IOSR Journal of Engineering (IOSRJEN) www.iosrjen.org
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 09, Issue 1 (January. 2019), ||V (II) || PP 44-50

A Tool for Classification of Malicious App

CH. Priyanka Raj[1], A.Krishan Mohan[2]
[1]M.Tech (IT), Dept. of Computer Science and Engineering,
[2] Professor, Dept. of Computer Science and Engineering, Jawaharlalnehru Technological University, Kakinada, Andhra Pradesh, India.
Corresponding Author: CH. Priyanka Raj

Abstract: The utilization of online social networking sites has turned into a necessary portion of our lives. It causes us to speak with our dear and separated ones. We can likewise utilize these sites for different media sharing purposes, for example, music, recordings, and so on. Additionally, nowadays these sites are increasing substantially more notoriety because of the outsider applications that exist on these stages. Be that as it may, interlopers have understood the capability of these applications and utilize it as a medium to spam users. In a great deal of cases, as indicated by a review done these applications are malicious. A spammer can profit by these applications in different ways like, can achieve countless, can acquire user's personal information, and furthermore with the assistance of a solitary user, he can spam much different users as well. As the examination goes on, look into networks have concentrated on distinguishing malicious URLs and online social battles which are fake or spam. Here we build up an application, SecureU application, we help distinguish malicious application, fake or spam messages, shroud pictures and posts which are unseemly and it will likewise assist us with giving constant warnings.

Keywords: Facebook Apps, Malicious Apps, Profiling Apps, Online Social Networks.

Date of Submission: 02-01-2019 Date of acceptance: 17-01-2019

I. INTRODUCTION

In the Internet time, interactive media content is greatly created and disseminated. With the end goal to productively find content in an expansive scale database, content-based pursuit strategies have been created. They are utilized by substance based information recovery (CBIR) [1] frameworks to supplement traditional catchphrase based procedures in applications, for example, close copy discovery, programmed explanation, suggestion, and so on. In such a run of the mill situation, a user could give a recovery framework an arrangement of criteria or precedents as an inquiry; the framework returns applicable information from the database as an answer. As of late, with the development of new applications, an issue with substance based pursuit has emerged here and there the question or the database contains security touchy information [3][2]. In an organized domain, the jobs of the database proprietor, the database user, and the database specialist organization can be taken by various gatherings, who don't really confide in one another. A protection issue emerges when an untrusted party needs to get to the private information of another gathering. All things considered, measures ought to be taken to secure the comparing information. The primary test is that the hunt must be performed without uncovering the first question or the database. This inspires the requirement for protection saving CBIR (PCBIR) frameworks. Security brought early consideration up in biometric frameworks, where the question and the database contain biometric identifiers. Biometric frameworks infrequently keep information free, dreading burglaries of such profoundly significant information. Also, a user is hesitant in sending his biometric layout free. Customarily, biometric frameworks [4] depend on cryptographic natives to ensure the database of layouts. In the mixed media space, security issues as of late developed in substance suggestion. With suggestion frameworks, users are normally profiled. Profiles are sent to specialist organizations, which send back personalized substance.

II. RELATED WORK

Distinguishing and Characterizing Social Spam Campaigns Authors: HongyuGao, Jun Hu, Christo Wilson, Zhichun Li, Yan Chen, Ben Y. Zhao. Depiction: Authors displayed an essential report to ascertain and break down spam crusades propelled on online social systems. They ascertained a tremendous anonymized dataset of nonconcurrent "divider" messages in the middle of Facebook users. Framework distinguished for the most part 200,000 malicious divider posts with implanted URLs, beginning from in excess of 57,000 user accounts. Creators found that additional than 70% of all malicious divider posts promote phishing sites. To ponder the uniqueness of malicious records, and see that over 97% are endangered records, as opposed to "fake" accounts shaped exclusively for the guideline of spamming. At last, when acclimated to the nearby time of the sender, spamming commands real divider post in the early morning hours when users are regularly sleeping. Is this App Safe? A Large Scale Study on Application Permissions and Risk Signals Authors: PernHui Chia, Yusuke Yamamoto, N. Asokan Description: Third-party applications catches the engaging quality of web and stages giving versatile application. A considerable lot of these stages acknowledge a decentralized control methodology, depending on unequivocal user assent for yielding consents that the apps request. Users need to depend basically on network appraisals as the signs to order the conceivably perilous and unseemly apps despite the fact that network evaluations traditionally reflect feelings in regards to guessed usefulness or execution as opposed to concerning dangers. With the ascending of HTML5 web apps, such frameworks depending on user-assent consent will be more across the board. To think about the benefits of user-assent consent frameworks through a vast information gathering of Facebook apps, Chrome augmentations and Android apps. The investigation affirms that the current types of network appraisals utilized in application advertises today are not solid for demonstrating security chances an application makes. We discover some proof demonstrating endeavors to deceive or allure users for allowing authorizations: free applications and applications with develop content demand; "carbon copy" applications which have comparable names as that of famous applications likewise ask for a larger number of consents than is commonplace. Creators find that over each of the three stages well known applications ask for a greater number of consents than normal. LIBSVM: A Library for Support Vector Machines. Creators: Chih-Chung Chang and Chih-Jen Lin Description:LIBSVM is a library for Support Vector Machines . Creators have been effectively building up this bundle. The reason for existing is to push users to easily apply SVM to their applications. LIBSVM has increased wide status in machine learning and numerous territories. In this, creators possible all usage points of interest of LIBSVM. Issues, for example, tackling SVM advancement issues, hypothetical assembly, likelihood assessments, and parameter choice are examine in detail. Bolster Vector Machines are a famous machine learning strategy for arrangement, relapse, and other learning assignments. LIBSVM is right now a standout amongst the most generally utilized SVM programming.

III. 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, posttagging, 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.

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 profile 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.

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.

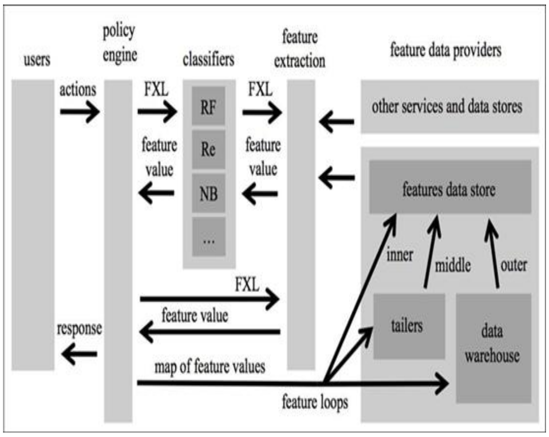

Fig. High level design diagram of the immune system deployed by Facebook.

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 profile 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 modeled the virus propagation with an email virus model and compared the behaviors 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. Facebook itself has confirmed spam as a serious issue, and taken steps to reduce spam content in users, newsfeed recently [Owens and Turitzin 2014]. Identifying spam on Facebook, however, evidently remains a hard problem. Despite of Facebook having a high performance immune system of their own [Stein et al. 2011][8,9,10], users still encounter an enormous number of spamand malicious content on regular basis. Existing approaches to detect spam in other online social media services like Twitter [Benevenuto et al. 2010; Grier et al. 2010; McCord and Chuah 2011; Wang 2010], cannot be directly ported to Facebook due to multiple issues. These include the public unavailability of critical pieces of information like profile, and network information, age of the account, no limit on post length, etc. There exists dire need to study spam content on Facebook, and develop techniques to identify it cogently, and automatically.

IV. METHODOLOGY

Our Facebook dataset are monitored by My Page Keeper which is our security application for Facebook. Basically the applications scan your wall and news feed and any time it detects something that looks malicious, it notifies you and inverts you to remove contents. My Page Keeper scans each URL using machine learning based classifiers that classifies social context associated with URL. MyPageKeeper has false positive rate is 0.005% and false negative rate is 3%. For Implementation and accuracy of My Page Keeper we refer interested readers to [10]. In our work, we need to store data of malicious applications and also need to store data for profiling of users to know which user is fake.

1, Data Collection Methodology

- ***D-Sample Dataset:***To identify malicious Facebook applications in our dataset, we start with a simple heuristic: If any post made by an application was flagged as malicious by MyPageKeeper, we mark the application as malicious. For every malicious app in the D-sample dataset we consider the time at which we observed the first post made by this app as the time at which the app was launched.
- ***D-Summary Dataset:***To select an equal number of benign apps from the initial D-Total dataset, we use two criteria:

1) None of their posts were identified as malicious by
MyPageKeeper, and
2)They are ―vetted‖ by Social Bakers [17], which monitors the ―social marketing success‖ of apps.
To match the malicious apps from total number of apps i.e. from D-sample dataset it is used.

 D-inst Dataset:App Permissions: We also want to study the permissions that apps request at the time of installation. For every application App_ID, we crawl https://www. facebook.com/apps/application.php?id= App_ID, which usually redirects to the application’s installation URL. Also for in fake user detection we need to collect data related to that account just like facebook real user and fake user.

2. Future Identification:

After collecting all user related data next step is identifying and retrieving set of features from that data attributes. Future Identification retrieves the relationships among all data attributes and used to demonstrate between fake user and real user.

3. Learning Classifiers:

It is a final stage of determination of fake accounts and malicious applications by using supervised machine learning classification algorithm. Supervised learners take datasets as input and construct predictive model. In machine learning we uses algorithms i.e. KNN algorithm and K-MEAN algorithm.

KNN Algorithm:

KNN is K- Nearest Neighbour algorithm. In KNN algorithm it classifies the data as nearest element according to its Euclidian formula.

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

The Euclidian distance between two points or tuples, say,
x= (x1,x2,…….,xn) y= (y1,y2,……..yn)

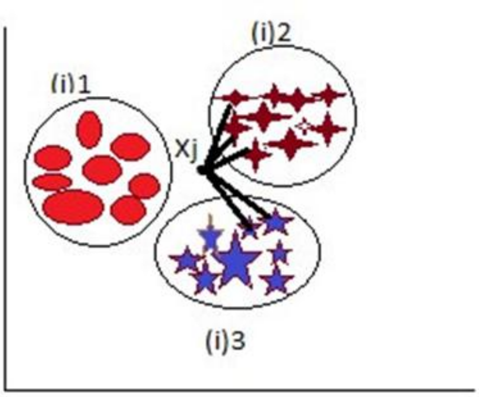

Figure.1. KNN Algorithm K-MEAN Algorithm:

K-MEAN is a clustering algorithm. in k mean algorithm same type of elements are grouped together. In which each cluster is represented as a centre of cluster.

Algorithm:-

1) Calculate mean value of cluster.
2)Assign each item to cluster which has closest mean.
3) Repeat 1 and 2 until we get same mean. Note:- Initially select 2 random values.

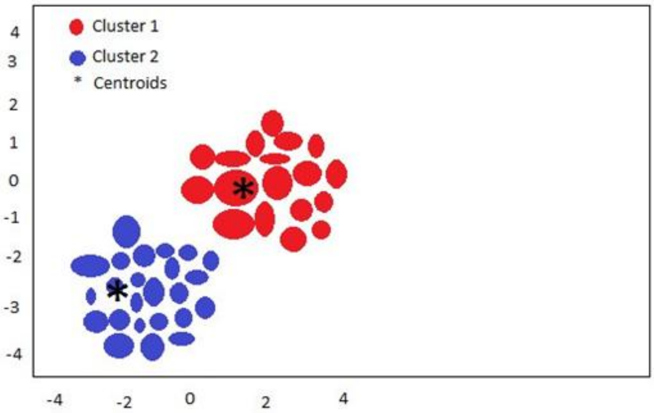

Figure. K-Mean Algorithm Input:- K= No. of clusters.

D={t1,t2,t3,………..,tn}
Output:- K= set of clusters.

V. Proposed System

Third party apps provide interesting features that means addictiveness in Facebook application. It has several disadvantages like hackers uses its potential for spreading malware and spam. Also due to black market services causes growth in fake user accounts.

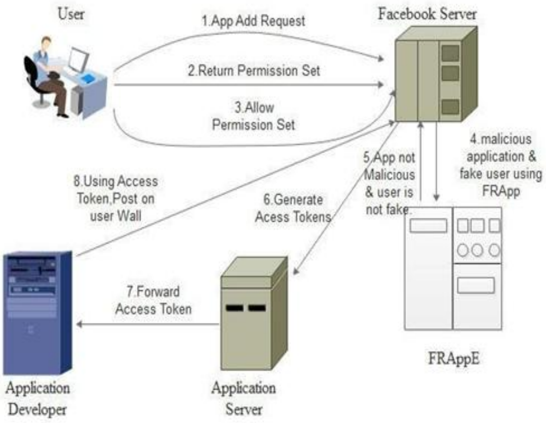

Figure. Proposed System Architecture

In proposed system we develop tool which detect the malicious application as well as fake user in facebook. In facebook when user trying to download some application then FRAppE detect that the app is malicious or not and also we detect the fake user present on facebook. First we see how the malicious applications are work: Unlike typical desktop and smart phone applications, installation of a Facebook application by a user does not involve the user downloading and executing an application binary. Instead, when a user adds a Facebook application to her profile, the user grants the application server:

1)Permission to access a subset of the information listed on the user's Facebook profile (e.g.,the user's e-mail address) 2)Permission to perform certain actions on behalf of the user (e.g., the ability to post on the user's wall). Facebook grants these permissions to any application by handling an OAuth 2.0[5] token to the application server for each user who installs the application. For detection of fake accounts on a very popular (and difficult for data collection) online social network, Facebook. Key contributions of our work are as follows. The first contribution has been collection of data related to real and fake accounts on Facebook. Due to strict privacy settings and ever evolving API of Facebook with each version adding more restrictions, collecting user accounts data became a major challenge. Our second contribution is the use of user-feed information on Facebook to understand user profile activity and identifying an extensive set of 17 features which play a key role in discriminating fake users on Facebook with real users. Third contribution is the use these features and identifying the key machine learning based classifiers who perform well in detection task out of a total of 12 classifiers employed[6].

Operation of our proposed system:

- **Step 1:** Hackers convince users to install the app, usually with some fake promise (e.g., free iPads).
- **Step2:** Once a user installs the app, it redirects the user to a Web page where the user is requested to perform tasks, such as completing a survey, again with the lure of fake rewards.
- Step3: The app thereafter accesses personal information (e.g. birthdate) from the user's profile, which the hackers can potentially use to profit.
- Step4: The app makes malicious posts on behalf of the user to lure the user's friends to install the same app (or some other malicious app, as we will see later).
- Step5: For fake user it admits the user request into FRAppE. It collects all data according to that account.
- Step6: After that it shows the result for malicious apps and fake user. It prevents the Personal information or surveys to sold it to third parties [7] to eventually profit the hackers.

V. 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.

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.

REFERENCES

[1]. Sazzadur Rahman, Ting-Kai Huang, Harsha V. Madhyastha, MichalisFalatous, "Detecting Malicious Facebook Applications," In IEEE/ACM Transactions on Networking, 2015.

[2]. V. Sri Roja, A. Vineela, Y. Sri Sanjana, U. PrasannaAnjanyulu, "FRAppE: Detecting Malicious Facebook Users".

[3]. Pete Burnap, Amir Javed, Omer F.Rana, Malik S.Awam, "Real Time Classification of Malicious URLs on TWITTER Using Machine Activity Data," In IEEE/ACM International Conference on Advance in Social Network Analysis and Mining, 2015.

[4]. KiranBhise, R. S. Shishupal, "Survey on Recognize Malignant Facebook Applications," In Vol. 4, IJSR, December 2015.

[5]. Yajin Zhou, Zhi Wang, Wu Zhou, Xixian Jiang. Hey You, "Get on My Market: Detecting Malicious Apps in Official and Alternative Android Market".

[6]. Application Authentication Flow using oauth 2.0. http://developers.facebook.com/docs/authentication.

[7]. D. Goldman, "Facebook tops 900 million users," 2012 [Online]. Available: http://money.cnn.com/2012/04/23/technolog y/facebookq1/ index.htm

[8]. A. Wang, "Machine Learning for Detection of Spam on Twitter Networks," In proceedings of the 26th Annual Computer Security Applications Conference, ACSAC‟2010, ACM.

[9]. HackTrix, "Stay away from malicious Facebook apps," 2013 [Online]. Available: http://bit.ly/b6gWn5

[10]. 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.

CH. Priyanka Raj " A Tool for Classification of Malicious App" .IOSR Journal of Engineering (IOSRJEN), vol. 08, no. 12, 2019, pp. 44-50.