Tracking Social Networks Events

Hugo Fonseca, Paulo Salvador and António Nogueira
DETI-University of Aveiro/Instituto de Telecomunicações
Email: {diogolopes, salvador, nogueira}@ua.pt

***Abstract*—Online social media represent a fundamental shift of how information is being produced, transferred and consumed. User generated content in the form of blog posts, comments, and tweets establishes a connection between information producers and consumers. Tracking the pulse of the social media outlets enables companies to gain feedback and insight in how to improve and market products better. For consumers, the abundance of information and opinions from diverse sources helps them make more informed decisions. However, the huge level of online interactions leads to permissive usage behaviours, opening the door for viruses, worms, trojan horses and other threats to install and easily spread, without being noticed. Being able to track users activities, organize the retrieved information in a comprehensive way and analyze it can be very useful for several management, engineering and security tasks. This paper proposes a framework to collect social network events and store them in a relational database for posterior analysis. A graphical user interface was developed to allow flexible access to stored information, according to the type of event, thus facilitating the analysis of users behaviours. From a privacy perspective, the proposed framework is not intrusive because it only gathers the actions timestamps and not their complete contents. By computing statistical models over the obtained data, it is possible to define ”normal or typical” usage profiles and detect possible deviations that can be indicative of a compromised user account.**

Keywords–Monitoring Framework; User Behaviour Modelling; Social Network; Compromised User Account.

I. INTRODUCTION

Web 2.0 paved the way for the boom of various online social communities, such as Facebook, LinkedIn or Twitter, which successfully facilitate information creation, sharing, and diffusion among users. Nowadays, information sharing and seeking are common user interaction scenarios on the Web. Information can appear in many different forms, like status updates, photos and videos, messages containing comments and/or reactions to certain events. According to the well-known traffic analytics website Alexa [1], the most important Online Social Networks (OSNs) are among the top websites in regard to traffic rankings, being a relevant phenomenon on the Internet.

Understanding the dynamics of these services and being able to track their users activities can be very important for several service design, management, engineering and security tasks. As part of their business model, social media companies make their Application Programming Interfaces (APIs) available to third parties. Besides, some of these companies make their databases on users and usage patterns available through their APIs: using simple software scripts, researchers can access the API to retrieve, store and manipulate digital traces left by the users of a service for further empirical analysis. However, it is not clear to what extent social media APIs can actually offer valid and reliable access points for collecting empirical data. So, APIs should certainly be used together with other methodologies to fully understand the activities and profiles of OSN users and the dynamics of these services.

Cybercriminals can steal information from each social networking profile and its associated posts and, then, tailor their attacks based on the interests and likes of each particular user. This is commonly known as ”social engineering” and makes security threats much more difficult to recognize. A complete analysis of the contents associated to a certain profile allows the identification of compromised accounts, that is, accounts that are under the control of cybercriminals without the knowledge of their licit owners. Illicit contents, containing, for example, links for *phishing* attacks, can be easily disseminated across the social network without raising any suspicion from targeted users.

Social network event collectors, which are able to collect and analyze information posted on a profile, are crucial for the identification of compromised accounts; by collecting the different types of contents posted on each user account, and organizing those events in a timely manner, it becomes easier to identify suspicious events/behaviours.

This paper presents a social network event collector that is able to periodically collect all information posted in a certain profile, as well as all profile related activities (like for example, creation or deletion of new connections, interactions with contents posted by other users, contents the user has shared on his profile). The proposed framework also includes a profile modeling module that can calculate and display several statistical information related to the usage behaviour of any profile. This module interacts with a database containing all collected information for the different user profiles: by computing the average occurrence of certain profile related events and comparing those values with the most recent profile usage statistics, it is possible to detect deviations from a ”normal” behavior in an efficient way. These deviations can be indicative of an illicit profile usage, enabling the detection of possible account hijacks. From a privacy perspective, this framework is not intrusive because it only gathers the actions timestamps and not their complete contents.

The remaining of this paper is organized as follows: Section II presents the most relevant related works; Section III describes the general architecture of the proposed system and discusses its most relevant performance issues; Section IV proposes a methodology to detect usage profile deviations; finally, Section V presents the main conclusions and discusses some possible directions for future work.

II. RELATED WORK

Online social networks have emerged in recent years and became a significant component of the digital life in our society. Several OSNs have millions of users [2][3][4], allowing them to share and find different types of multimedia contents and information. The popularity of these services provides an unique insight into the dynamics of social behaviors.

Mislove et al. [5] proposed a social network growth model based on the analysis of data collected from the Flickr social network. The main obstacle to their approach was the lack of information, which forced the authors to use an API that was made available by Flickr. Motoyama at al. [6] developed a system for searching and matching individuals in order to provide some insight into the dynamics and structures of OSNs. The efficiency of the proposed system was assessed by evaluating the overlap between different OSNs. These results were compared with the ones provided by state-of-the-art matching tools and the authors were able to conclude that their approach provided accurate matching results.

Kazienko et al. and Zanda et al. [7][8] proposed two social network activity recommendation systems. The first implemented a multidimensional social network based on the semantics of the social links between individuals, which enabled the creation of a social recommendation system to present recommendations to the different users. Personal weights are used to allow the system to become personalized and adaptive. The second reference proposed a recommendation system for mobile devices that accesses information from three different sources: mobile, sensor and Facebook. Facebook and mobile data are used to generate a social graph representing the relationships between a user and his friends, or subgroups of friends that can be used to build a recommender for the mobile device. Context data is then used for filtering purposes, allowing the system to present recommendations to the user. These are computed in the mobile device, thus guaranteeing the privacy of sensitive information. However, since the system analyzes the social connections of each user, its complexity depends exponentially on the number of connections.

Beach et al. [9] proposed a system that ties OSNs with mobile devices in an attempt to bind the identity with the corresponding physical location. The goal was also to propose a system on which complex context-aware applications can be built. The basic idea is that users store handles on their mobile devices pointing to their social networking profile that is stored on a remote site. Such profiles, which store information regarding their personalities and preferences as well as the different social network accounts (Facebook, Twitter, LinkedIn and more), are then exchanged between people sharing a common physical place. In this manner, the authors intended to establish a new social interaction paradigm. Several privacy and legal issues were raised by this approach, which also limited its deployment. Besides, the proposed system lacks the ability to create behavior models that could enrich it and turn it in a safer platform.

A framework for reducing the spread of threats between social network users was proposed by Tubi et al. [10]. It comprises a Distributed Network Intrusion Detection System for monitoring the propagation of threats and viruses over the monitored social network. This network was inferred from email addresses obtained from the logs of email servers from Ben-Gurion University. The authors were able to slow down and prevent the propagation of threats by cleaning the traffic from central users of the network. An algorithm for labeling nodes in a social network as honest or illicit was proposed by Danezis et al. [11]. The algorithm, named SybilInfer, uses a probabilistic model of honest OSNs and an inference engine for detecting regions of dishonest nodes. Simulated and real network topologies were used to assess that the algorithm is able to identify compromised nodes. Jin et al. [12] addressed Identity Clone Attacks, which consist of using fake identities for illicit purposes on OSNs. A feasible and effective framework focused on the detection of suspicious identities was proposed. A spectrum based attack detection framework based on the spectral space of underlying network topology was presented by Ying et al. [13]. The work focused on Random Link Attacks, which are filtered by using their spectral coordinate characteristics, which are mainly determined by the coordinates of the victims. The approach improved effectiveness and efficiency when compared to other topology detection approaches. Recently, Cai et al. [14] proposed a statistical model and some associated learning algorithms, named Latent Community model, for the detection of Sybil attacks (where an adversary creates multiple bogus identities to compromise the normal running of the targeted system) on OSNs. This model groups nodes into closely linked communities, which are then linked with the rest of the community. This way, authors could create communities for launching this type of attack, being able to accurately detect them. The results obtained allowed the authors to state that the proposed approach can be the best method for the automatic detection of Sybil attacks. However, the model efficiency should be improved in order to detect low-density attacks.

Perez et al. [15] proposed a dynamic behavioural framework for the identification of suspicious profiles in social networks. The approach is based on three main indicators, balance, energy and anomaly, which are all synthesized from daily user data. Balance refers to the visibility contained within a number of posted messages; energy accounts for the energy that is consumed by a profile for increasing its visibility; the third indicator indicates the anomaly score of an observed activity-visibility pair. The authors argue that suspicious users will have unusual visibility and activity pairs, which is then reflected on the anomaly score. The analysis spans throughout a time period where each indicator is computed and a score is associated to each analysed profile, indicating its level of suspicion. The proposed approach was applied to a set of 2000 Twitter profiles and the obtained results show that suspicious profiles have a more heterogeneous behaviour than normal ones. Moreover, the authors also stated that suspicious profiles present more extreme values of balance, are more likely to spend higher energy than normal profiles and are also likely to have an unusual activity-visibility pairing.

Stringhini et al. [16] analysed spam and spammers in OSNs by creating a large set of ”honey-profiles” and collecting data about spamming. Anomalous users behaviours were identified and a mechanism was proposed to automatically detect spammers. Shen et al. [17] proposed the SOcial network Aided Personalized and effective spam filter (SOAP), where each node connects to its social friends and forms a distributed overlay by using social links, being able to collect information and check spam in an autonomous way. Mahmood [18] identified several privacy leaks of Facebook and Twitter, which allow attackers to collect information for launching targeted attacks such as spam and phishing contents. Solutions for the identified security breaches were also proposed, although their efficiency has not been evaluated. The *SafeGo* tool [19] intends to bring Internet security features to social networking, by protecting users from malware threats that attempt to exploit the trust a user has with his/hers connections. This application uses the BitDefender antimalware and antiphishing engines for scanning Uniform Resource Locators (URLs) through an in-cloud approach based on a blacklist of untrusted URLs. However, unknown threats cannot be detected by this application.

SybilGuard [20] and SybilLimit [21] are two decentralized algorithms for the identification of Sybil attacks in social network topologies. A scalable and efficient solution, named SybilDefender, is able to detect sybil nodes and the underlying community.

III. System Description

The developed system has two main components: the core component, which is coded in Python and Hypertext Preprocessor (PHP), is responsible for estimating/calculating the profiles of the users interactions based on timestamps of events collected from social networks; the web interface component, which is developed in PHP and HyperText Markup Language (HTML) with Cascading Styles Sheets (CSS), allows users to interact and visualize their own profile metrics.

The system architecture is depicted in Figure 1. This distributed architecture dynamically supports several machines, which means that machines can be added or removed at any time. The relational database is used to store metadata corresponding to the different users of each social network, which is extracted by different social network crawling modules. A set of different statistics is then calculated by a Statistics Generator Module and stored on a specific database, the Statistics Database. The Alert Generator Module uses information stored in the Statistics Database to trigger alerts, saving also the data it generates into the same database. Besides these databases, the system also includes a Management Database that is exclusively accessed by the Manager Entity, which is the module responsible for the system coordination.

This framework includes a different relational database for each supported social network. This distributed approach is very important for performance, consistency, troubleshooting and backup issues. Besides, it will also allow system scalability, which is crucial for this kind of continuously evolving tools. Regarding the statistics and alerts database, only one was considered because the amount of data is very small when compared to data retrieved from the OSNs.

The system has a login possibility for each supported online social network. It does not have an autonomous login method, which means that a user is only allowed to login using a social network authentication method. In fact, it does not make sense to have an account on the system for a user that is not a social network user. Since a user can login from several social networks, that is, the system is based on third party authentication, a user has multiple login credentials that have to be associated to only one account. A specific database was created to store information about users identification.

For a particular user, the system starts by fetching information (events timestamps) from all social networks. After this module finishes its execution, the statistics generator module is launched to calculate personal statistics. It is possible to select the social network from which we want to retrieve/check statistics, no matter which social network was used to login. For example, a user can log into the system using Facebook and check his Twitter related statistics, or vice versa. Since the framework supports multiple social networks, the different user profile records should obviously be consistent.

Modules corresponding to a specific user, even if they are from different social networks, run at same time, on the same or on different machines. There are several services available that allow posting on social networks on behalf of users or collecting and generating users statistics, like for example *http://www.tweetstats.com*, *http://twittercounter.com*, *http://datasift.com* and *http://gnip.com*. Some are paid, others are free, can include more or less functionalities and are mostly used to monitor brands and provide insights into areas like business intelligence, marketing, finance, among others. However, our option was to develop this functionality from scratch in order to avoid being limited on the type data we can get or on the mathematical algorithms that we can apply to the retrieved data.

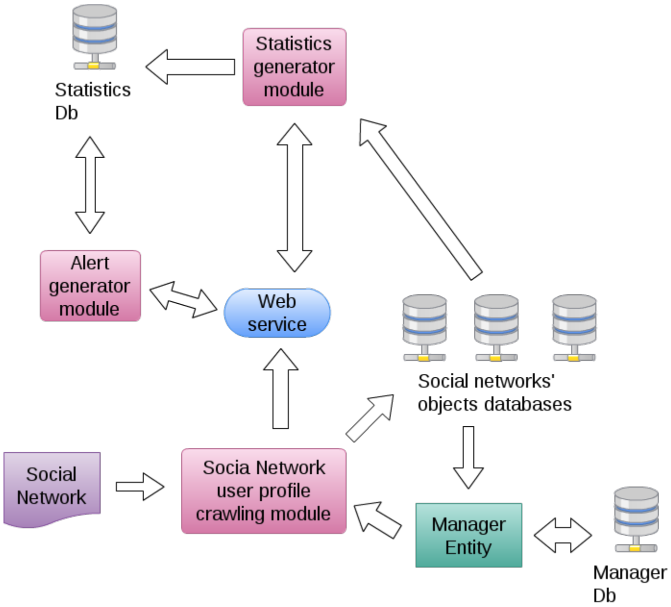

Statistics Db
Statistics generator module
Alert generator module
Web service
Social networks' objects databases
Social Network
Socia Network user profile crawling module
Manager Entity
Manager Db

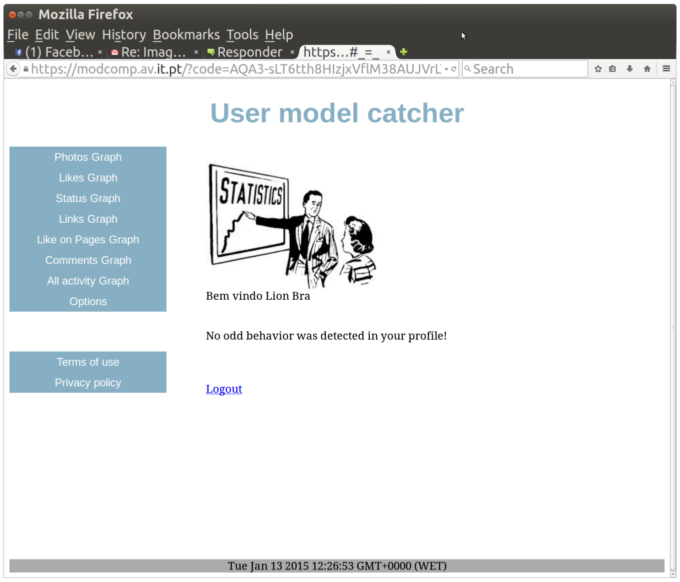
Figure 1. System Architecture.

Figure 2 shows the page that is displayed after a user logs into the system. If there are any alerts, a table showing its details will be presented. Nagivation links are displayed on the left side of the page.

Application users are able to see charts with their total and average amounts of social network activity per day, for a chosen number of days, besides being also able to see the activity of their friends, in some cases. This possibility does not imply any privacy issue because the friend that accepted the application is only allowed to see his current friends on the social network. Indeed, if a friend ceases his relationship, he is no longer able to inspect his ex-friend statistics, unless that ex-friend becomes a friend again on the social network. As illustrative examples of the statistics that can be displayed, Figures 3 and 4 show the number of comments in the last 30 days and the number of all interactions in the last 90 days, respectively.

The developed framework also has the ability to compare the activity of different users, determine which periods correspond to high or low activity and which are the most relevant activity types at each time period (comments, wall status, photos, posted links, likes, etc).

Figures 5 and 6 show some system management functions, namely the ability to add, remove and pause alerts modules, as well as the possibility to manage which users have access to which modules.


Mozilla Firefox
File Edit View History Bookmarks Tools Help
(1) Faceb... Re: Imag... Responder https...#_=_
https://modcomp.av.it.pt/?code=AQA3-sLT6tth8HIzjxVFlM38AUJVrL
Search
User model catcher
Photos Graph
Likes Graph
Status Graph
Links Graph
Like on Pages Graph
Comments Graph
All activity Graph
Options
Terms of use
Privacy policy
STATISTICS
Logout
Tue Jan 13 2015 12:26:53 GMT+0000 (WET)

Bem vindo Lion Bra

No odd behavior was detected in your profile!

Figure 2. Fist page after user login.

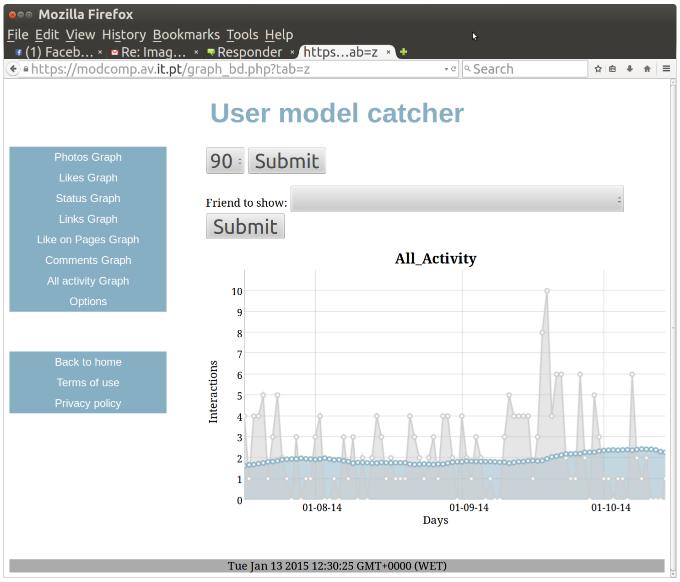


Figure 4. Results - All activities in the last 90 days.

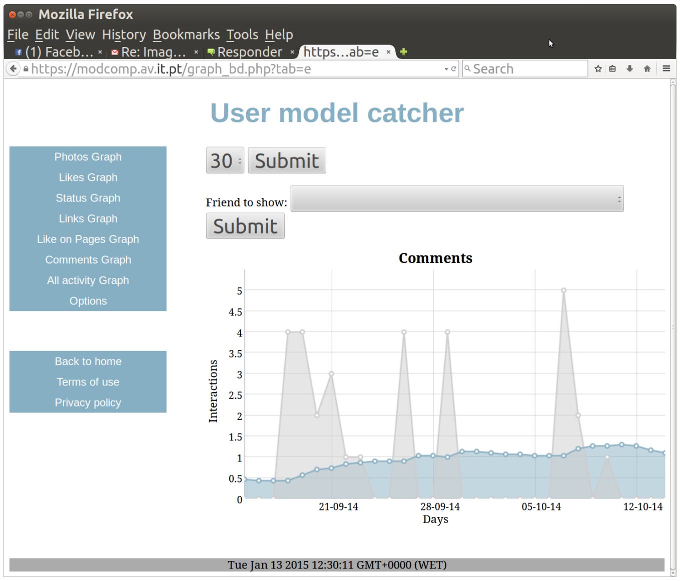
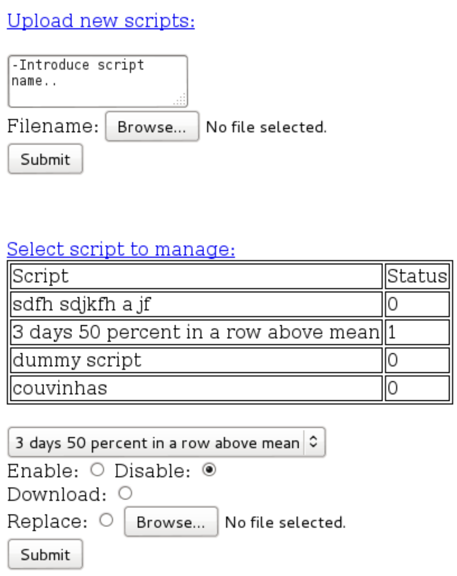


Figure 3. Results - Number of comments in the last 30 days.



Upload new scripts:

-Introduce script name..

Filename: Browse... No file selected.

Submit

Select script to manage:

Script	Status
sdfh sdjkfh a jf	0
3 days 50 percent in a row above mean	1
dummy script	0
couvinhas	0

3 days 50 percent in a row above mean

Enable: ○ Disable: ●

Download: ○

Replace: ○ Browse... No file selected.

Submit

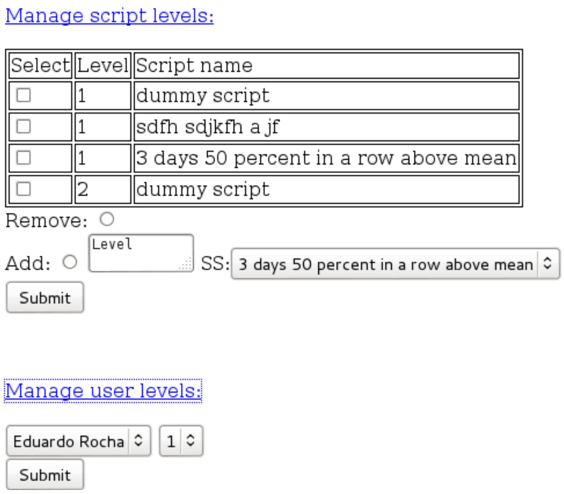
Figure 5. Uploading new management scripts.

The proposed system is based on the timestamps of the social networks events, it does not analyse the contents of user data (photos, text, videos, etc), so user privacy is guaranteed.

Finally, let us briefly discuss system efficiency and performance. The web component is very light and does not require any powerful end device at the user side. At the server side, it is also a light application because it is a PHP webpage with no heavy plugins. Obviously, the performance of the web component will decrease if the number of users accessing the system simultaneously increases significantly.

Core system operations are essentially composed by database accesses and mathematical calculations. Although the core system performance may slightly oscillate depending on the database size, it usually takes from one to two minutes to execute all the operations dedicated to each user.

The retrieval of social network contents is a very slow process: besides the high number of requests and connections that have to be established to retrieve the data, we need to take into account the activity level of social network servers that sometimes may increase download time or even completely prevent data retrieval. To circumvent this drawback, it is necessary to wait for a period of time between connections, thus increasing the time that is needed to complete the process. Another issue that has to be accounted for is the size of the crawled time window: a wider time window usually means more requests, increasing substantially the execution time.



Manage script levels:

Select	Level	Script name
□	1	dummy script
□	1	sdfh sdjkfh a jf
□	1	3 days 50 percent in a row above mean
□	2	dummy script

Remove: ○

Add: ○ Level SS: 3 days 50 percent in a row above mean

Submit

Manage user levels:

Eduardo Rocha 1

Submit

Figure 6. Managing existing scripts.

IV. Methodology to Detect Profile Usage Deviations

The Alert Generator Module triggers alerts whenever there is a significant difference in user behaviour. Alarms can be immediately sent to the social network profile of each user, if he signs into the application, or sent to his mailbox. For each type of action (comments, likes, etc.), it is necessary to define thresholds for the statistics values.

By defining a set of events $E_t^i = \{e_t^i, t = t_1, ..., t_2\}$ as the set of events of type $i \in \{\mathcal{S}, \mathcal{L}, \mathcal{C}, \mathcal{P}\}$, where each element denotes status updates, likes, comments, photos, respectively, during a time-window of analysis of width $t = t_2 - t_1$, we can compute the average of events of type i within the defined period of analysis as:

$$\overline{E_t^i} = \frac{\sum_{t=t_1}^{t_2} e_t^i}{T_{e_i}} \qquad (1)$$

in which T_{e_i} denotes the number of time units (years, months, days,...) considered for analysis within the width of the analysis window defined by $t = t_2 - t_1$. By defining a threshold δ

$$\delta = \alpha.\overline{E_t^i}, \alpha \in \mathbb{R}^+ \qquad (2)$$

as a function of the previously defined average, an alarm will be triggered if the average number of events, computed according to (1), within the time period defined by $t' = t'_2 - t'_1 : t'_2 \neq t_2, t'_1 \neq t_1$, exceeds the value computed as:

$$E_{t'}^i > \overline{E_t^i} + \delta \qquad (3)$$

The threshold can be dynamically redefined in order to decrease/avoid false positives. Threshold δ is defined by the user and, initially, three levels can be considered: aggressive, normal and permissive. In the first level, the tool should be able to detect slight deviations from normal usage profiles; in the normal level, only significant deviations should be detected and in the third level the detection approach is quite permissive, detecting only very important deviation on the usage profiles. Of course, the number of false positives can be quite high in the aggressive mode, so the user can redefine the threshold whenever the detection accuracy falls bellow an acceptable level.

V. Conclusions and Future Work

Online social networks are among the most popular Web sites. Social networking will certainly play an important role in future personal and commercial online interaction, as well as the location and organization of information and knowledge. Examples include browser plug-ins to discover information viewed by friends and social network based, cooperative Web search tools. Social network event collectors can be used to build users behaviour and activity profiles, which are very useful to several management and engineering tasks: evaluate the performance of existing social network platforms, conduct social studies, develop viral marketing strategies, design new content distribution systems, detect anomalous behaviors (such as bots or compromised accounts) and trigger the corresponding alerts, etc. In this paper, we proposed a framework that is able to periodically collect metadata corresponding to all user profile related activities. By retrieving only metadata, no privacy issues are risen by this methodology, because the conducted analysis is only based on timestamps and no activity details are searched and stored in the database. Several online social networks (Facebook, Twitter and Google+) are already supported and the system architecture was designed to be fully compatible with additional social network platforms.

Regarding future developments, we believe that the chat feature of social networks would be interesting to endorse in order to better characterize relationships between users, although this corresponds to a critical privacy issue. We are also planning to incorporate user models based on multi-scale analysis: using concepts that exploit different scales of analysis [22], it is possible to identify the different frequency components that are created by a social network user profile and build mathematical models that can accurately describe the different user interactions.

Acknowledgements

This research is supported by Fundação para a Ciência e Tecnologia through research projects PTDC/EEI-TEL/2016/2012, PEst-OE/EEI/LA0008/2013 and UID/EEA/50008/2013.

References

[1] Alexa traffic statistics for facebook. http://www.alexa.com/siteinfo/facebook.com/. [retrieved: May, 2015]

[2] Welcome to facebook – log in, sign up or learn more. https://www.facebook.com/. [retrieved: May, 2015]

[3] Google+: real life sharing, rethought for the web. https://plus.google.com/up/start/. [retrieved: May, 2015]

[4] Welcome to linkedin. http://www.linkedin.com/. [retrieved: May, 2015]

[5] A. Mislove, H. S. Koppula, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Growth of the flickr social network," in Proceedings of the First Workshop on Online Social Networks, ser. WOSN '08. New York, NY, USA: ACM, 2008, pp. 25–30. [Online]. Available: http://doi.acm.org/10.1145/1397735.1397742

[6] M. Motoyama and G. Varghese, "I seek you: Searching and matching individuals in social networks," in Proceedings of the Eleventh International Workshop on Web Information and Data Management, ser. WIDM '09. New York, NY, USA: ACM, 2009, pp. 67–75. [Online]. Available: http://doi.acm.org/10.1145/1651587.1651604

[7] P. Kazienko, K. Musial, and T. Kajdanowicz, "Multidimensional social network in the social recommender system," Trans. Sys. Man Cyber. Part A, vol. 41, no. 4, Jul. 2011, pp. 746–759. [Online]. Available: http://dx.doi.org/10.1109/TSMCA.2011.2132707

[8] A. Zanda, E. Menasalvas, and S. Eibe, “A social network activity recommender system for ubiquitous devices,” in Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on, Nov 2011, pp. 493–497.

[9] A. A. Beach et al., “Whozthat? evolving an ecosystem for context-aware mobile social networks,” Netwrk. Mag. of Global Internetwkg., vol. 22, no. 4, Jul. 2008, pp. 50–55. [Online]. Available: http://dx.doi.org/10.1109/MNET.2008.4579771

[10] M. Tubi, R. Puzis, and Y. Elovici, “Deployment of dnids in social networks,” in Intelligence and Security Informatics, 2007 IEEE, May 2007, pp. 59–65.

[11] G. Danezis and P. Mittal, “Sybilinfer: Detecting sybil nodes using social networks,” Tech. Rep. MSR-TR-2009-6, January 2009. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=78896

[12] L. Jin, H. Takabi, and J. B. Joshi, “Towards active detection of identity clone attacks on online social networks,” in Proceedings of the First ACM Conference on Data and Application Security and Privacy, ser. CODASPY ’11. New York, NY, USA: ACM, 2011, pp. 27–38. [Online]. Available: http://doi.acm.org/10.1145/1943513.1943520

[13] X. Ying, X. Wu, and D. Barbara, “Spectrum based fraud detection in social networks,” in Data Engineering (ICDE), 2011 IEEE 27th International Conference on, April 2011, pp. 912–923.

[14] Z. Cai and C. Jermaine, “The latent community model for detecting Sybils in social networks,” in Proceedings of the 19th Annual Network & Distributed System Security Symposium, Feb. 2012.

[15] C. Perez, M. Lemercier, and B. Birregah, “A dynamic approach to detecting suspicious profiles on social platforms,” in Communications Workshops (ICC), 2013 IEEE International Conference on, 2013, pp. 174–178.

[16] G. Stringhini, C. Kruegel, and G. Vigna, “Detecting spammers on social networks,” in Proceedings of the 26th Annual Computer Security Applications Conference, ser. ACSAC ’10. New York, NY, USA: ACM, 2010, pp. 1–9.

[17] H. Shen and Z. Li, “Leveraging social networks for effective spam filtering,” IEEE Transactions on Computers, vol. 99, no. PrePrints, 2013, p. 1.

[18] S. Mahmood, “New privacy threats for facebook and twitter users,” in P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2012 Seventh International Conference on, Nov 2012, pp. 164–169.

[19] Bitdefender safego - unfriend your phishy links! http://www.bitdefender.com/solutions/bitdefender-safego.html. [retrieved: May, 2015]

[20] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, “Sybilguard: Defending against sybil attacks via social networks,” SIGCOMM Comput. Commun. Rev., vol. 36, no. 4, Aug. 2006, pp. 267–278. [Online]. Available: http://doi.acm.org/10.1145/1151659.1159945

[21] H. Yu, P. Gibbons, M. Kaminsky, and F. Xiao, “Sybillimit: A near-optimal social network defense against sybil attacks,” in Security and Privacy, 2008. SP 2008. IEEE Symposium on, May 2008, pp. 3–17.

[22] E. Rocha, P. Salvador, and A. Nogueira, “Detection of illicit network activities based on multivariate gaussian fitting of multi-scale traffic characteristics,” in Communications (ICC), 2011 IEEE International Conference on, June 2011, pp. 1–6.