iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 6 **Issue:** III **Month of publication:** March 2018

DOI: http://doi.org/10.22214/ijraset.2018.3506

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 6 Issue III, March 2018- Available at www.ijraset.com

Detecting the Pernicious Applications in Facebook and Restrict the user

V.Amulya[1], Almaas Nazneen[2], K.Chandhana[3], G.Harsha[4]
[1, 2 3, 4,] *Department of IT, LBRCE*

Abstract: ***Among all the Online Social Networks (OSN), Facebook is the prevailing one and it is used by billions of people transversely the world. Arbitrator apps are the major reason for the prevalence and habit-forming of Face book. Hackers are uncovering many new ways to inseminate spam and malware on these platforms, which we refer to as social malware. They can easily access the user personal details. By using existing mechanisms we cannot identify the Social malware. The main disadvantage of existing system is security is missing .FRAppE means Facebook Rigorous Application Evaluator, amicably the first tool converge on detecting pernicious apps on Facebook. In this paper, we proposed an approach to detect pernicious application in facebook using FRAppE tool under some set of constraints and also certain techniques are implemented to restrict the user by sending an email. The user can view it personally by login into his/her account in our website. Offensive information will not be added in our wall. So it is safe and secure.***

Keywords: Facebook apps, pernicious, online social networks, spam.

I. INTRODUCTION

Hackers are involved in many virus spreading process. Many people are easily propagating virus and they can easily get all the users personnel information. Hackers are taking the upper hand of other apps platform in order to develop the pernicious applications. There are many number of ways that hackers can get bonus from harmful apps. These poisonous apps can reach to huge number of users and their friends to spread malware. At first existing system contains a security app in facebook named as Mypagekeeper ,this app informs about virus propagation in facebook on daily basis. In existing system, Frappe tool is implemented to detect this harmful applications in facebook and this tool uses the data from the facebook security app .This tool only detects those applications that spreads virus. There is no security for users' data. The drawback in the existing system we face many problems and we focused on detecting pernicious applications. It gives only an outline about the warnings in facebook.

In proposed system we try to find out the differences between original and harmful apps. Pernicious apps share common names with other apps and they require more permissions than other apps. When user is trying to install a harmful application or trying to post malicious content then user wall gives a warning message that the application found is malicious and also we will block that user. It is safe and secure, because user can see it personally.

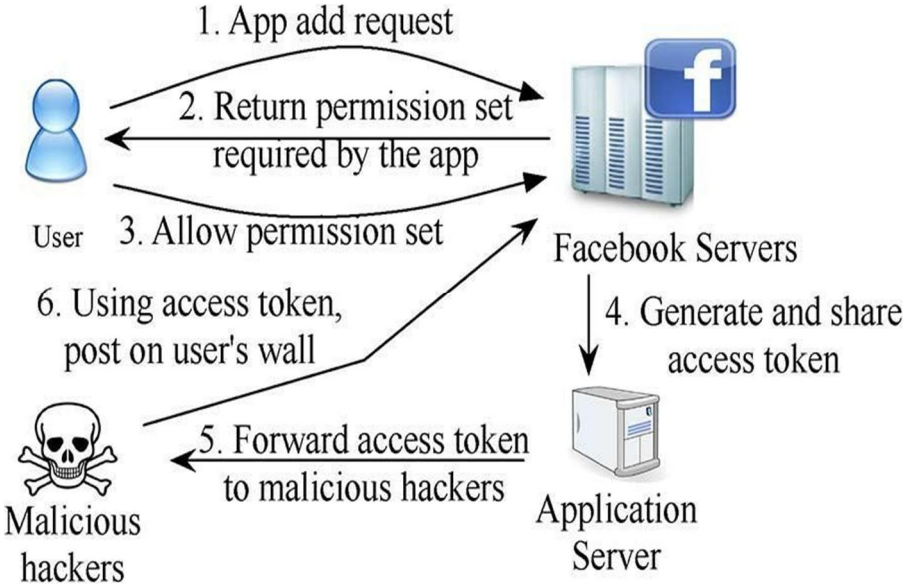


Fig 1.1 System Architecture for Existing System

II. RELATED WORK

In this paper[1] ,they made an analysis of the avoidance of accusation used by the twitter spammers and designed huge number of features to detect twitter spammers. Finally, they gave a definite structure of different features that are used by in their system and also proposed systems. They compare Navie Bayesian and Decision tree algorithms and they showed that navie Bayesian gave high accuracy. The advantage in the paper is It activates detection of malicious apps that propagate irrelevant message and virus by attracting normal Users to install them.In this paper[2], they analyze the productiveness of user authority systems by collecting large amount of data from facebook apps, chrome extensions and android apps. They analyzed that the current apps are not suitable to provide security for user's private information. They differentiate original and virus spreading apps by virus applications requires more user personnel information than original applications. The algorithm used in this paper is LSB algorithm. High complexity is the main disadvantage in their paper. In this paper[3],their main goal is to develop a tool i.e., FRAPPE to detect he applications that are the main source for populating virus. They introduced two classifiers FrapeLite and Frappe to identify the virus spreading apps. They used support vector machines and decision trees and bayes algorithms. In this paper they automatically detect the apps those are used to spread virus. In this paper[4] their main goal is incursion user information and expose users unpopular junk. The attackers have gigantic resources at their distribution. They founded on full time skilled labour and have a control over contaminate accounts and access to global systems that are used to spread spam. The techniques used in this paper are natural language processing and sequential analysis algorithms. The disadvantage is it is not able to attract many users.

III. EXISTING SYSTEM

In previous systems they focused on only identifying harmful applications, but they did not concentrated on detecting the main source of those applications. They only detect separate URLs and posts as spam or not, but they failed in telling whether an application is malicious or not. Previous system works concentrated on accounts created by spammers instead of malicious application.

IV. PROPOSED WORK

These problems are overcomes in the proposed system. In the proposed system we detect malicious apps using Frappe tool. When user is trying to post harmful posts in their facebook page, those posts are detected by Frappe tool. When we found any installation of those virus spreading apps, we block the user that the app found is a malicious app by displaying an alert message. It is very secure and safe. Offensive posts which are not related are detected by Frappe tool. These posts will not post on the users public wall. Unneeded information will not be added in user's wall. FRAppE stands for Face book's Rigorous Application Evaluator which is helpful in controlling the entire system. In Authentication and Authorization module, the user will register their details and login into their account to view their profile to see all the contacts, the user will do all the works here. They can easily access the data from the database. If any harmful app is found in the profile, it will be detected using FRAppE and after detecting it we will block the user who is trying to install that virus propagating app. Here, we use the SVM Algorithm as a classifier which consists of various parameters that are specified to identify the app, post; picture, etc. are malicious or benign.

V. IMPLEMENTATION

A. There are four modules in our project. Those are

1) *Data collection:* The data collection component has two subcomponents: the gathering of facebook apps with URLs and moving for URL redirections. Whenever this component obtains a facebook app with a URL, it executes a crawling yarn that follows all redirections of the URL and looks up the consequent IP addresse
2) *Feature extraction:* The feature extraction component has three subcomponents: assemblage of identical domains, discovering entry point URLs, and extracting feature vectors. To categorize a post, MyPageKeeper evaluates every embedded URL in the post. Our key novelty lies in considering only the social context for the classification of the URL and the related post.
3) *Training:* The training module has two modules: retrieval of account statuses and training of the classifier. Because we use an offline supervised learning algorithm, the feature vectors for training are relatively older than feature vectors for classification. To label the training vectors, we use the account status; URLs from suspended accounts are considered spiteful whereas URLs from active accounts are considered original apps. We regularly update our classifier using labelled training vectors
4) *Classification & Detection:* The classification component executes our classifier using input feature vectors to classify mistrustful URLs. When the classifier returns a number of malevolent feature vectors, this component flags the corresponding URLs information as guarded. The classification module uses a Machine Learning classifier based on Support Vector

Machines, but also utilizes several local and external white lists and blacklists that help speed up the process and increase the total performance.

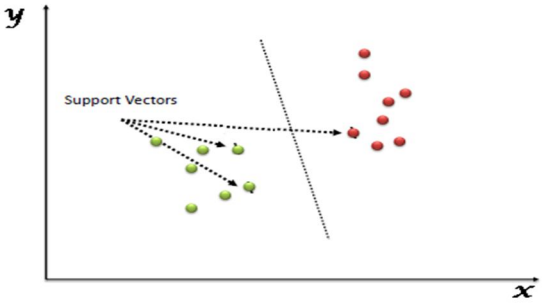
The main aim is to protect oneself from social malware. One the user log in to the account the system works in the following manner: After log in, the user can upload any post he wishes to, this is then passed to a training set block. The next state is that of a feature extraction which checks the specified features.

The following diagram describes about the modules involved in our project.

Fig 5.1: Flowchart of Modules

B. Svm classifier

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well. Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships.



C. Classification svm Type

For this type of SVM, training involves the minimization of the error function:

$$\frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i$$

here C is the capacity constant, w is the vector of coefficients, b is a constant, and ξ_i represents parameters for handling non separable data (inputs). The index i labels the N training case

D. Authentication & authorization

In this module first user has to register then only the person can access the database. After the Registration, the user can login into the site. Only the authorized user can login and able to do all their works. In this registration all the personnel information about the user who wants to use the application and view their profile.

E. Detect harmful app

When user trying to install virus propagating app then the Frappe tool identifies whether the app is malicious or not. It uses two classifiers FrappeLite, Frappe. When a malicious post is appear on users wall,it will send a message whether to continue the installation or not.

F. Block the user

After the detection of offensive posts a warning message is sent to the user saying that the user is blocked. The user can view it by login to his/her account .This is highly guarded.

VI. CONCLUSION

Applications present in now a days are easy for the hackers to populate virus on facebook. But, it is necessary to understand the basic operations of harmful apps. In presented approaches they used many techniques, which are not secure and safe. So here we used a tool name called Frappe to detect malicious content in facebook . In addition to that we sent a warning mail to the user and also we will block the user who is trying to install harmful apps. So, here we presented a technique that is beneficial and safe to the end users.

VII.EXPERIMENTAL RESULTS

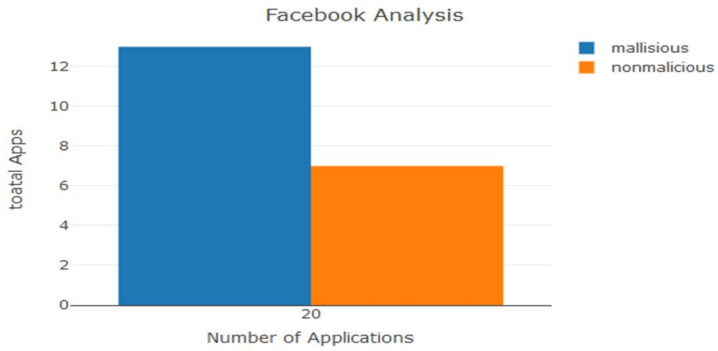


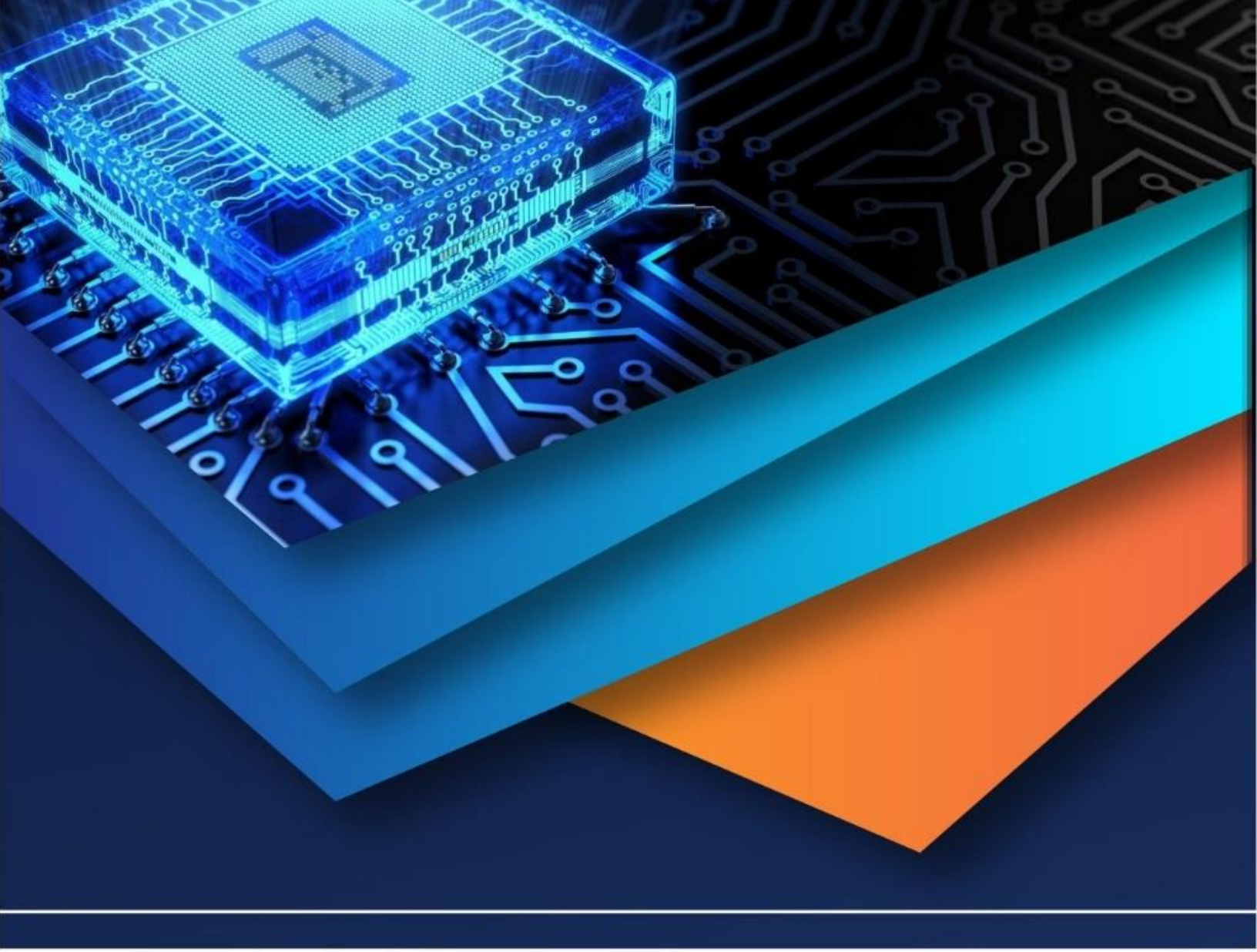
Figure7. 1: Observed results based on some parameters.

In x-axis we are taking the total number of applications in our dataset and in y-axis we are taking range of values .In the bar graph we shown that number of malicious and non malicious applications among the total applications. We are taking different attributes to check whether an application is malicious or not.

The attributes that we had taken are reputation of URLs, number of required permissions, Use of different client IDs, app name similarity. Malicious apps use a different client ID value in the app installation URL, thus enabling the propagation and advertising of other malicious apps. If any malicious app redirects to different other URLs and requires more number of permissions from the user then we will mark it as malicious application. Malicious apps often share similar name with other applications.

REFERENCES

[1] C. Yang, R. Harkreader, and G. Gu, “Die free or live hard? Empirical evaluation and new design for fightingevolving Twitter spammers,” in Proc. RAID, 2011, pp. 318–337.
[2] P. Chia, Y. Yamamoto, and N. Asokan, “Is this app safe? A large scale study on application permissions and risk signals,” in Proc. WWW,2012,pp. 311–320
[3] J. King, A. Lampinen, and A. Smolen, “Privacy: Is there an app for that?,” in Proc. SOUPS, 2011, Art. no. 12.
[4] T. Stein, E. Chen, and K. Mangla, “Facebook immune system,” in Proc.4th Workshop Social Netw. Syst., 2011, Art. no. 8.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY