Malware Detection in Android Apps

Nishad Sunil Mantaprasad[1] Dubey Shivam Rakesh[2] Mishra Vinay[3] Prof. Ankit Sanghavi[4]
[1,2,3,4]Department of Computer Engineering
[1,2,3,4]Alamuri Ratnamala Institute of Engineering & Technology, A.S Rao Sapgaon, Shahapur Maharashtra, India

Abstract— Android is the leading market in the world and we can easily say that it is one of the fast growing technologies which influence lots of users all over the world. But due to its popularity & huge crowd it became the main targeting center for the attacker. To solve this problem there are many methods being used such as Anti-virus to find the malware in the app which match the signature of the app with its signature stored in its database which is also bypassed by the attacker by using new tactics. There is a security check done by the play store to stop the uploading of malicious applications into it. But the truth is that there are a lot of malicious applications available in the Play Store even after the security check. We use machine learning to classify whether an application is benign or malware. The static analysis is mainly focused on the manifest.xml file of an Android application and the dynamic analysis will be based on the actions it will be triggering while running on a mobile device. Our approach is to design an app which is truly based on the tensor flow model. Huge collection of dataset is used to perform static analysis of an app. In order to detect malware in the app which is miserably loaded by the user in their smartphone from the external sources.

Keywords: Android, Malware Analysis, Tensor Flow, Machine Learning, Static Analysis

I. INTRODUCTION

In present growing market android is the most famous platform for Smartphone's, the market growth rate increases gradually and it is now near to 90.1%. Android is one of the most used Operating System in smartphones. With the rapid development of mobile technology, tablets and smartphones have evolved sophisticated functions at lower cost.

Android occupies a major share in the mobile application market. The popularity of the Android platform has stimulated the interest for cyber-criminals to develop the malicious applications to steal sensitive information and compromise Android smartphones. Android mobile has become an easy target for the attackers. The main reason is the user ignorance in the process of installing and usage of the apps.

In order to protect the android app from the attacker there are various anti-viruses present but it is easily bypassed by the attacker.

So, In order to protect the smartphone we need to adopt smart techniques which will be able to detect every possible malicious app.

Hence, there is no doubt to use machine learning to prevent the Android smartphone from the malware.

II. PROBLEM DEFINITION

Our approach to build the software to detect the malware containing app which is surpassed by the anti-virus security and other security provided by the smartphone and app store services. Static analysis is done by targeting the manifest file which provides the user-permission to the android app. After collecting all the permission access by an app, put no. 0 & 1 based on their access granted or not whereas 1 for granted and 0 for not granted. For the dynamic analysis we extract the apps API call information and determine it, which is based on trained dataset. Overall the result of malware is the combination of static and dynamic analysis.

III. RELATED WORK

In earlier research paper we came to know that there are two methods for detecting, android app is malicious or genuine. One is static and the other one dynamic. Static methods occur during the app installation process and the Dynamic method is held during runtime of an app [1].They also stated that there are two ways to detect and classify the android app that it is malicious or not i.e., static and dynamic. They also include that we can verify the android app by checking the user-permission setting in the app. They were told to create a dataset of permission taken by an app & by using ML algorithm detect the malware app.In order to use machine learning algorithms in classification of android malware, first the permission data set of all the apps are to be extracted. Database is formed by placing a 1 if the permission is present in the Androidmanifest.XML file and a 0 if not [2].

The android manifest.xml file to detect the malicious app. By considering that each app is decomposed from the manifest file, a.xml into permission and intent filters using Python's Element tree. Hence, they classify the extracted feature into three categories such as Request permission, Declare permission & Intent filter. Its scope is limited because it deals with the manifest file only.

Android.apk file are used for study. Initially they categorize the file into two forms which are malicious or benign. Later the APK file was unpacked to extract the classes.dex file. It is a binary file containing Dalvik opcode. Tensor flow model can we used in which we deploy the machine learning trained dataset in android app.

Paper also proposes a method to detect malware in Android smartphones using API (application programming interface) classes. They use machine learning to classify whether an application is benign or malware. Furthermore, we compare classification precision rate from machine learning. This research uses 51 APIs package classes from 16 APIs classes and employs cross validation and percentage split test to classify benign and malware using Random Forest, J48, and Support Vector Machine algorithms. They use 412 total application samples (205 benign, 207 malware). We obtain that the classification precision average is 91.9%.

All rights reserved by *www.ijsrd.com*

IV. PROPOSED SYSTEM

Firstly, we create an android app which is more user friendly and having the effective interface for the user to interact with the app. App gives the list of all the other apps through the package manager which is installed in our mobile device [2].We have huge collection of datasets of malware and benign apps in the form of csv file. The Dataset consist of 1 or 0 format to reveal that app contain required permission.

Dataset is used to obtain the trained model with the help of Naïve Bayes classifiers algorithm to classify the app is benign or malware. Tensorflow model also imported to deploy the system in app format.

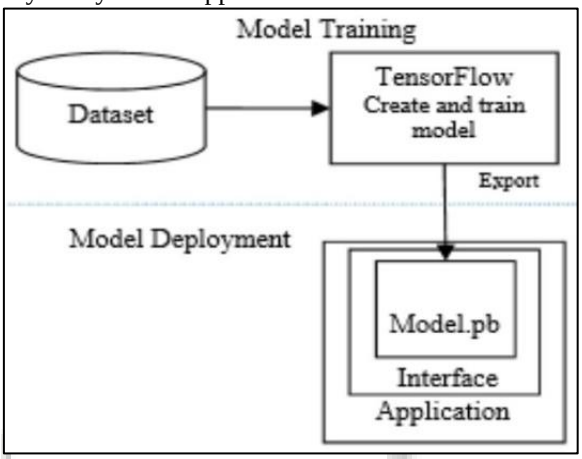

Fig. 1: System Architecture

For training purpose, Gaussian Naive Bayes classifier algorithm is used because it provides the highest accuracy for the malware detection in android app.

One of the simplest and most powerful machine learning techniques is Bayes classifier. This is probably due to its simplicity, linear computational complexity and accuracy. It is also referred to as Naive Bayesian Classifier. Because it makes the naive assumption that all the features representing the data are independent for a particular choice of the behavior one is trying to learn. The naive Bayesian classifier consists of training and testing phases. In the training phase, a model learns from a sufficient number of training data containing both benign and malicious android apps. Then during testing or detection phase, the model infers whether the given test app is benign or malicious using the model learnt during the training. We use Gaussian Naïve Bayes for our project.

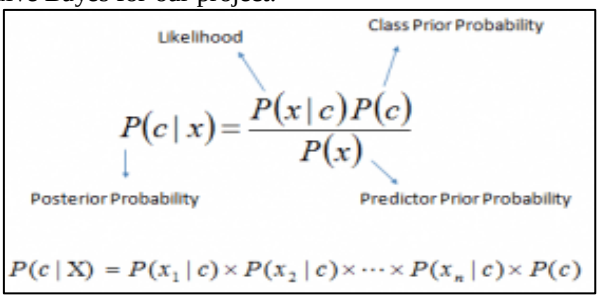

Fig. 2:

For our Data set we have a large collection of genuine and malicious app. To evaluate the effectiveness of the proposed approach, we calculate true positive, true negative, false positive and false negative rates in our experiments. These measures are defined as follows:

– True Positive

Number of android malware apps that are correctly detected.

– False Negative

Number of malware apps that are detected as malicious.

– True Negative

Number of malware apps that appear as genuine apps.

– False Positive

Number of genuine apps which appear as malicious apps.

– True positive rate (TPR):

Percentage of good ware applications correctly identified. (TP/(TP+FN))

– False positive rate (FPR):

Percentage of malware applications wrongly identified. (FP/(TN+FP))

– Overall accuracy (ACC):

Percentage of applications correctly identified.

TP +TN /TP+TN+FP+FN

Once the dataset is trained and tested by using the algorithm. Then it's time to use the tensorflow model which helps to deploy the project in the android version. Tensorflow Inference function is used for deploying purpose.

Malware detection app can be easily understand by parsing through the flowchart given below.

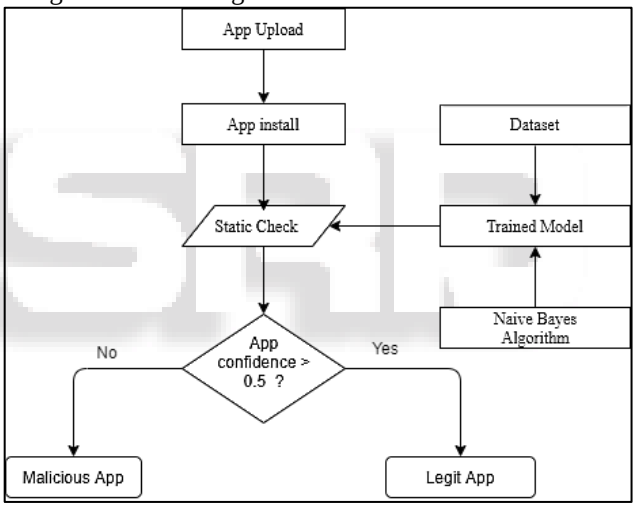

Fig. 3: Flowchart for Static Analysis of an Apps

ACKNOWLEDGE

I would like to express our gratitude to our advisers Prof. Ankit Sanghavi and Prof. Mayank Mangal. It was great to have this combination guiding my research as I had the best of both. Mayank sir helped me staying focused on problems and provided new directions to approach them, while Ankit Sir will explain the concepts at an abstract level, opening up the mind to apply the same techniques on various similar problems. Working with them, I have developed our problem solving skills, learnt about research, about life in general. I would be thankful to them for all my life, for providing me the guidance on various matters, always motivating and boosting us with confidence, which really helped me in shaping my life.

I want to take this opportunity to thank Prof. Mayank Mangal for not just teaching me courses like Principles of Information Security and System and Network Security, but also for providing me a wonderful opportunity

All rights reserved by *www.ijsrd.com*

for Principles of Information Security which gave me even more insight into the fundamentals of security than when I was just a student in the course.

V. CONCLUSION

As the technology is opening so many new methods for the attackers, we also need to utilize the same technology to implement counter methods to safeguard our privacy from the attackers. When anti-virus software makers started using signature analysis to find a malware, the attackers started creating a new signature to bypass such solution. This made such solution a less reliable. So the need for introducing a different solution which is more reliable, secure and efficient is very high is being used. That why the Machine Learning technique comes into play. Machine Learning is the future and in this system, the Machine-Learning technique will look for patterns in the program properties. At this project report static analysis implementation has been completed and further have implement dynamic analysis .This will be the base for differentiating between a malicious application and benign apps. But the problem will arise when the malware creator start attack using dynamic tactics which is difficult to catch. In future there must be open war between the attacker and the protector of android app attacker.

REFERENCE

[1] K. Shibija and R. V. Joseph, "A Machine Learning Approach to the Detection and Analysis of Android Malicious Apps," 2018 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, 2018, pp. 1-4.
[2] Harshvardhan.C.Takawale and Abhishek Thakur, "On-device Machine Learning Using TensorFlow to Detect Android Malware" published by Department of Electrical and Electronics Engineering , Department of Computer Science BITS Pilani, Hyderabad Campus Hyderabad, India, 03 December 2018.
[3] Monica Kumaran and Wenjia Li, "Lightweight Malware Detection based on Machine Learning Algorithms and the Android Manifest File", Electrical Engineering and Computer Sciences University of California, Berkeley Berkeley, U.S.A.Wenjia Li School of Engineering and Computing Sciences New York Institute of Technology New York, U.S.A, 08 February 2018.
[4] Ravi Kiran Varma P,Kotari Prudvi Raj ,K. V. Subba Raju, "Android mobile security by detecting and classification of malware based on permissions using machine learning algorithms" in International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2017), 05 October 2017.
[5] Westyarian ,Yusep Rosmansyah, Budiman Dabarsyah "Malware Detection on Android Smartphones using API Class and Machine Learning",Electrical Engineering Department, Institut Teknologi Bandung Jl. Indonesia,August 2015.
[6] Wei-Ling Chang and Hung-Min Sun, Taiwan ,Wei Wu,"An Android Behavior-Based Malware Detection Method using Machine Learning", Fujian Provincial Key Laboratory of Network Security and Cryptology, Fujian, China, 24 November 2016.
[7] Sagar Sabhadiy,Prof. Jaydeep Bara and Prof. Jaydeep Gheewal, "Android Malware Detection using Deep Learning" Third International Conference on Trends in Electronics and Informatics, 10 October 2019
[8] Radoniaina Andriatsimandefitra and Valerie Viet Triem Tong," Detection and Identification of Android Malware Based on Information Flow Monitoring", 2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing, 07 January 2016.
[9] Zinal D. Patel," Malware Detection in Android Operating System", International Conference on Advances in Computing, Communication Control and Networking, 01 July 2019.
[10] YU lu, PAN Zulie, LIU Jingju & SHEN Yi," Android malware detection technology based on improved Bayesian Classification", 2013 Third International Conference on Instrumentation, Measurement, Computer, Communication and Control, 23 June 2014.
[11] Tianliang Lu and Su Hou, "A Two-Layered Malware Detection Model Based on Permission for Android", 2018 IEEE International Conference on Computer and Communication Engineering Technology (CCET), 22 November 2018.
[12] Shahid Alam,Serdar Yildirim Mahamat Hassan and Ibrahim Sogukpinar, "Mining Dominance Tree of API Calls for Detecting Android Malware", 10 December 2018.
[13] Cangzhou Yuan, Shenhong Wei, Yutong Wang, You Yue and ShangGuan ZiLiang , "Android Applications Categorization Using Bayesian Classification", 2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 28 February 2017
[14] Jiawei Zhu, Zhengang Wu, Zhi Guan and Zhong Chen,"API Sequences based Malware Detection for Android", UIC-ATC-ScalCom-CBDCom-IoP 2015, 21 July 2016.
[15] Dong-Jie Wu, Ching-Hao Mao, Te-En Wei, Hahn-Ming Lee1 and Kuo-Ping Wu," DroidMat: Android Malware Detection through Manifest and API Calls Tracing", 2012 Seventh Asia Joint Conference on Information Security, 13 September 2012.

All rights reserved by *www.ijsrd.com*