International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 4 Issue: 3
ISSN: 2454-4248
52 – 56

A Survey on Mobile App Security

ER. Mani Goyal
Department of Computer Science
Maharishi Markendeshwer University
Ambala , India
er.mani.goyal@gmail.com

Dr. Avinash Sharma
Department of Computer Science
Maharishi Markendeshwer University
Ambala, India
sh_avinash@yahoo.com

***Abstract*—** Today's era a large number of Android apps appearing, personalization functions have been the focus of the apps' competition. However, the ongoing implementations of personalization mainly rely on gathering and analyzing users' sensitive information, which may not only break the good eco-friendly environment but also threaten users' privacy security. The smart phones industries are reaching to a new level because of these Android applications. Smart phone users can perform their everyday tasks by using the different types of applications provided by the Android Play Store. Generally, before the installation of the Android apps, users have to agree on the permissions which are requested by the apps, they are not given any other option. Essentially, users may not conscious on some security issues that may appear from the permissions. Some apps request the right to operate sensitive data, such as photos, GPS location, contact, calendar, email and files. The objective of this survey paper is to study and analyze the different techniques which are used to enhance the security of Android.

Keywords-*Android, eco-friendly, smart phones.*

*****

I. INTRODUCTION

Cell phones have increased enormous notoriety in the course of the most recent couple of years. In this developing business sector of cell phones, Android, an open source stage of Google has gotten to be a standout amongst the most mainstream Operating Systems. Android is for the most part utilized as a part of cell phones and tablets.

Cell phones are acknowledged and respected by numerous predominantly on the grounds that they are prepared to do giving administrations, for example, keeping money, person to person communication and so on all on the go. They are outfitted with a few elements, for example, Wi-Fi, voice, information, GPS, and so forth

The sudden increment in cell phone applications causes worry as far as client security. Cell phones have turned into a delicate and defenseless focus for noxious application

The android engineering is comprised of the accompanying four layers (3):-

- A Linux Kernel that backings multiprocesses and multithreads. Each application has its own Linux ID and keeps running in a different procedure. Two applications with a similar ID can trade information between them.
- Some Open source libraries.
- Android run-time environment, wherein a Dalvik Virtual Machine runs an applications in the dex combined association.
- An application system that has a Java interface. This layer comprises of the Android NDK and SDK.

The android security model is based on sandbox mechanism and permissions. Each application runs in its own Dalvik Virtual Machine with a distinctive ID allocate to them. This prevents an application from hampering information/data of another application.

In spite of the fact that Android is most generally used, there exists a lack of applications in order to entirely benefit from this operating system. Hence new applications are created by the third party application developers and launch them in the Android Market. This gives users retrieve to thousands of applications. It is however important that before installing the applications the user needs to entirely trust the applications. It is for this reason that every application publishes the permissions that it needs during installation. Even the user can either grant all permissions or reject all, in this case, the installation of the application is aborted.

So as to disperse these applications Google thought of Android Market. Here clients can get to both paid and free applications. Each Android telephone has this application and thus clients can peruse and download any application they require from Android Market.

Android is an open source operating system that has attained a great deal of importance and popularity in the mobile phone world. This operating system is powered by Google and study has shown that it has now become the world's second most popular operating system.

The most appealing element of Android is that it gives an open stage to designers to make their own applications. Android gives a SDK and NDK to the engineers to make different applications not at all like the Apple iphone applications that should be downloaded from the Apple Appstore.

The openness of this operating system is both an benefit as well as a hinderance.Android permits designers to effectively distribute their application however it likewise takes into account the publication of pernicious applications.

 ISSN: 2454-4248
52 – 56

Let us now discuss the architecture of Android. It is basically a software stack that contains three components, namely, Middleware, Operating System and Key Applications.
Its features include :-

- Bluetooth, EDGE, WiFi and 3G.
- SQLite for structured data storage.
- Application Framework
- Dalvik Virtual Machine, etc.

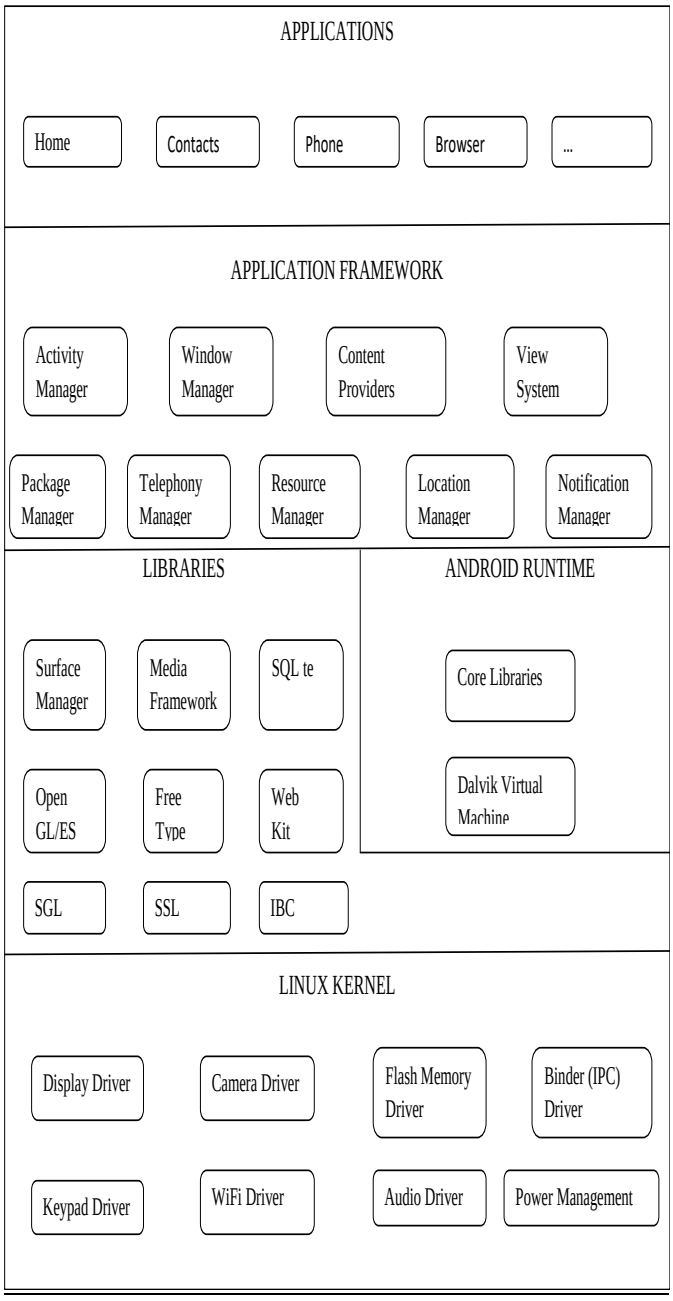

Fig:1Architecture of Android operating system

Android architecture consists of the following:-

- Application :- Android phones usually come with some default applications such as email client, browser, calendar, SMS, maps, etc. The applications are programmed using Java.
- Application Framework :- Android developers are offered the utilization of information of access location, set alarms, device hardware, run background services, etc. Android developers also have access to the framework APIs that is also used by the core applications. Reuse of components is exercised by the application architecture design.

All applications have a rendezvous of frameworks and administrations basic them:-

- Views:- These are accustomed to build applications, such as, text boxes, grids, buttons, and additionally an internet browser that's embedded.
- Content Providers:- These permit applications to share their own knowledge and to access into from different applications.
- Resource Manager:- This provides access to resources (non-code) like graphics, layout files and localized strings.
- Notification Manager:- Custom alerts are shown on the status bar via a notification manager.
- Activity Manager:- It provides a standard navigation backstack and it additionally deals with the lifecycle of an application.

- **LIBRARIES:-** Android is prepared with a group of C/C++ libraries that is employed by several elements of the system. These libraries are provided to the designers by the Android Application Framework.

Some of the core libraries along side with their practicality is shown below:-

- System C Library:- It's used for embedded Linux-based devices. It is a BSD derived implementation of the C system Library libc.
- Media Libraries:- These libraries basically support recording of audio and video formats, playback and static image files, such as, JPG, AMR, MP3, PNG, AAC, MPEG4 and H.264.
- SGL:- This comprises the 2D graphics engine.
- SQLite:- This is a relational database engine that is accessible to any or all applications.
- Surface Manager:- it controls the access to the 2D and 3D graphics layers from varied applications and to the show subsystem.

- **Linux Kernel :-** Android services, like process management, driver model, memory management, security and network security, of the core system

 ISSN: 2454-4248
52 – 56

depends on Linux version 2.6. The kernel also behaves as a layer of abstraction between the software stack and the hardware.

1.1 Android application framework

An android application consists of various components that use Intent messages to communicate with one another. These components are summarized below :-

- Activity :- It is the visual interface that is utilized by the user in order to process actions.
- Broadcast Receiver :- This component receives and reacts to broadcast announcements/messages by intiating an Activity. It has no user interface.
- Service :- This component has no user interface, but runs in the background for an imprecise period of time.
- Content Provider :- In order for an application to make data available to other applications, it makes use of a content provider that is a type of database SQL Database).

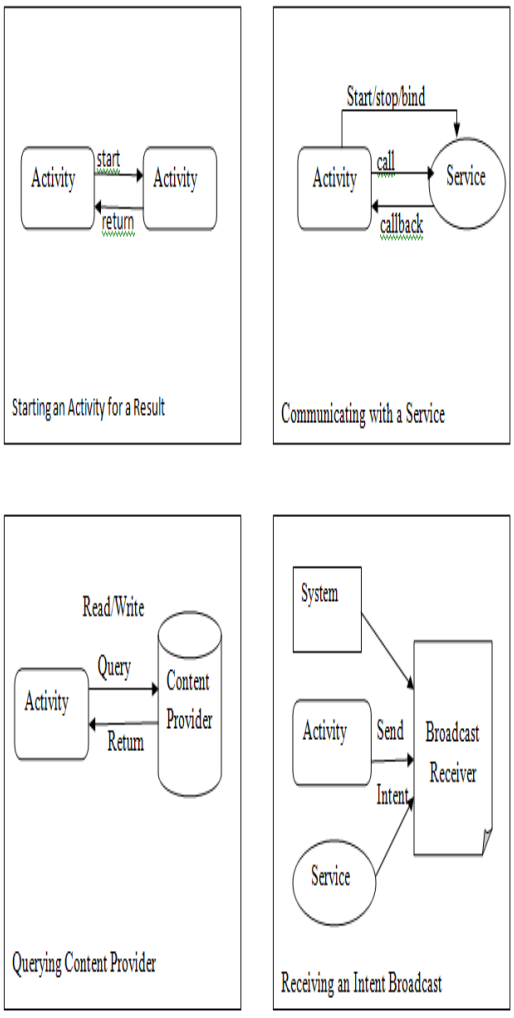

Fig:2 ICC between different components.

II. RELATED WORK

2.1 Chit La Pyae Myo Hein(2014) presented the permission based malware protection model, then use the self-organizing feature map algorithm for Android applications. This is express to make small consequent adjustments of the protection level and to raise the accuracy of the android permissions. Only manifest files is used to detect malware. Manifestfiles are necessary in all Android applications, and therefore, the proposed method is applicable to all Android applications. Empirically analyzing permission-based models and using the Self- Organizing Map (SOM) algorithm. Testing and training applications are then passed via the behavior based module for identification of android permission model.

2.2 Lydia Kraus, Ina Wechsung(2014) introduced an approach to provide users with additional information in form of statistical data about the number of app permissions in relation to other apps with similar functionality. The goal was to help users to understand permission requests easily, to increase awareness regarding the p ermission issue, and to include the number of permissions in the decision-making process.

2.3 Abdullah Mohammed Rashid and Ali Taha Al-Oqaily(2015) discussed the techniques to competently detect and prevent the mobile malware and suggest an improvement towards existing techniques which gives better mobile malware detection and prevention. The main contribution of the proposed solution is to generate a new model, method and technique to detect and prevent malware through a combination of cloud service and dynamic analysis.

2.4 Hamid Bagheri, Alireza Sadeghi et. al (2015) presents a novel approach for automatic synthesis and enforcement of security policies, allowing the end-users to safeguard the apps installed on their device from inter-app vulnerabilities. This approach allows the end-users to protect a given pack of apps installed on their device from such attacks. The approach, achieve in a tool, called DROIDGUARD, merging static code analysis with lightweight recognized methods to automatically understand security-related properties from a pack of apps.

2.5 Vinothini.S (2015) provides a survey about, the proposed system privacy and security on smartphone applications for the entire smartphone user. This is achieved by using a policy-based framework for enforcing software isolation of data and applications on the Android platform. It describes different security profiles within a single smartphone. Dynamic switching from one security profile to any another profile will access and control to applications and data. Even this approach provides compartments where data and apps

 ISSN: 2454-4248
52 – 56

are stored. These compartments are called Security and Privacy Profiles. A security and privacy profile is a set of policies that regulates what applications can be implemented and what data can be retrieved.

2.6 T.Nandhini and V.Arulmozhi (2015) introduced tracking and monitoring of malicious activity of the apps that are installed by the user from playstore using trusted permission based security model. The proposed framework identifies the apps installed and its behavior according to the permission granted upon installation. This real time tracking framework monitors the installed apps for any violation in the permissions agreed.

2.7 Zhenlong Yuan, Yongqiang Lu(2016) proposed to combine the features from the static analysis with the features from dynamic analysis of Android apps and distinguish malware using deep learning techniques. An online deep-learning-based Android malware detection engine (DroidDetector) is implemented that can automatically identify whether an app is a malware or not.

III. SUMMARY OF VARIOUS PURPOSED TECHNIQUES

The different purposed techniques along with their finding which are discussed above are compared in the following table1 as :

Author & year	Technique Used	Findings
Chit La Pyae Myo Hein 2014	Self organizing feature map algorithm	permission levels of android a small number permissions are very frequently used and a large number of permissions are only occasionally used and show that it can achieve high accuracy rate.
Lydia Kraus Ina Wechsung Sebastian Möller 2014	Cochran's Q test repeated-measure ANOVA, with UI and permission level (low and high) as within-factors	users tend to choose more often the app with a lower number of permissions. the privacy-intrusiveness and trustworthiness of apps is perceived differently when statistical information is given.
• Abdullah Mohammed Rashid & Ali Taha Al-Oqaily 2015	cloud service, model-based stage, static and dynamic analysis and power consumption observation.	produced a new model, method and technique to detect and prevent malware through a combination of cloud service and dynamic analysis.
Hamid Bagheri Alireza Sadeghi Reyhaneh Jabbarvand Sam Malek 2015	DROIDGUARD	a novel approach for automatic synthesis and enforcement of security policies, allowing the end-users to safeguard the apps installed on their device from inter-app vulnerabilities.
Vinothini.S 2015	a policy-based framework for enforcing software isolation of applications and data on the Android platform	security and privacy profile is a set of policies that regulates what applications can be executed and what data can be accessed
T.Nandhini and V.Arulmozhi 2015	trusted permission based security model.	real time tracking framework monitors the installed apps for any violation in the permissions
Zhenlong Yuan, Yongqiang Lu, and Yibo Xue 2016	DroidDetector	By using DroidDetector with a deep learning model canachieve a superior accuracy under different conditions,significantly outperforming traditional machine learning techniques.

IV. CONCLUSION & FUTURE SCOPE

In this survey we studied the different techniques that can enhance the security on the Android apps. The permission based malware protection model for android application is presented then use the self-organizing feature map algorithm to make small subsequent adjustments of the protection level and to improve the accuracy of the android permissions. Droid Detector with a deep learning model is used to achieve a superior accuracy under different conditions, significantly

 ISSN: 2454-4248
52 – 56

outperforming traditional machine learning techniques. Even the malicious activities of the apps that are installed by the user from the Playstore are tracked and monitored by using trusted permission based security model.

V. FUTURE WORK

A further analysis is to determine how much percent of security is enhanced after injected the wrapper in the APK installer file and resizing the advertising banners to zero.

VI. REFERENCES

[1] Wook Shin and Sanghoon Kwak, "A Small but Non-negligible Flaw in the Android Permission Scheme" 978-0-7695-4238-6/10 $26.00 © 2010 IEEE, pp. 107-110.

[2] Wook Shin, Shinsaku Kiyomoto, Kazuhide Fukushima, and Toshiaki Tanaka," A Formal Model to Analyze the Permission Authorization and Enforcement in the Android Framework" 978-0-7695-4211-9/10 $26.00 © 2010 IEEE, pp. 944-951.

[3] Erika Chin, Adrienne Porter Felt, Kate Greenwood and David Wagner, "Analyzing Inter-Application Communication in Android" June 28-July 1, 2011.

[4] Alexandre Bartel, Jacques Klein, Yves Le Traon and Martin Monperrus, "Automatically Securing Permission-Based Software by Reducing the Attack Surface: An Application to Android", September 3-7, 2012.

[5] Kathy Wain Yee Au, Yi Fan Zhou, Zhen Huang and David Lie, "PScout: Analyzing the Android Permission Specification" October 16-18, 2012.

[6] Chit La Pyae Myo Hein, "Permission Based Malware Protection Model for Android Application", vol.2, 2014, pp. 222-226.

[7] Lydia Kraus, Ina Wechsung, Sebastian Möller Quality and Usability lab, "Using Statistical Information to Communicate Android Permission Risks to Users" 978-1-4799-7901-1/14 $31.00 © 2014 IEEE, pp. 48-55.

[8] Abdullah Mohammed Rashid and Ali Taha Al-Oqaily, "Detect and Prevent the Mobile Malware" Volume 5, Issue 5, May 2015, pp. 1-3.

[9] Hamid Bagheri, Alireza Sadeghi, Reyhaneh Jabbarvand and Sam Malek, "Automated Dynamic Enforcement of Synthesized Security Policies in Android" GMU-CS-TR-2015-5, pp. 1-15.

[10] Vinothini.S, "Survey on Policy Based Framework for Smartphone Application's Privacy Using Multiple Profiles" Vol.3, Issue 3, March 2015, pp. 1619-1621.

[11] T.Nandhini and V.Arulmozhi, "Permission Tracking Security Model in Android Application" AJCST Vol.4 No.2 July-December 2015, pp. 6-12.

[12] Zhenlong Yuan, Yongqiang Lu, and Yibo Xue, "DroidDetector: Android Malware Characterization and Detection Using Deep Learning" Vol. 21, No. 1, 2016, pp. 114-123.