International Journal of Advanced Technology and Innovative Research

www.ijatir.org

ISSN 2348–2370
Vol.08,Issue.09,
August-2016,
Pages:1857-1861

Discovering Fraud Apps in Facebook using Frappe in Mobile Apps

M. GEETADEVI[1], T. SIVA RAMA KRISHNA[2]

[1]PG Scholar, Dept of CSE, BVCITS, JNTUK, AP, India, E-mail:gitadevi524@gmail.com.
[2]Assistant Professor, Dept of CSE, BVCITS, JNTUK, AP, India, E-mail: sivarkt@gmail.com.

Abstract: Now a day's third party apps are major for the extensive usage and easy to operate, within a day over 1 million installations. For these reasons attackers came to known that the potentiality of using apps for streaming malware and spam. In our dataset over 13% of apps are to considered as malicious. Social research community has concentrated on detecting malicious past campaigns. If suppose we download a facebook from store that we cannot determine if it is malicious or not. Our proposed key FRAppE- Facebook Rigorous Application Evaluator this is a new tool used to fine malicious apps on facebook. To develop this new tool, we gathering the information from different users over 2.2 million user on facebook over 111k facebook apps are using. In first step we identifying a set of features to identify the malicious app or not often share names with other apps. Secondly, we concentrate on distinguished features, by using our tool FRAppE it can easily detect malicious app 95% accuracy. By identifying all mechanism that we explore the ecosystem of malicious facebook apps. In our dataset we conclude and support many apps over 1584 apps are out from viral proportion and 3723 other app through postings. Our tool FRAppE will warn before installing the app, by FRAppE is use to creating security of app assessment and ranking.

Keywords: FRAppE, Evaluator, Community, Rigorous Application.

I. INTRODUCTION

Online Social Networks (OSN's) enable and inspire third-party applications (apps) to enhance the user experience on these platforms like FaceBook, Twitter. Interesting or entertaining ways of communicating among on-line friends and diverse activities such as playing games or listening to songs are examples of such enhancements. For example, Facebook provides developers an API that facilitates app integration into the Facebook user experience. There are 500K apps available on Facebook, and on average, 20M apps are installed every day. Further-more, many apps have acquired and maintain a really large user database. It has been observed that FarmVille and CityVille apps have 26.5M and 42.8M users to date. Recently, hackers and malicious users have started taking advantage of the popularity of this third-party apps platform and deploying malicious applications. Malicious apps can provide a lucrative business for hackers, given the status of OSN's, with Facebook leading the way with 900M active users. There are many ways that hackers can benefit from a malicious app:

- The app can reach large number of users and their friends to spread spam.
- The app can obtain users personal information such as e-mail address, home town, and gender, and
- The app can "reproduce" by making other malicious apps popular.

In other words, there is motive and opportunity, and as a result, there are many malicious apps spreading on Facebook every day. Despite the above worries, today a user has very limited information at the time of installing an app on his Facebook profile. In other words, the problem is the following: Given an app's identity number (the unique identifier assigned to the app by Facebook), can we detect if the app is malicious? Currently, there is no commercial service, publicly available information, or research-based tool to advise a user about the risks of an app. Malicious apps are widespread and they easily spread, as an infected user jeopardizes the safety of all its friends. So far, the researches has been done regarding spam and malware on Facebook which has focused on detecting malicious posts and social spam campaigns. At the same time, in a seemingly backwards step, Facebook has dismantled its app rating functionality. A recent study has shown how app authorizations correlate to privacy risks of Facebook apps. Finally, there are some community based feedbacks driven efforts to rank applications, such as WhatApp?; though these could be very powerful in the future, so far they have received little acceptance. The Fig.1 shows how the social malware is rampant on Facebook.

Fig.1.

In the Internet era, multimedia content is massively produced and distributed. In order to efficiently locate content in a large-scale database, content-based search techniques have been developed. They are used by content based information retrieva (CBIR) systems to complement conventional keyword-based techniques in applications such as near-duplicate detection, automatic annotation, recommendation, etc. In such a typical scenario, a user could provide a retrieval system with a set of criteria or examples as a query; the system returns relevant information from the database as an answer. Recently, with the emergence of new applications, an issue with content-based search has arisen sometimes the query or the database contains privacy-sensitive information. In a networked environment, the roles of the database owner, the database user, and the database service provider can be taken by different parties, who do not necessarily trust each other. A privacy issue arises when an untrusted party wants to access the private information of another party. In that case, measures should be taken to protect the corresponding information. Users are today forced to trust the service providers for the use of their profiles. Although CBIR systems have not been widely deployed yet, similar threats exist. Recently, the one-way privacy model for CBIR was investigated. The one-way privacy setting assumes that only the user wants to over the past decade, online social media (OSM) has stamped its authority as one of the largest information propagators on the Internet.

OSN services have deled all regional, cultural, and language boundaries, and provided every Internet user on the planet with an equal opportunity to speak, and be heard. Nearly 25% of the world's population uses at least one social media service today. 1 People across the globe actively use social media platforms like Twitter and Facebook for spreading information, or learning about real world events these days. A recent study revealed that social media activity increases up to 200 times during major events like elections, sports, or natural calamities [Szell et al. 2014]. This swollen activity contains a lot of information about the events, but is also prone to severe abuse like spam, misinformation, and rumour propagation, and has thus drawn great attention from the computer science research community. Since this stream of information is generated and consumed in real time, and by common users, it is hard to extract useful and actionable content, and later out unwanted feed. Twitter, in particular, has been widely studied by researchers during real-world events [Becker et al. 2011; Hu et al. 2012; Kwak et al. 2010; Sakaki et al. 2010; Weng and Lee 2011]. However, few studies have looked at the content spread on social media platforms other than Twitter to study real-world events [Chen and Roy 2009; Hille and Bakker 2013; Osborne et al. 2012]. Surprisingly, there has been little work on studying content on Facebook during real world events [Westling 2007], which is five times bigger than Twitter in terms of the number of monthly active users. Range of research attempts which would help to explore malicious content spread on Facebook during events. In particular, we look at three distinct areas, viz.

- The Facebook social graph,
- Attack and detection techniques with respect to malicious content on Facebook, and
- Analysis of events using online social media data. Then, we look at the various limitations that Facebook poses, which makes event analysis, and detection of malicious content on this network a hard problem. Towards the end, we discuss the implications and research gaps in identifying and analysing malicious user generated content on Facebook during events.

II. PROBLEM STATEMENT

Currently, malicious apps often do not include a category, company, or description in their app summary. To detect the malicious facebook applications which may affects to user's private information on his/her profile. As we see user did not get much information about application expect name of that application while installing as a result no security available on Facebook.

III. RELATED WORK

A. Detecting and Characterizing Social Spam Campaigns Authors

Hongyu Gao, Jun Hu, Christo Wilson,Zhichun Li, Yan Chen, Ben Y. Zhao. Description: Authors presented a primary study to calculate and analyze spam campaigns launched on online social networks. They calculated a huge anonymized dataset of asynchronous "wall" messages in between Facebook users. System detected generally 200,000 malicious wall posts with embedded URLs, originating from more than 57,000 user accounts. Authors found that more than 70% of all malicious wall posts advertise phishing sites. To study the distinctiveness of malicious accounts, and see that more than 97% are compromised accounts, rather than "fake" accounts formed solely for the principle of spamming. Finally, when adjusted to the local time of the sender, spamming dominates actual wall post in the early morning hours when users are normally asleep.

B.Is this App Safe? A Large Scale Study on Application Permissions and Risk Signals Authors

Pern Hui Chia, Yusuke Yamamoto, N.Asokan Description: Third-party applications capture the attractiveness of web and platforms providing mobile application. Many of these platforms accept a decentralized control strategy, relying on explicit user consent for yielding permissions that the apps demand. Users have to rely principally on community ratings as the signals to classify the potentially unsafe and inappropriate apps even though community ratings classically reflect opinions regarding supposed functionality or performance rather than concerning risks. To study the advantages of user-consent permission systems through a large data collection of Facebook apps, Chrome extensions and Android apps. The study confirms that the current forms of community ratings used in app markets today are not reliable for indicating privacy risks an app creates. It is found with some evidences, indicating attempts to mislead or entice

users for granting permissions: free applications and applications with mature content request; "look alike" applications which have similar names as that of popular applications also request more permissions than is typical. Authors find that across all three platforms popular applications request more permissions than average.

C. Social Applications: Exploring a More Secure Framework Authors

Andrew Besmer, Heather Richter Lipford,Mohamed Shehab, Gorrell Cheek Description: OSNs such as Orkut, Facebook and others have grown-up rapidly, with hundreds to millions of active users. A new feature provided on several sites is social applications and services written by third party developers that supply additional functionality linked to a user's profile. However, present application platforms put users at risk by permitting the discovery of huge amounts of personal data and information to these applications and their developers. This paper generally abstracts main view and defines the current access control model gave to these applications, and builds on it to generate a more secure framework.

IV. MALICIOUS CONTENT ON FACEBOOK

The popularity and reach of Facebook has also attracted a lot of spam, phishing, malware, and other types of malicious activity. Attackers lure victims into clicking on malicious links pointing to external sources, and in literate their network. These links can be spread either through personal messages (chats), or through wall posts. To achieve maximum visibility, attackers prefer to post links publicly. Typically, an attacker initiates the attack by posting memes with attention grabbing previews, which prompt users to like, share, or comment on them in order to view them. The actions of liking, commenting or sharing spread these memes into the victim's network. Once the meme is spread, the victim is redirected to a malicious website, which can further infect her computer, or friends network through phishing, malware, or spyware. This phishing page asks the victim to share this video with their friends in order to view it. However, once the victim shares this video, the page redirects to a random advertisement page. The video corresponding to the preview / thumbnail shown in the post does not actually exist. Multiple other sources have cited such examples of scams and malicious posts on Facebook in the past few years. 11, 12 In addition to phishing scams, other malicious activity on Facebook includes unsolicited mass mentions, photo tagging, post tagging, private / chat messages etc. Intuitively, a user is more likely to respond to a message or post from a Facebook friend than from a stranger, thus making this social spam a more effective distribution mechanism than traditional email. This increased susceptibility to such kind of spam has prompted researchers to study, and combat social spam and other malicious activity on Facebook. We now look at the various attack and detection techniques that have been used in the past to identify and spread malicious content on Facebook respectively.

A. Attack Techniques

In order to identify and contain malicious posts on Facebook, or any OSM, it is essential to explore and understand the techniques that are, or can potentially be deployed by attackers to spread such content. To this end, Patsakis et al. [Patsakis et al. 2009] described how Facebook can be exploited and converted into an attack platform, in order to gain some sensitive data, which can complete a perfect attacking pro le against a user. Authors created a Facebook application for demonstration purposes that on the surface was a simple application, but on the background it collected useful data. This app executed malicious code on the victim's browser, and collected the IP address of the user-victim, the browser version, the OS platform and whether some specific ports are open or closed. This data was then transmitted to the authors over email. Authors also pointed out that their app was indexed on the main list of Facebook applications, despite the fact that the description of app clearly stated that it was generating malicious transaction, and had been created for penetration testing purposes. Huber et al. presented a friend-in-themiddle attack through hijacking session cookies. Authors explained how it was possible to impersonate the victim using this technique, and interact with the network without proper authorization.

However, this technique was proposed in 2011, when using HTTPS to connect to the website was optional. 13 Post 2013, all communication on Facebook uses encryption (HTTPS) by default, which means that such attacks are no more possible. Fan et al. [Fan and Yeung 2010] proposed a virus model based on the application network of Facebook. Authors also modelled the virus propagation with an email virus model and compared the behaviours of virus spreading in Facebook and email network. Their findings revealed that while Facebook provides a platform for application developers, it also provides the same chance for virus spreading. In fact, the virus was found to spread faster on the Facebook network if users spend more time on it. The result of their simulation showed that, even though a malicious Facebook application attracts only a few users in the beginning, it can still spread rapidly. That is because users may trust their friends of Facebook and install the malicious application. It is important to understand that in addition to the techniques described above, a large proportion of attacks on Facebook, and even other social networking platforms, make use of social engineering. This is evident since it is hard to initiate the spread of a malicious piece of content on a network without any human involvement. Attackers lure victims into using malicious apps, clicking malicious links, and sharing pieces of content, and in some cases, even pretend to provide various kinds of benefits in return. Since these attacks are well-crafted in most cases, it becomes hard for a legitimate user to be able to comprehend the results of her actions. We now look at the various techniques that have been proposed to detect malicious content on the Facebook social network.

B. Detection Techniques

Facebook has its own immune system to safeguard its users from unwanted, malicious content [Stein et al. 2011]. Researchers at Facebook built and deployed a coherent, scalable, and extensible real time system to protect their users and the social graph. This system performs real time checks and classifications on every read and write.

V. THE PROPOSED FRAMEWORK

In this work, we develop FRAppE, a suite of efficient classification techniques for identifying whether an app is malicious or not. To build FRAppE, we use data from My Page Keeper. To build FRAppE, we use data from My Page Keeper, a security app in Facebook that monitors the Facebook profiles of 2.2 million users. We analyse 111K apps that made 91 million posts over nine months. This is arguably the first comprehensive study focusing on malicious Facebook apps that focuses on quantifying, profiling, and understanding malicious apps, and synthesizes this information into an effective detection approach. We have introduced two features i.e. classifiers to detect the malicious apps FRAppE Lite and FRAppE . In first classifier it detect the initial level detection e.g. apps identity number , name and source etc. and in second level detection the actual detection of malicious app has been done.

A. Advantageous

- Facebook Rigorous Application Evaluator is arguably is the tool to detect malicious apps.
- It provides security to users profiles from malicious apps.

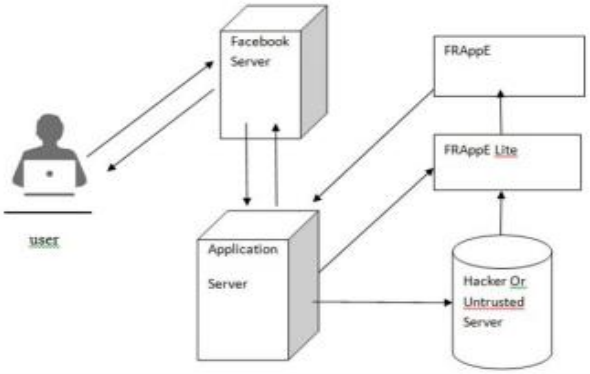


Fig.2. System Architecture

Feature extraction component. The extracted feature vectors are capable of characterizing the underlying content. They first undergo an orthogonal transform and dimension reduction. Only significant features are preserved. The elements of a feature vector are divided into n groups .A robust hash value hi ($i = 0,1,\cdots,n-1$) is computed from the ith group. We call it asub-hash value. The above step creates a new coordinate system, with each coordinate represented by a sub- hash value. Finally, a multimedia object in the database is indexed by the overall hash value $H = h0||h1||\cdots||hn-1$, i.e., the concatenation of sub-hash values. Each sub-hash value is associated with an inverted index list (also called a hash bucket). The list contains the IDs (identification information) of multimedia objects corresponding to the sub-hash value. The size of a sub-hash value l depends on the significance of its corresponding feature elements

VI. CONCLUSION

This Application performs about all the fake users who were existed in FRAppE. Here in Facebook it is a convenient process to Fake users for sending Messages and Posts on Facebook. However, a little is understood about this project of blocking users and how they unblock the users. In this process, large amount of Fake Users are involved. Fake users differ significantly to all other users with respect to several process. For example, Fake users are much more likely to send messages, post pictures with other users, So we develop FRAppE, a tool for "Detecting Malicious Facebook Users"between User and Admin. So that all the fake users can be de-activated and they can't login with their account.

VII. FUTURE WORK

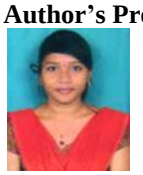
Already Facebook Application is Existed in real time, but in this project we have enhanced with more reliable in detecting.Implement this project in Facebook for Real time.While the user is blocked, the Alert Message should exist on Email, So that user knows that he/she was Blocked

VIII. REFERENCES

[1]C. Pring, "100 social media statistics for 2012," 2012 [Online].
[2]Facebook, Palo Alto, CA,USA, "Facebook Opengraph API," [Online].
[3]"Wiki: Facebook platform," 2014 [Online]. Available: http://en. wikipedia.org/wiki/Facebook_Platform
[4]"Pr0file stalker: Rogue Facebook application," 2012 [Online].
[5]"Which cartoon character are you—Facebook survey scam," 2012 [Online].
[6]G. Cluley, "The Pink Facebook rogue application and survey scam," 2012 [Online].
[7]D. Goldman, "Facebook tops 900 million users," 2012 [Online].
[8]HackTrix, "Stay away from malicious Facebook apps," 2013 [Online].
[9]M. S. Rahman, T.-K. Huang, H. V. Madhyastha, and M. Faloutsos, "Efficient and scalable socware detection in online social networks," in Proc. USENIX Security, 2012, p. 32.
[10]H. Gao et al., "Detecting and characterizing social spam campaigns," in Proc. IMC, 2010, pp. 35–47.
[11]H. Gao, Y. Chen, K. Lee, D. Palsetia, and A. Choudhary, "Towards online spam filtering in social networks," in Proc. NDSS, 2012.
[12]"WhatApp? (beta)—A Stanford Center for Internet and Society Website with support from the Rose Foundation," [Online].
[13]"MyPageKeeper," [Online]. Available: https://www .facebook.com/apps/application.php?id=167087893342260
[14]Facebook, Palo Alto, CA, USA, "Application authentication flow using OAuth 2.0," [Online].

[15]"11 million bulk email addresses for sale—Sale price $90," [Online].
[16]"bit.ly API," 2012 [Online].
[17]Facebook, Palo Alto, CA, USA, "Permissions reference," [Online].
[18]Facebook, Palo Alto, CA, USA, "Facebook developers," [Online].
[19]"Web-of-Trust," [Online]. Available: http://www. Mywot .com/
[20]C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," Trans. Intell. Syst. Technol., vol. 2, no. 3, 2011, Art. no. 27.
[21]J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: Learning to detect malicious Web sites from suspicious URLs," in Proc. KDD, 2009, pp. 1245–1254.
[22]A. Le, A. Markopoulou, and M. Faloutsos, "PhishDef: URL names say it all," in Proc. IEEE INFOCOM, 2011, pp. 191–195.
[23]Facebook, Palo Alto, CA, USA, "Facebook platform policies," [On-line].

Author's Profile:



Mr M. Geeta Devi is a student of Bonam Venkata Chalamaiah Institute of Technology and Science (BVCITS), Amalapuram. Presently he is pursuing M.Tech [Computer Science and Engineering] from this college and he also completed his B.Tech from BVCITS, affiliated to JNTU, Kakinada.

Mr T. Siva Rama Krishna, working as Assistant Professor, in the Department of Computer Science & Engineering from Bonam Venkata Chalamaiah Institute of Technology & Science (BVCITS), Amalapuram. He completed his M.Tech degree from JNTU Kakinada and he has a total teaching experience of 7 years. His area of Interest includes Computer Networks, Mobile Computing and other advances in Computer Applications.