Detection of Malicious Applications using Permission Based Retrieval

Chandrika Kapre[1], Aparna Harikumar[2], Rajeshwari Chandratre[3] and Dhaval S.Adhav[4]

[1] (BE COMPUTER) Department of computer engineering ,VIIT, Savitribai Phule University ,Pune , India
[1]chandrikakapre@gmail.com
[2] (BE COMPUTER) Department of computer engineering ,VIIT Savitribai Phule University ,Pune , India
[2]aparnahk16@gmail.com
[3] (BE COMPUTER) Department of computer engineering ,VIIT, Savitribai Phule University ,Pune , India
[3]rajeshwarichadratre@gmail.com
[4] (BE COMPUTER) Department of computer engineering ,VIIT Savitribai Phule University ,Pune , India
[4]dhaval12593@gmail.com

ABSTRACT

Smart phones play an important role in today's world as they help reduce man's efforts. The forthcoming open operating system will focus not on desktops system, but will also be on devices that we use every day (E.g. Mobile). The main objective of the project is to design and develop an effective algorithm to detect malicious applications that possess a threat to hand held devices (in Android platform). This mainly focuses on the Vulnerability of Android applications and its Permissions. For example critical data may be manipulated using hand held devices for Banking etc. Since Android is an open source platform a lot of third party applications are developed and uploaded on Android market and Third party application stores. Users download it via their devices and usually applications are downloaded in bulk from the Play store. The main problem is that users are unaware of the third party applications which work as a simple application but may contain malicious application in it. The proposed algorithm realizes a malware detection system that continuously monitors various features and events obtained from the device. In case the application is declared as a malware by the algorithm then it is prevented. The proposed approach simply detects malware based on the patterns of already known malwares. The new environment will lead to new applications in their markets to enable greater integration. The proposed work not only checks permissions but also involves comparison with threshold values and declarations of why the application is malicious. The results suggest that the proposed work is effective in detecting malware on Android devices. Thus as a result we get a clear idea about which applications are using features which they are not supposed to use or they do not need to use.

Keywords: Android, malicious, malware, mobile device

1. INTRODUCTION

Earlier mobile phones were simple devices performing basic phone functionalities like call, send messages, receive messages, by the introduction of new smartphones having own operating system, mobile phones began to include advanced features like full-featured web browsers, Internet Connectivity, multimedia capabilities and desktop which caused inexperienced users and made application developers to think differently about mobile devices. For entertainment purpose additional applications such as games, productivity and communications are developed by third-party developers. This developed applications can be placed directly on Android market, while using Android based phone for first time, it requires application to be signed once. Users then can

download any application from anywhere including Google playstore. There have been plenty of malicious applications in Android; most of these applications are available in Third party application store and in markets other than Android. The Android market is open for all developers for their easy use; this constraint also allows ease of entry to malware developers. Android Developers utilize a comprehensive SDK (Software Development Kit), with ample tools for development of powerful, accurate, rich featured applications. The basic architecture diagram is given in Figure 1. It is type of layered architecture, the first layer Application involve applications with which user interact or work. The Framework involves the communication of applications with activity. The Libraries SQL includes data storage, web kit contains viewing HTML page, and Android is built in Linux Kernel[1].

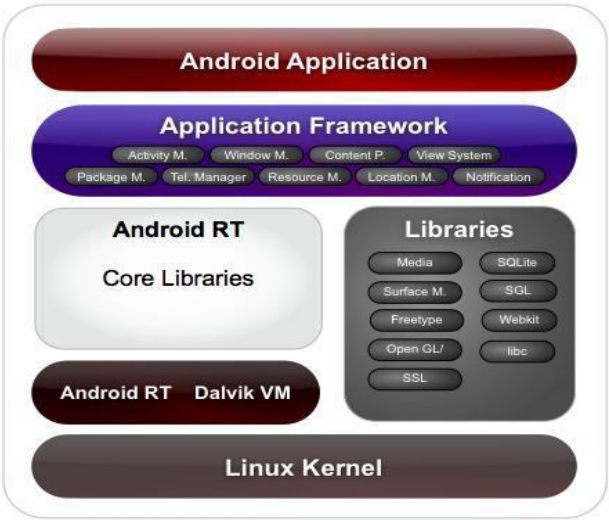

Fig -1: Android Architecture (source internet)

2. LITURATURE SURVEY

In Shadow Manifest[7] permission that an app requires are stored by prior execution of the app.Unecessary permissions are stored with a mask which are to be revoked by generating an empty resource when an app requests them. COPES(Correct Permission Set)[6] is a tool which uses static analysis to extract a table from from Android framework bytecode.This table the set of permissions that an app needs and maps every method of the API to these permissions which are called.So no unnecessary permissions are stored in the table and mapped with API methods. In Apex[5] users are allowed to specify what an app can access.An extended installer is used to set user policies.

3. MOBILE MALWARE

The Mobile phones are different than the conventional desktop, in mobile devices/handheld devices, there are limited resources in terms of memory,power and energy consumption mainly security.A malicious (or) malware application targets mainly these weaknesses. Android is an open source operating system. Android developers can upload their applications on the Android market. Android phone requires signature for authentication mainly Google will be using these signature as authentication of users and that's why android users able to download application not only from the Android market, but from any other application store. An application in Android is said to be malicious on the basis of permissions used by that applications which are not supposed to be and comments and rating from user on the application as well . The database will maintain this information like how many users downloaded the application and reported in their comments. If more number of users suggests negative aspect on a particular application. Till today for android users, the only way to secure devices from malicious application is to download application in Android market, not in any other app-store and also before downloading the application, the users need to check how many times the application have been downloaded, with positive comments. Since Android market's openness has gain which provides easy way to the developers organized around the world, and has loss like allowing entry to malware developers.

4. ARCHITECTURE DIAGRAM

The Proposed architecture diagram follows the basis of Android Architecture in which naive users works in Android Mobile with the help of Graphical User Interface(GUI), user can download Android application not only from Android market, but also from others since it uses signature for authentication. During downloading the application, the user is unaware whether the app is malicious.

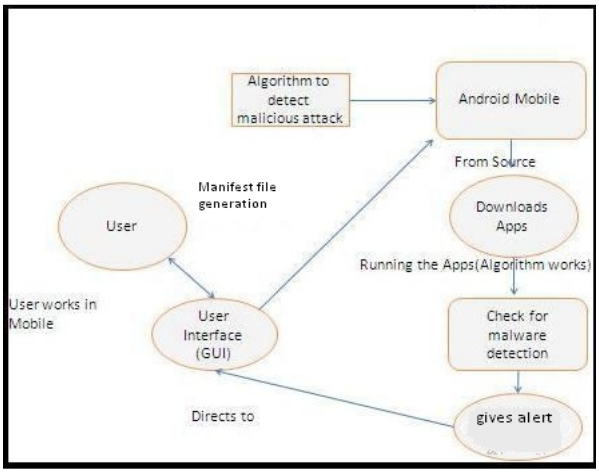

Fig. 2 Architecture Diagram for Proposed work

Here our proposed algorithm checks keywords occurance in manifest files ,comparing with known malicious keywords stored in database .Each comparision adds to the count .Then average is calculated based on count per total number of words.If the average value for features goes beyond the predefined threshold then alarm is given to the user that this application is suspended to be malicious.

Malicious Attack Illustration

Developer takes legitimate application and repackages it with malware using Android manifest file with .xml or .txt or any other extension. As discussed in Figure 3 Malicious Developer uploads Application on third party application store; Malicious Developer can control the androide device remotely and access user's private information like view the location, contact information, send and read sms, place phone calls etc., Such an
application is said to be malicious.

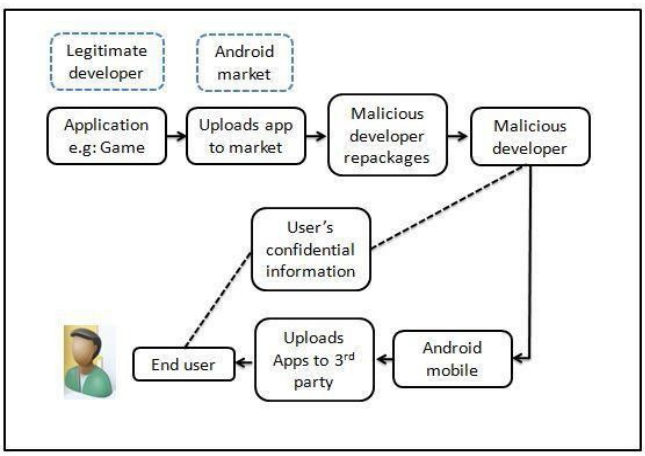

Fig -3: Malicious Attack Illustration [1]

5. ALGORITHM

1. Read Manifest File
   1.1 Search for manifest file
   1.2 Read file into string
2. Preprocess the string to convert it into best form
2.1 Remove stopwords and perform stemming
3.Identify all the topwords using the vector form of the strings
   3.1 Add words in Vector and for every word of vector search for count and sort in descending order
4. Identify all malicious keywords
   4.1 compare the words in the string vector with the keyword vector obtained from the database containing malicious keywords
5.Calculate the threshold value for application being scanned.
6.Compare threshold value with malicious threshold value and judge if its malicious or not.

6. CONCLUSION AND FUTURE WORK

Android is a widely used and anticipated open source operating system, since it has been introduced and still explored for its security mechanisms. There are many solutions which were proposed to prevent malware applications invading in our. Many applications are developed for mobile devices, since critical data is manipulated in areas as E-Banking, E-billing etc. in which transactions are involved; there is a need for security in using these applications through mobile devices. In our proposed work to detect application that cause malicious threat in mobile devices using algorithm which gives alert on detection. Our future work of Android malware detection tool is targeted in two different directions;

First, exploration of all features during installation itself and if detected malware then automatically stopping the installation. Overcoming crashing of system due to thread synchronization problem.

Second, the dynamic analysis on the malware system, which includes detecting, changes in the device which persists for a long time and also new malwares detection which are not previously known by using limited resources.

REFERENCES

[1] Saranya .T, Shalini .A.P., Kanchana .A, International Journal of Innovative Research in Computer and Communication Engineering(An ISO 3297: 2007 Certified Organization), Detection and Prevention for Malicious Attacks forAnonymous Apps ,Vol. 2, Issue 3, March 2014,

[2] Alexandre Bartel, Jacques Klein, Member, "Static Analysis for Extracting Permission Checks of a Large Scale Framework: The Challenges and Solutions for Analyzing Android", VOL. 40,NO. 6, JUNE 2014

[3] Bilal Shebaro, Oyindamola Oluwatimi,Elisa Bertino, "Context-based Access Control Systems for Mobile Devices", IEEE Transactions on Dependable and Secure Computing, 2014

[4] Aditi Tripathy, Prof.G.P. Potdar, "A Framework for Providing Selective Permissions to Android Applications", IOSR Journal of Computer Engineering (IOSR-JCE), Volume 13, Issue 3 (Jul. - Aug. 2013), PP 53-58

[5] Mohammad Nauman and Sohail Khan., " Design and Implementation of a Fine-grained Resource Usage Model for the Android Platform", The International Arab Journal of Information Technology, Vol.8, No.4, Oct 2011

[6] Alexandre Bartel,Jacques Klein,Martin Monperrous, "AutomaticallSecuring Permission-Based Software by Reducing the Attack Surface:An Application to Android".

[7] Lakhmi Priya Sekar,Vinitha Reddy Gankidi, Selvakumar Subramanian, "Avoidance of Security Breach through Selective Permissions in Android Operating System", ACM SIGSOFT Software Engineering Notes,September 2012 Volume 37 Number 5.

