Android Malware Detection Using Machine Learning

Rishab Agrawal
Department of Information Technology,
A.P. Shah Institute of Technology,
Thane(M.H), India 400615
Email:agrawalrishab210@gmail.com

Vishal Shah
Department of Information Technology,
A.P. Shah Institute of Technology,
Thane(M.H), India 400615
Email: vishalshah571@gmail.com

Sonam Chavan
Department of Information Technology,
A.P. Shah Institute of Technology,
Thane(M.H), India 400615
Email: sonamchavan48@gmail.com

Prof. Ganesh Gourshete
Department of Information Technology,
A.P. Shah Institute of Technology,
Thane(M.H), India 400615
Email: gdgourshete@apsit.edu.in

Prof. Nahid Shaikh
Department of Information Technology,
A.P. Shah Institute of Technology,
Thane(M.H), India 400615
Email: nashaikh@apsit.edu.in

2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE) 978-1-7281-4142-8/20/$31.00 ©2020 IEEE 10.1109/ic-ETITE47903.2020.491

***Abstract*— Malware is one of the major issues regarding the operating system or in the software world. The android system is also going through the same problems. We have seen other Signature-based malware detection techniques were used to detect malware. But the techniques were not able to detect unknown malware. Despite numerous detection and analysis techniques are there, the detection accuracy of new malware is still a crucial issue. In this paper, we study and highlight the existing detection and analysis methods used for the android malicious code. Along with studying, we propose Machine learning algorithms that will be used to analyze such malware and also we will be doing semantic analysis. We will be having a data set of permissions for malicious applications. Which will be compared with the permissions extracted from the application which we want to analyze. In the end, the user will be able to see how much malicious permission is there in the application and also we analyze the application through comments**.

I. INTRODUCTION

Malware is nothing but the short name for malicious software, in general, referred to many forms of hostile or intrusion creating software, spyware, Trojan horses, backdoor, and rootkits. The main aim of malware is to damage, steal, disrupt or do some bad actions. Malware is powerful enough to infect any kind of computing machine running application, and the prevention of malware is being well studied for personal computers (PC). A Smartphone device the detection techniques used is lagging far behind as compared to the fast growth of the mobile population is being

Some recent survey has shown that there are about 2.1 million android applications are there in the market. Due to increase in usage of the android system has led to more rollout of android malware. This malware is spreading in the market by the third parties developing applications. The Google android market also doesn't promise to guarantee that all the applications listed are threat free. There are also such reports about Trojans applications that if downloaded, their malicious code is also installed and cannot be easily detected by Google's technologies during publication in the Google android market. The android threats include banking Trojans, spyware, bots, root exploits, SMS fraud, phishing & fake installer.

II. OBJECTIVES

To to provide security to the android user from the various malwares by analyzing the permissions given to that particular application

to analyze the android application comments by semantic analyzing technique

to give interactive user interface to the user to detect the malicious percentage of the application.

III. VARIOUS TECHNIQUES TO INSERT MALWARE

There are various ways and methods through malware or any malicious file can enter your system or application. Some of the common techniques of malware getting intruded into the system are as follows: -

A. Penetration:

Penetration techniques commonly used for malware applications for installation activation & running on the android system are repackaging, updating and downloading.

B. Repackaging:

It is among the common techniques for malware developers to install malicious applications on an android platform.Repackaging approach for popular applications and misuse them as malware. The developer downloads such types of application and recodes them and adds their own malicious code and uploads that application to the official Android app store or on the different markets.

C. Updating:

This technique is much more difficult for detecting malware.The malware developer may still use repackaging but instead of encoding inflict code to the application the developer may include an update component that will able to download malicious code at the run time.

D. Downloading:

This is the most traditional attacking technique. The malware developer needs to attract the user to download interesting and attractive applications

IV. LITERATURE REVIEW

In literature[1], the paper was published in the year 2018. They have performed the malware detection with the help of 300 malware files and 300 benign apk files, also they managed to generate only 183 malware and 300 benign gray-scale images. The other 117 malware samples were unable to generate into images because the apk files were corrupted or either that files did not contain classes.dex file. Also, the accuracy was much less in all the algorithms they used. They have detected with the help of three different classifier techniques namely the k-nearest neighbor(KNN), Random Forest (RF), and Decision Tree(DT).

In literature[2], the paper was published in the year 2017. They had used different machine learning algorithms such as Naive Bayes,j48, random forest, Multiclass classifier and multilayer perceptron to detect android malware and evaluate the performance of each algorithm. Here they implemented a framework for classifying android applications with the help of the machine learning techniques to check whether it is a malware or normal application. For validating their system they have collected 3258 samples of android apps and those have to be extracted for every application, extract their features and have to train the models going to be evaluated with the help of classification accuracy and time taken for the model.

In literature[3], the paper was published in the year 2016. They have proposed a Robotium program in an Android sandbox that can trigger any android application automatically and monitor its behavior. The program has a UI Identification automatic trigger program that can click the mobile applications in a meaningful order. The program was able to perform larger-scale experiments. They also tried to build a decision model using behavior that has collected with the help of the random forest algorithm. It has been able to determine whether the unknown application is malware and also shows its confidence value. They could store the result and also the confidence value of the unknown apk file in their database.

In literature[4], the paper was published in the year 2018. They have proposed the android malware detection system with the help of permissions, APIs, and also with the presence of different key apps information such as, the dynamic code, Reaction code, native code, cryptographic code, database, etc. as the feature to train and build classification model just by using various machine learning techniques which can automatically distinguish malicious Android apps(Malware) from the legitimate ones.

V. EXISTING SYSTEM

In the existing system, the application permissions are extracted to detect the malware and executed through the command prompt. A proper GUI was not provided to execute the tasks. All the commands were run through the command prompt. It was difficult for the non-technical user to use the system. And also Semantic analysis was not implemented.

VI. PROPOSED SYSTEM

In the proposed system, we are doing the permission-based analysis and also the semantic analysis. The permission-based analysis is been done on the web-based UI while the existing systems were just doing it all on the local machine in the command prompt.

In our system, we have implemented an admin panel as well as a user panel. In the admin panel admin have the access to upload the apk files and its details along with its categorization and also the admin can upload the comment that can be used for semantic analysis.

In the user-panel the user can see the select the category of the application and can see its details like pricing description name. User can see the malicious percentage of the application. And the processed output of the semantic analysis will be displayed to the user in the form of graph and the user will get a proper review of the application.

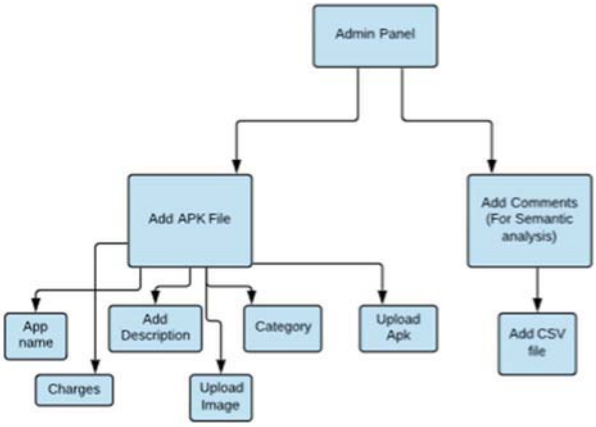

Fig No. 1.1 Block diagram of Admin panel

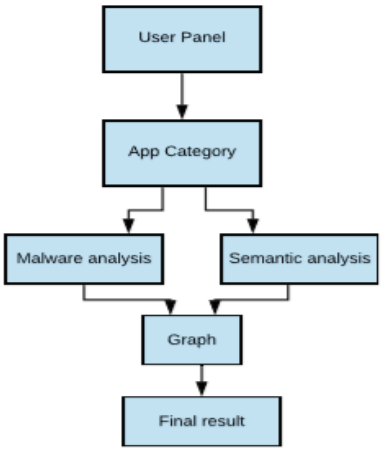

Figure No 1.2- Block diagram of User panel

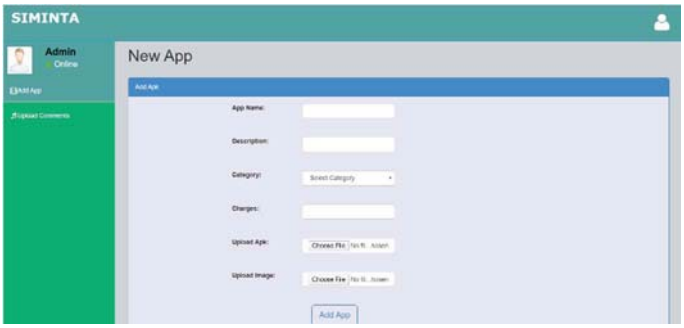

Figure No 2.1- GUI of admin panel(Add Apk)

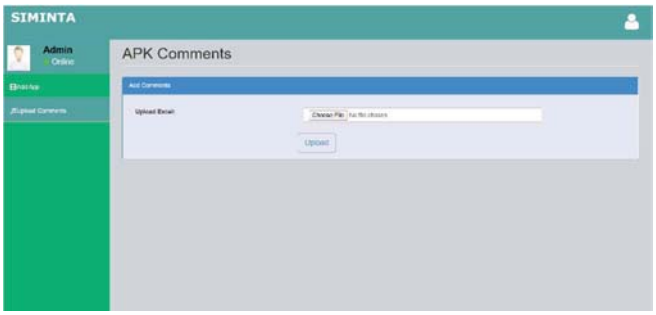

Figure No2.2-GUI of Admin Panel(Semantic analysis)

In the above figure 2.1 & 2.2, we have shown the GUI of our admin panel. Here the admin will fill the details of the particular application and upload the apk file along with its image. The permissions will be extracted from this apk file and will be stored in the database.

Figure No 2.3- GUI of user panel

In the above figure 2.3, we have shown the GUI of the user panel, where the user will be able to see the applications genre wise and can explore various applications according to its requirements. Once the user selects particular genre, the user will get list of various applications under that genre and once the user clicks on particular application, the comments along with semantic analysis result will we available to the user.

VII. RESULTS

The Malware Detection can detect many different permissions based on the which it has been asked and also which of the permissions which it has been taken by default. Also, the semantic analysis is been used to get the proper comments resultant it in to get if the application is been proper or not. The permission-based analysis and also the semantic analysis gives the proper output so that the user can use those particular apps or not.

VIII. CONCLUSION AND FUTURE WORKS

In our work, we propose a system for permission analysis and semantic analysis. Our system is also used to detect malware permissions based on an application by comparing it with a dataset. This proposed system can be applied in the fields of the security system and also for the n users like a malware detection software. However, there are limitations in our system. The permissions which we are defining are as per our but it can differ from users to users. The permissions which the user likes that it is not a malware-based can be malware for any other user. Future works will contain the improvement of that.

IX. REFERENCES

[1] Darus,Fauzi Mohd,Salleh Noor Azurati Ahmad,and Aswami Fadillah Mohd Ariffin.”Android Malware Detection Using Machine Learning on Image Patterns”2018 Cyber Resilience Conference(CRC).IEEE,2018.

[2] Vrama,P.Ravi Kiran,Kotari Prudvi raj,and KV Subba Raju.”Android mobile security by detecting and classification of malware based on permissions using machine learning algorithms.”2017 InternationalConference on I-SMAC (IoT in Social,Mobile,Analytics and Cloud)(I-SMAC).IEEE2017.

[3] Chang,Wei-Ling, Hung-Min Sun,and Wei Wu.”An Android Behaviour-Based Malware Detection Method using Machine Learning.” 2016 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC).IEEE, 2016.

[4] Koli, J. D. “RanDroid: Android malware detection using random machine learning classifiers.” 2018 Technologies for Smart-City Energy Security and Power (ICSESP).IEEE,2018.