ISSN No. 2395-0900
International Journal of Engineering Studies and Technical Approach
IJESTA

Title: Search Rank Fraud and Malware Detection in Google Play

Prajakta Rane*, Priya Mishra, & Dr. Archana Chaugule**

,**&Department of Computer Engineering, PCCOER-Ravet.***

ABSTRACT:

The business success of Android app markets such as Google Play and the enticement model they tender to trendy apps, make them attractive targets for fake and malicious behaviors. Some fake developers dishonestly increase the search rank and reputation of their apps (e.g., through false review and fake installation counts), while malicious developers use app markets as a launch pad for their malware. The motivation for such behaviors is impact: app popularity surges translate into financial benefits and expedited malware proliferation. In this paper, we establish FairPlay, a new system that discover and leverages traces left following by fraudsters, to detect both malware and apps subjected to search rank fraud. In this paper, we look for to recognize both malware and search rank fraud subjects in Google Play.

Keywords- *Android Applications, Fairplay, Fraud rating*

I INTRODUCTION:

False developers repeatedly utilize crowd sourcing sites to hire teams of agreeable people to assign fraud cooperatively, emulate reasonable, impulsive behavior from dissimilar people. We call this performance "search rank fraud". In addition, the efforts of Android market to recognize and eliminate malware are not at all times doing well. For case in point, Google Play uses the Bouncer system to remove malware. Preceding mobile malware discovery work has paying attention on active investigation of app executables as well as static analysis of code and permissions. However, recent Android malware examination revealed that malware evolves quickly to avoid anti-virus tools.

II LITERATURE SURVEY

1. Paper Name: Android Permissions: A Perspective Combining

   Authors: Bhaskar Pratim Sarma, Ninghui Li, Chris Gates, Rahul Potharaju, Cristina Nita-Rotaru, and Ian Molloy.

   Year: 2012.

   Description: In this paper we exploit earlier approaches for dynamic analysis of application behaviour as a method for detection malware. The detector is embedded associate exceeding overall framework for assortment of traces from an unlimited variety of real users that support crowd sourcing. Our framework has been incontestable by analyzing the information collected within the central server victimization.

2. Polonium: Tera-scale graph mining and inference for malware detection.

   Authors: D. H. Chau, C. Nachenberg, J. Wilhelm, A. Wright, and C. Faloutsos.

Year: 2011.

Description: In this paper, author developed four malicious applications, and evaluated ability to notice new malware supported samples of renowned malware. Author evaluated many mixtures of anomaly detection algorithms, feature selection technique and also the variety of high options so as to seek out the mixture that yields the most effective performance in detecting new malware in android application. Result shows that the projected framework is effective in detecting malware on mobile devices normally and on android applications specifically.

3. Fair Play: Fraud and malware detection in Google play

   Authors: Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, Duen Horng Chau.

   Year: 2014

   Description: In this paper, author proposes a proactive theme to identify zero-day android malware. Without using malware samples and their signatures, our scheme is actuated to assess potential security risks exposed by untrusted apps. Specifically, we have developed a automatic system referred to a risk ranker to scalably analyze whether a specific app exhibits malicious behaviour (e.g,launching a root exploit or causing background SMS messages).

4. Paper Name: Discovering opinion spammer groups by network footprints. In Machine Learning and Knowledge Discovery in Databases

   Authors: Junting Ye and Leman Akoglu

   Year: 2015.

   Description: In this paper, we have studied a way to conduct effective risk communication for mobile devices. This has emerged jointly on the quickest growing operative systems. In Gregorian calendar year 2012, Google announced that four hundred million devices are activated, with one million devices being activated daily. The Google Play crossed more than fifteen billion downloads including year 2012, and was adding around one billion downloads per month from December 2011 to December 2012.

III ARCHITECTURE OF PROPOSED SYSTEM

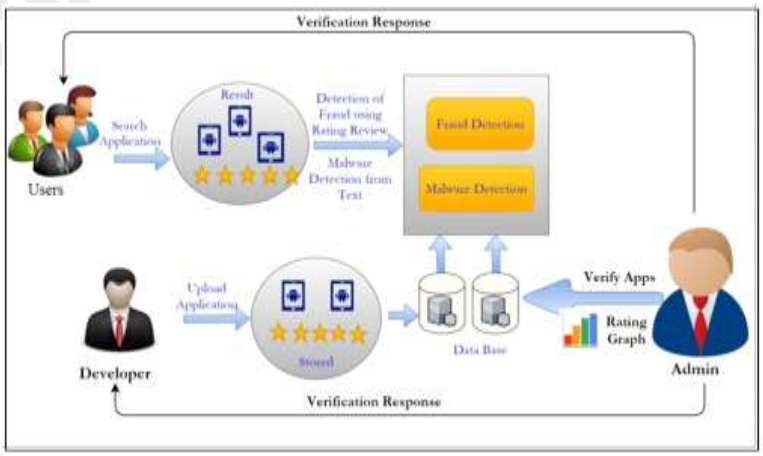

Fig 1: Proposed System Architecture

In proposed system user and developer both have to do the registration. Developer will login into the system and upload the application .This application is stored in the database. Admin has the authority of accessing the database and reviewing accordingly using PCF algorithm. Admin verifies the app through the graph rating. After that user will login and search for the required application. The application uploaded by the developer is viewable to the user. The fraud application is detected using rating review and through this we come to know whether application is fraud or not. Malware detection refers to malicious software that exploits target system vulnerabilities that could be detected in application. Fraud detection detects background server-based processes that examine users and other defined entities access and behavior patterns, and typically compares this information to a profile of what is expected.

IV PROPOSED SYSTEM

We propose PCF (Pseudo clique Finder), an algorithm that takes input as the set of the reviews of associate app, organized by days, and a threshold value. PCF outputs a set of known pseudo-cliques and are shaped throughout contiguous time frames. Once the app has established a reviews , it find the day's most promising pseudo-clique that begins with each analysis and then add different reviews to a candidate pseudo-clique. It manages to keep the pseudo set (of the day) with the very best density. With this work-in-progress, pseudo-clique adds diverse reviews whereas the weighted density of the new pseudo-clique is either equal or it exceeds to previous density. In proposed system user and developer have to register. Developer can login to the system and upload the application. Then user can login and rummage around the appliance. User will see the appliance uploaded by the developer. Once finding application that user needs to transfer user can choose search rank fraud detection and then he can check the malware within the application. Once user is satisfied, he can transfer the application.

V PROPOSED ALGORITHM

Input: Days, an array of daily reviews, and q, the weighted threshold density.

Output: All Cliques, set of all detected pseudo-cliques.

Step 1 for d :=0 d <days.size(); d++

Graph PC := new Graph();

bestNearClique(PC, days[d]);

c := 1; n := PC.size();

Step 2 for nd := d+1; d <days.size() c = 1;

bestNearClique(PC, days[nd]);

c := (PC.size() >n); endfor

Step 3 if (PC.size() >2)

allCliques := allCliques.add(PC); return

Step 4 function bestNearClique(Graph PC,

if (PC.size() = 0)

ISSN No. 2395-0900
International Journal of
Engineering Studies and Technical Approach
IJESTA

Step 5 for root := 0; root <revs.size();
Graph candClique := new Graph ();
candClique.addNode

Step 6 do candNode :=
if (density(c and Clique [c and Node) q))
candClique.addNode(candNode);

Step 7 while (candNode != null);
if (candClique.density() >maxRho)
maxRho := candClique.density();
PC := candClique; endfor;
else if (PC.size() >0)
if (density(candClique (candNode) PC.addNode(candNode);
while (candNode != null);
return

VI RESULT

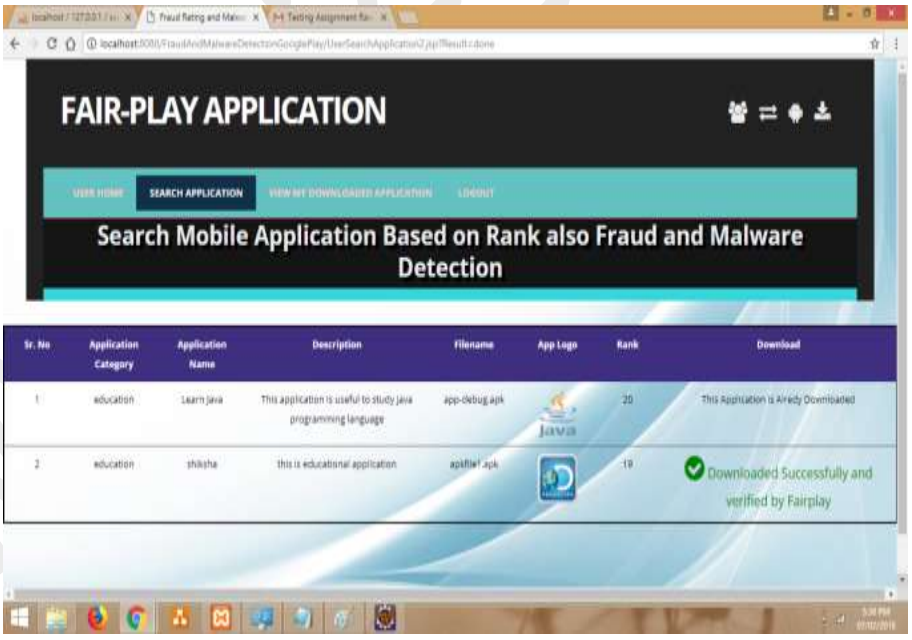

Fig 2: User Download Application

ISSN No. 2395-0900
International Journal of
Engineering Studies and Technical Approach
IJESTA

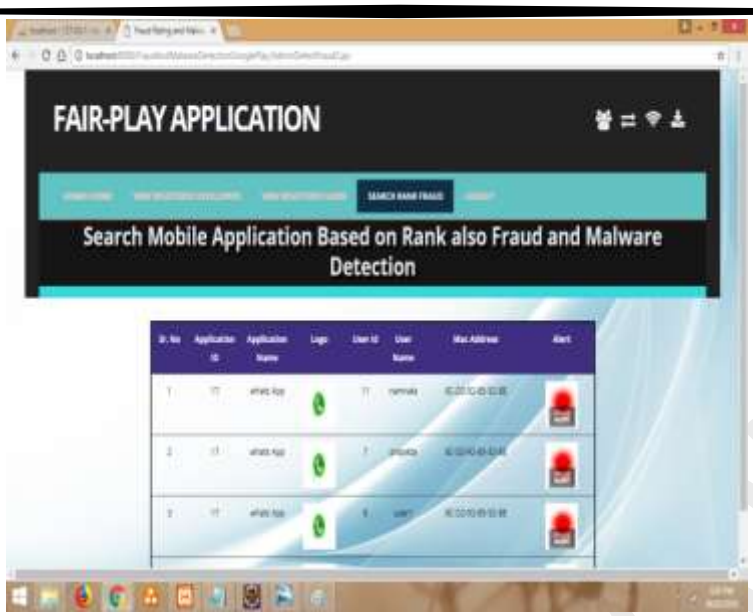

Fig 3: Fairplay (Fraud Application Detection)

VII CONCLUSION

Hence we developed PCF that reviews pseudo-cliques fashioned by reviewers with onsiderably overlapping co-reviewing activities across short time windows. We have introduced Fairplay, a system to find each deceitful and malware Google Play apps through search ranking using graph ratings.

VIII ACKNOWLEDGEMENT

With huge joy, we are publishing this paper as a part of the syllabus of B.E. Computer Engineering. It gives us proud opportunity to total this paper effort under the precious leadership of Principal for given that all amenities and help for level development of paper work. We would also like to express thanks all the Staff Members of Computer Engineering Department, Management, friends, Who have directly or indirectly guided and helped us for the guidance of this paper and gives us an endless sustain right from the step the idea was conceive.

IX REFERENCES

i. Bhaskar Pratim Sarma, Ninghui Li, Chris Gates, Rahul Potharaju, Cristina Nita-Rotaru, and Ian Molloy, "Android Permissions: a Perspective Combining Risks and Benets," in Proceedings of ACM SACMAT, 2012.

ii. D. H. Chau, C. Nachenberg, J. Wilhelm, A. Wright, and C. Faloutsos, " Polonium: Tera-scale graph mining and inference for malware detection, " in Proceedings of the SIAM SDM, 2011.

iii. Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, Duen Horng Chau, " Fair Play: Fraud and malware detection in Google play."

iv. Junting Ye and Leman Akoglu. "Discovering opinion spammer groups by network footprints." in Machine Learning and Knowledge Discovery in Databases, 2015.

v. Takeaki Uno, "An efficient algorithm for enumerating pseudo cliques," In Proceedings of ISAAC, 2007.

vi. Steven Bird, Ewan Klein, and Edward Loper, " Natural Language Processing with Python," O'Reilly, 2009.

vii. Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan, "Thumbs Up? Sentiment Classification Using Machine Learning Techniques," In Proceedings of EMNLP, 2002.