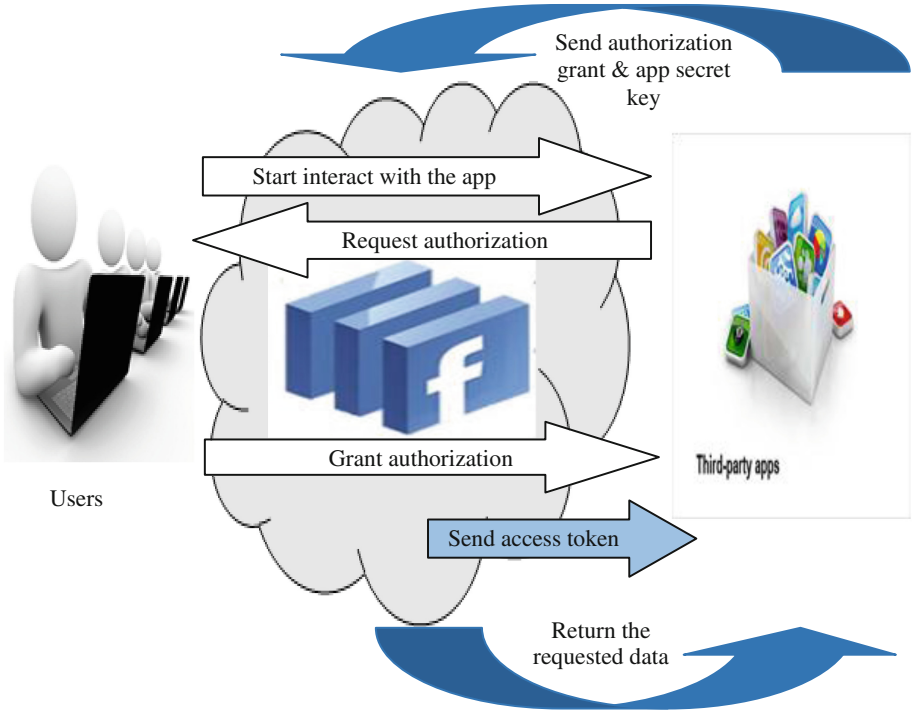

Fig. 1. Architecture of information interaction flow between TPAs and Facebook users

4.2 Privacy Setting Options on Facebook

A user can control propagation of his/her information among other users partially by adjusting privacy settings options which is provided by Facebook. They are able to limit who can see what they share, control what others can share on their wall and block users. Facebook use the OAuth 2.0 protocol to authenticate and authorize TPAs. TPAs can access a protected resource on the server once they authorize through OAuth 2.0 by using Facebook users credentials [15]. The default information that can be accessed by the application is "access my basic information" that contains a user name, user ID, profile picture, gender, and any information which the user shared with everyone.

Application developer can request for more permissions if additional information is required such as contact information, friends' list, email, sending messages to users. The top three common requested permissions are: access my profile information, post to my wall and send me email [11]. Once a user installs an application and clicks on it, the authentication dialog will appear on the screen to get the user permission. Developers can make the display of the authentication dialog confusing in order to request extended permissions [11, 16, 17]. This can be done by using difficult English grammar or ambiguous words, so users may find the implication of the extended permissions difficult to understand. Therefore, the authentication dialog may not accurately reveal the TPAs information practices.

On Facebook application settings, users may grant access to applications that are used by their friends by default unless they manually unclick this setting [8]. In this case, if an application request for a user and his/her friends' details such as birthday, the requested data can be accessed upon the user grants access. The birthday details will be available to all even though the user make his/her birthday private. Hence, TPAs can pass users' privacy settings and privacy default settings are not transparent enough to aware users about the nature and amount of data that will be gathered by TPAs and used further than the user's expectations.

5 TPAs Classification Framework

In this section, we present a classification framework that can control the transmission of user's data to TPAs. This proposed framework is designed in order to preserve users' privacy from TPAs whilst retaining the functionality of the applications.

5.1 The Design

Figure 2 depicts an architecture of the TPAs classification framework. Once a TPA sends request for a user data, it will be classified through decision tree learning. Decision tree divides the population into smaller parts which are homogeneous in respect to a single feature or target variable. Decision trees are a very popular tool for predictive analytics as they are relatively easy to use, provide highly interpretable output and explicit visualization in a tree diagram [18]. Classification is based on the features that provide the most information to be used to assign the TPA to the accurate class.

5.2 The Phases

The three different phases of TPAs classification framework are described in details as follows.

5.2.1 Searching Features and Constructing a Model Phase

All TPAs which have more than 10 monthly active users are listed by Facebook in the search feature. Thus, we searched for listed applications to collect TPAs features and information by comparison of malicious and non-malicious applications on Facebook. Most of these features are available on TPAs description pages. We select application category, rating, requested permission set, external link to post ratio and website reputation score (WOT) as they often assist more in detecting malicious applications [6]. Table 1 depicts these predefined features by the values details.

The main focus of this research is on TPAs requests on users' profile data. Users' profile data are divided into three categories which are identifier (ID), quasi-identifiers (QI) and sensitive attribute (S). Identifiers indicate a user directly while an explicit sequence of quasi-identifiers can lead attackers by connecting their dataset to other

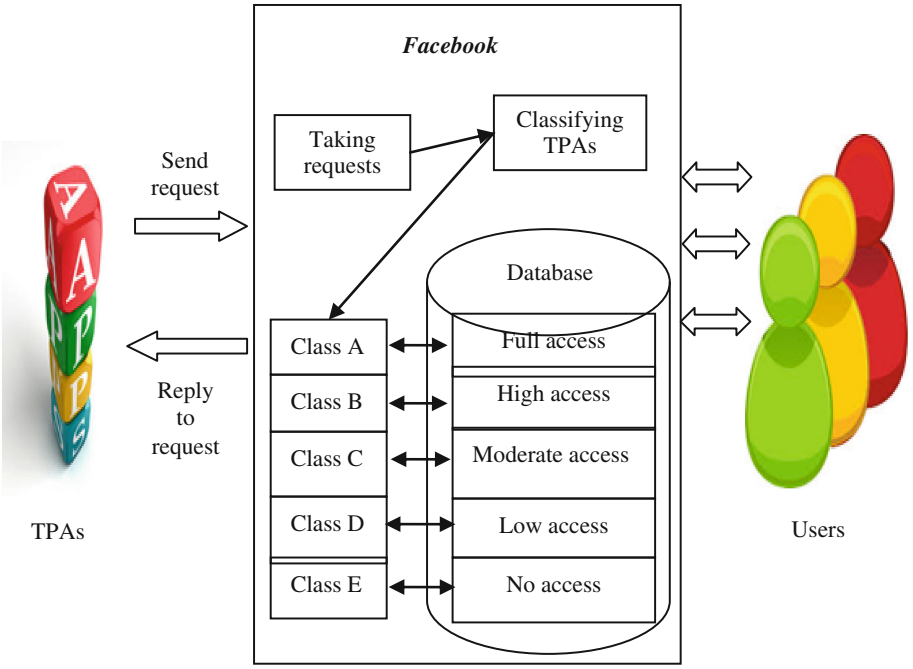

Fig. 2. Architecture of TPAs classification framework

published dataset to identify a user or ultimately gaining sensitive information. Sensitive data contains sensitive information which need to be hide from public.

Table 1. Application features and values

Features	Values
App category	Communication, Business, Fashion, Entertainment, Games, Finance, Health & Fitness, Food & Drink, Books, Education
Rating (1:5 stars)	5: excellent, 4: good, 3: fair, 2: poor, 1: very bad
Required permission set	ID,QI,S; ID,QI;ID,S; QI,S; ID;QI;S
External link to post ratio	Yes, No
WOT	Good, Caution, Bad, Unknown

To construct a model, first part of the proposed implemented algorithm fetched available information of predefined features automatically by crawling the generic application page and perceive the URL redirection behavior. The gathered information used for model construction which is the training dataset and defines a set of predetermined classes. The training dataset includes 362 TPAs instances with 5 features. The model depicts the classification rules in a tree structure. Sample of a training dataset is shown in Table 2 and the output of the classification rules (constructed model) is shown in Fig. 3 respectively.

Table 2. Sample of a training dataset

App category	Rating	Required permission set	External link to post ratio	WOT	Access	Class
Communication	4	ID,QI,S	No	Good	High	B
Communication	4	ID,QI	No	Good	High	B
Communication	2	QI	Yes	Bad	Low	D
Communication	2	QI	Yes	Unknown	Rejected	E
Business	4	ID,QI,S	No	Caution	Moderate	C
Business	4	QI,S	No	Good	High	B
Business	3	QI,S	No	Good	Moderate	C
Business	3	QI	Yes	Caution	Low	D
Fashion	2	QI	Yes	Bad	Rejected	E
Fashion	2	ID,QI	Yes	Unknown	Rejected	E
Entertainment	4	ID,QI,S	No	Caution	High	C
Entertainment	2	QI	Yes	Bad	Low	D
Games	4	QI,S	No	Caution	Moderate	C
Games	2	QI	Yes	Unknown	Rejected	E
Games	4	QI,S	No	Caution	Moderate	C

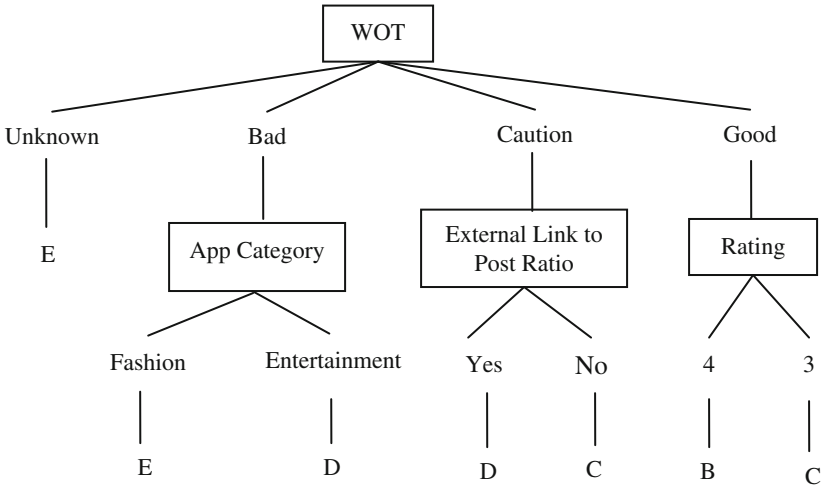

Fig. 3. Sample of a decision tree

A feature with highest information gain should be specified to be assigned as root in the tree. According to the training dataset described in Table 2, WOT provides the highest information among all variables. Thus, it will be assigned as the first node (root) in a tree. This variable have four values, namely, good, caution, bad and unknown. From unknown value, the values of dependent variables (access and class) can be determined directly which is class E and rejected access. For good, caution and bad values, other

variables need to be considered. For instance, when WOT is bad, app category is the next variable that provides highest information gain.

5.2.2 Classification and Prediction Phase

The proposed classification algorithm is optimized implementation of the C4.5 [19]. The classification algorithm classifies data based on the features in the training dataset and the known values (class labels) to predict unknown or missing values. It analyses and compares a TPA features with the training dataset. Then, a TPA will gain its class authority (class label) based on its infrastructure features and will grant access to requested users' data accordingly. Classification module in this algorithm encompasses two types of variables as follows.

1. Independent Variables. Application category, rating, required permission set, external link to post ratio and WOT are considered as independent variables which their values are illustrated in Table 1.
2. Dependent Variables. Access and class are defined as dependent variables. The values of dependent variables are as follows: Class A: Full Access, Class B: High Access, Class C: Moderate Access, Class D: Low Access, and Class E: Rejected (No Access).

5.2.3 Analysis Phase

The best results of classification analysis achieved when impurity or uncertainty in data is minimized as much as possible. This will occur due to the proper features selection with the maximum information gain to lead the classifier (classification algorithm) to assign all instances to the accurate class based on the classification rules which are defined as a tree branches. We consider three main evaluation criteria for our classifier analysis: generalization error, training error and confusion matrix that can be used to measure the classification accuracy. The definition and calculation of each criteria are described as follows.

1. Generalization error (GE) estimates the misclassification rate over the distribution D. Given a training set S with input attributes set A and a nominal target attribute y from an unknown fixed distribution D over the labeled instance space. Generalization error for the nominal attributes and the numeric attributes are calculated based on Eqs. (1) and (2), respectively.

$$\mathrm{GE}\left(\mathrm{I}(\mathrm{S}),\ \mathrm{D}\right) = \sum \mathrm{D}(\mathrm{x}, \mathrm{y}) \cdot \mathrm{L}(\mathrm{y}, \mathrm{I}(\mathrm{S})\,(\mathrm{x})) \tag{1}$$

$$\mathrm{GE}\left(\mathrm{I}(\mathrm{S}),\ \mathrm{D}\right) = \int \mathrm{D}(\mathrm{x}, \mathrm{y}) \cdot \mathrm{L}(\mathrm{y}, \mathrm{I}(\mathrm{S})\,(\mathrm{x})) \tag{2}$$

where L(y, I(S) (x)) is the zero-one loss function defined as:

$$\mathrm{L}(\mathrm{y}, \mathrm{I}(\mathrm{S})\,(\mathrm{x})) = \begin{cases} 0 & \text{if } \mathrm{y} = \mathrm{I}(\mathrm{S})\,(\mathrm{x}) \\ 1 & \text{if } \mathrm{y} \neq \mathrm{I}(\mathrm{S})\,(\mathrm{x}) \end{cases} \tag{3}$$

2. Training error (TE) depicts the number of correctly classified data by the classifier and can be calculated as Eq. (4).

$$TE\,(I(S),\ S) = \sum L(y, I(S)\,(x)) \tag{4}$$

3. Confusion matrix counts the test records that are predicted whether correctly or incorrectly by the classification model.

TP (true positive) and TN (true negative) denotes correctly classified instances as well as FP (false positive) and FN (false negative) indicates incorrectly classified instances. Subsequently, accuracy can be measured as the ratio of correctly classified instances to total number of instances (Fig. 4).

$$\text{Accuracy} = (TP + TN)/\,(TP + FN + FP + TN) \tag{5}$$

		Predicted	
		Positives	Negatives
Actual	Positives	TP	FN
	Negatives	FP	TN

Fig. 4. Confusion matrix

There is a correlation among these criteria. Once training error increases, generalization error will decrease and accuracy will rise accordingly.

6 Experimental Results

A dataset is generated randomly in order to give as an input to the implemented classification algorithm and the decision tree algorithm C4.5 in order to compare their results. From the generated data 150 instances were engaged in the experiment. After saving the data in Microsoft excel.CSV format and being processed in WEKA, the following described results were achieved. The percentage of correctly classified instances by the implemented classification algorithm which roots in C4.5 is around 79.23% while the percentage of incorrectly classified instances is about 29.7%. The C4.5 algorithm depicts around 67.93% correctly classified instances and about 38.1% incorrectly classified instances. Figure 5, delineates the comparison between C4.5 and the proposed classification algorithm.

The accuracy, true positive rate (TPR) and false positive rate (FPR) of the 150 test set from three pre-defined classes B, C and D are described in Table 3. According to the results, the classification accuracy of TPAs from class B is around 87.5%. The percentage of correctly classified instances in this class is 86.95 while 11.70% is classified incorrectly. TPAs classification accuracy from class C is 80% with 94.11% correctly classified and 38.46% incorrectly classified instances. TPAs classification from class D has about 86% accuracy with TPR 92% and FPR 20%.

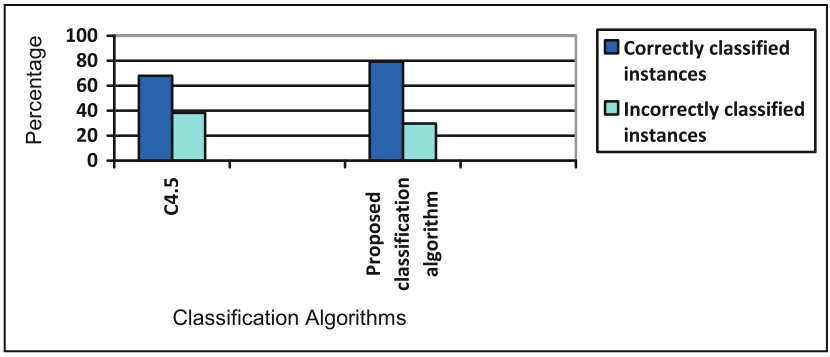

Fig. 5. Comparison of C4.5 and proposed classification algorithm

Table 3. Metrics values for TPAs from three class test

Class test	TP	TN	FP	FN	ACC %	TPR %	FPR %
B	20	15	2	3	87.5	86.95	11.70
C	32	16	10	2	80	94.11	38.46
D	23	20	5	2	86	92	20

The evaluation results of the proposed TPAs classification framework have shown that in the proposed framework training error is increased, hence generalization error is reduced which leads to improvement of classification accuracy. As accuracy is most significant metric to evaluate the performance of classification algorithm, improvement in accuracy results in the quality enhancement of the proposed framework.

7 Conclusions and Future Work

In this paper, the TPAs classification framework is proposed in order to provide mechanism in controlling TPAs access to the Facebook users' data. Classifying TPAs through the proposed TPAs classification framework based on their features can control unwanted dissemination of users' data to TPAs while still sustaining the functionality of the applications as they require users' data to deliver services. The proposed framework allows TPAs to utilize user data according to their class authority to preserve users' privacy and mitigate information leakage. The experimental results of the applied algorithm in the proposed framework have proved that this classification algorithm can provide acceptable accuracy in assigning each TPA to its accurate pre-defined class. Evaluating the proposed framework on Facebook if it could get authorization from administrator, can be followed up with further work on how to put this into practice and it is subject of our future work.

Acknowledgments. This study is funded by Research University Grant (RUG) from university of technology Malaysia (UTM), grant number Q.K130000.2538.09H32.

References

1. Barnes, S.B.: A privacy paradox: social networking in the United States. First Monday **11**(9), 4 September 2006
2. Arrington, M.: Facebook Launches Facebook Platform; they are the Anti-Myspace. http://techcrunch.com/2007/05/24/facebook-launches-facebook-platform-they-are-the-anti-myspace/
3. Felt, A., Evans, D.: Privacy protection for social networking APIs. In: Proceeding of Workshop on Web 2.0 Security and Privacy (W2SP 2008) (2008)
4. Bonneau, J., Anderson, J., Church, L.: Privacy suites: shared privacy for social networks. In: Proceedings of the 5th Symposium on Usable Privacy and Security, SOUPS 2009. ACM (2009)
5. Chaabane, A., Ding, Y., Dey, R., Kaafar, M.A., Ross, K.: A closer look at third-party OSN applications: are they leaking your personal information? In: Passive and Active Measurement Conference: (PAM 2014), Los Angeles, United States, pp. 235–246 (2014)
6. Rahman, Md.S., Huang, TK., Madhyastha, H.V., Faloutsos, M.: FRAppE: detecting malicious facebook applications. In: Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies, CoNEXT 2012, pp. 313–324 (2012)
7. Cheng, Y., Park, J., Sandhu, R.: Preserving user privacy from third-party applications in online social networks. In: The International World Wide Web Conference Committee (IW3C2). WWW 2013 Companion, Rio de Janeiro, Brazil (2013)
8. Symeonidis, I., Tsormpatzoudi, P., Preneel, B.: Collateral damage of Facebook Apps: an enhanced privacy scoring model. International Association for Cryptologic Research (2016)
9. Egele, M., Moser, A., Kruegel, Ch., Kirda, E.: PoX: Protecting Users from Malicious Facebook Applications. Comput. Commun. **35**(12), 1507–1515 (2012)
10. Beye, M., Jeckmans, A., Erkin, Z., Hartel, P., Lagendijk, R., Tang, Q.: Privacy in online social networks. In: Computational Social Networks: Security and Privacy, London, pp. 87–113 (2012)
11. Wang, N., Xu, H., Grossklags, J.: Third-party apps on facebook: privacy and the illusion of control. In: ACM CHIMIT 2011, Boston, MA, USA, 4 December 2011
12. Facebook. Facebook privacy settings and 3rd parties. http://www.ftc.gov/enforcement/cases-proceedings/092-3184/facebook-inc. Accessed 08 Feb 2015
13. McCarthy, C.: Understanding what Facebook apps really know (FAQ) (2010). http://cnet.co/1k85Fys
14. Facebook application center. https://www.facebook.com/appcenter. Accessed 13 Dec 2014
15. Jones, M., Hardt, D., Recordon, D.: The OAuth 2.0 Authorization Protocol: Bearer Tokens. draft-ietf-oauth-v2-bearer-12, 27 October 2011
16. Besmer, A., Lipford, H.R.: Users' (Mis) conceptions of social applications. In: Proceeding of Graphics Interface (GI 2010), Canadian Information Processing Society, pp. 63–70 (2010)
17. Huber, M., Mulazzani, M., Schrittwieser, S., Weippl, E.: AppInspect large-scale evaluation of social apps. In: Proceedings of the First ACM Conference on Online Social Networks (COSN 2013), pp. 143–154 (2013)
18. Patil, T.R., Sherekar, S.S.: Performance analysis of Naive Bayes and J48 classification algorithm for data classification. Int. J. Comput. Sci. Appl. **6**(2), 256–261 (2013)
19. Chambers, M., Dinsmore, T.H.W.: Predictive Analytics Techniques. Financial Times Press, New Jersey (2014)