Unmasking & Forfending Malware Techniques In Android Operating System

Komal Patel[1], Gayatri Pandi[2]

[1] P.G Student, Department of Computer Engineering, L.J.I.E.T, Ahmedabad, Gujarat, India
[2] Head of Department, PG Department, L.J.I.E.T, Ahmedabad, Gujarat, India

ABSTRACT

The decision cell phone stage these days is an Android. The open source highlight of these working framework prompt malware makers to presented new malware every day and are staying away from the likelihood of unmasking and forfending the malevolent code accessible in the product or application and control clients' private data without their insight. Malwares are made to present a wide range of cybercrimes, for example, abuse of administrations, sucks the data, and root or part access of the gadget. There are a few strategies are acquainted by a wide range of research with keep away from this circumstance by unmasking such vindictive movement and forfend future dangers of malware and give a protected domain to the Android PDA clients. The paper talks about the kind of malware and strategies to unmask and forfend their action.

Keyword: - *Android Architecture, Malware, Security, Malicious code, Monitor.*

1. Introduction

Android is most popular mobile operating system in 2015-16. According IDC (International Data Corporation), Second half of 2015 average 81.95% market share and in first half of 2016 average 85.5% market share for the Android operating system. Android popularity has encouraged the developers to provide unique and attractive applications which make users‟ life simpler. These applications are widely known as „apps‟.
The different versions of Android are shown in below figure.

Fig 1.1: Different Versions Of Android Operating System. [7]

The official Android app market is Google Play Store which hosts the third party developer apps for a nominal fee. It has more than a million apps and very big range of applications get downloaded every day. Google Play does not verify the uploaded third party apps manually. Therefore Malware app developers have total smartphone control by root permissions, gather private and/or confidential user information, to extract monetary benefits by exploiting the telephony services or creating botnet.

Some of very unique feature of Android mobile operating system are Beautiful UI, Connectivity, Storage, Media support, Messaging, Web browser, Multi-touch, Multi-tasking, Multi-tasking, Resizable widgets, Multi-Language, GCM, Wi-Fi Direct and Android Beam.

The platform Architecture of Android operating system is as follows:

Linux Kernel:

Linux kernel is core of Android operating system. For example, the Android Runtime (ART) based on the Linux kernel for several functionalities to be used.

Hardware Abstraction Layer (HAL):

The hardware abstraction layer (HAL) connects interfaces that expose device hardware capabilities to the higher-level Java API framework. Multiple library modules provided by this layer, which implements an interface. Hardware component get loads when a call is made by framework API to access the hardware device.

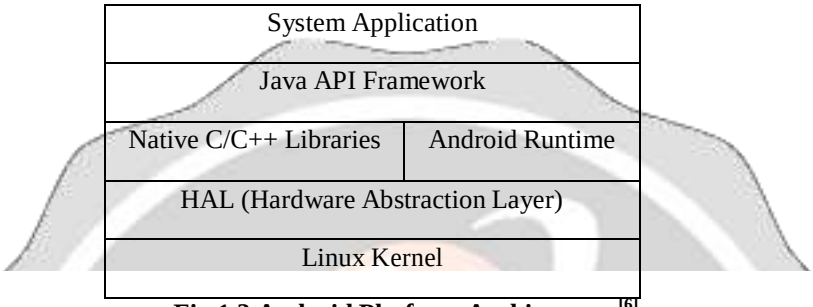


Fig 1.3 Android Platform Architecture [6]

Android Runtime:

Some of the major features of this are:

Ahead-of-time (AOT) and just-in-time (JIT) compilation

Optimized garbage collection (GC)

Better debugging support

Native C/C++ Libraries:

ART and HAL are built from native code that requires native libraries written in C and C++. The Android platform provides Java framework APIs to use these native libraries to applications.

Java API Framework:

The feature-set of the Android OS is provided through APIs written in the Java language. These APIs form the building blocks you need to create Android apps by simplifying the reuse of core, modular system components and services, which include the following:

- A rich and extensible View System you can use to build an app"s UI, including lists, grids, text boxes, buttons, and even an embeddable web browser
- A Resource Manager, providing access to non-code resources such as localized strings, graphics, and layout files
- A Notification Manager that enables all apps to display custom alerts in the status bar
- An Activity Manager that manages the lifecycle of apps and provides a common navigation back stack
- Content Providers that enable apps to access data from other apps, such as the Contacts app, or to share their own data

System Apps:

Android comes with a set of core apps for email, SMS messaging, calendars, internet browsing, contacts, and more. Apps included with the platform have no special status among the apps the user chooses to install. So a third-party app can become the user's default web browser, SMS messenger, or even the default keyboard. The system apps function both as apps for users and to provide key capabilities that developers can access from their own app.

1.1 Malware

Malware is kind of software that harm & infect the computer system or mobile system. Each day new kind of malware are introduced thus to unmasking and forfending techniques should be adoptive to the stubborn nature of malwares.

Types of Malware:

Malware can be mainly classified in five types:

Virus is type of virus that can able to spread it; it is a piece of code and affects other software.

Adware is type of malware based on advertising software and websites. It generates revenue for its writer, also track users internet logs.

Spyware is most difficult malware to detect. It spies user, track their activities in the system, gather important and confidential information and manipulate according to requirements.

Worms are replicating it. They are able to access root and destroy information, files or even operating system itself.

Ransomware is the most advanced malware till now. It locks your files, data or PC and extorts money from you in order to provide access to collect funds for their legitimate activities in the web.

2. Literature Survey

In "A detection method for malicious codes in android apps" paper, the malicious activity is tried to eliminate by the combination of two approaches that is static and dynamic approaches. In static approach the targeted software is analyzed first. It will check if the software containing any known signature that is already defined then it will be added to white list else it will added to black list and by using behavior based techniques the new signature are added to the database. Now the APK file will be decompile and keyword match algorithm applied and malicious code found that it will be replaced and track the data flow. In dynamic approach the newly defined signature were re-packed with the file and modified kernel will be loaded. Now it will analyses the log. [1]

In "Dissecting SMS Malwares in Android" paper, a backup activity is created and it starts the scheduler. When the filter mode is enable, which means it is filtering messages then it sends SMS to premier numbers. Some filter time is also set for which filter mode will be enabled during which other method is created to mute the phone will be performed and it also delete the notification of charges to the user and aldo delete the SMS from the messaging application. Once the scheduler time is finished, the phone will be back to general mode and abort the request for SMS sent by malicious code. [2]

In "Detection and Identification of Android Malware Based on Information Flow Monitoring" paper, the novel approach defined named as AndroBlare, which knows the data flow between files, objects and in Operating system. It uses hooks to intercept the SysCall and is also able to control such call. AndroBlare track the information flow and update the taint whenever it detects the new data flow in the system. It is also able to observe the Kernel logs so that even application with root permission can"t act any malicious activity. [3]

"Detecting and tracing leaked private phone number data in Android Smartphone" paper is based on a service „safe phone number" which is virtual private data based service. Here, the proxy server for each phone is created which virtualized the users" telephone number data. So whenever a suspicious application refers the contacts number using some malicious code it will referring the virtual number and thus the application is being detected and announced as malicious. It provides two mechanism with which the solution can be provided. During SMS phishing the sender get notification whether to send the data to receiver or not and second is receive get the notification that it can be malicious whether he/she wants to receive it or abort it. [4]

In "Prevention Mechanism for Prohibiting SMS Malware Attack on Android Smartphone" paper, basically two android applications are created to resolve the issue of SMS malware. First application is Sudoku application containing some malicious code and another one is monitoring application which monitors the activity performed in the smartphone. The Sudoku app asks for the permission to access the contacts and inbox SMS. Now it combines the information and sends it to C&C server. As the server receives the contacts it will send SMS to premier number which cost the user and generate revenue for the attacker. Monitor application display all running applications and background processes and then checks whether the SMS coming from legitimate app or not. It no, then it will alert the user and user will decide whether to allow to send the data or decline it. [5]

3. Proposed System

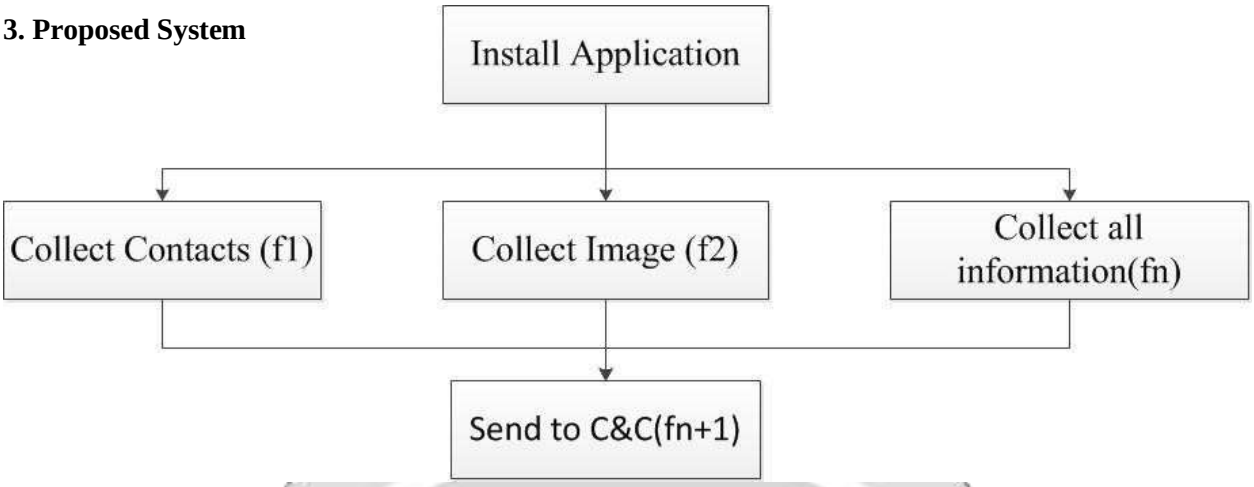


Fig 3.1: Proposed Malicious App

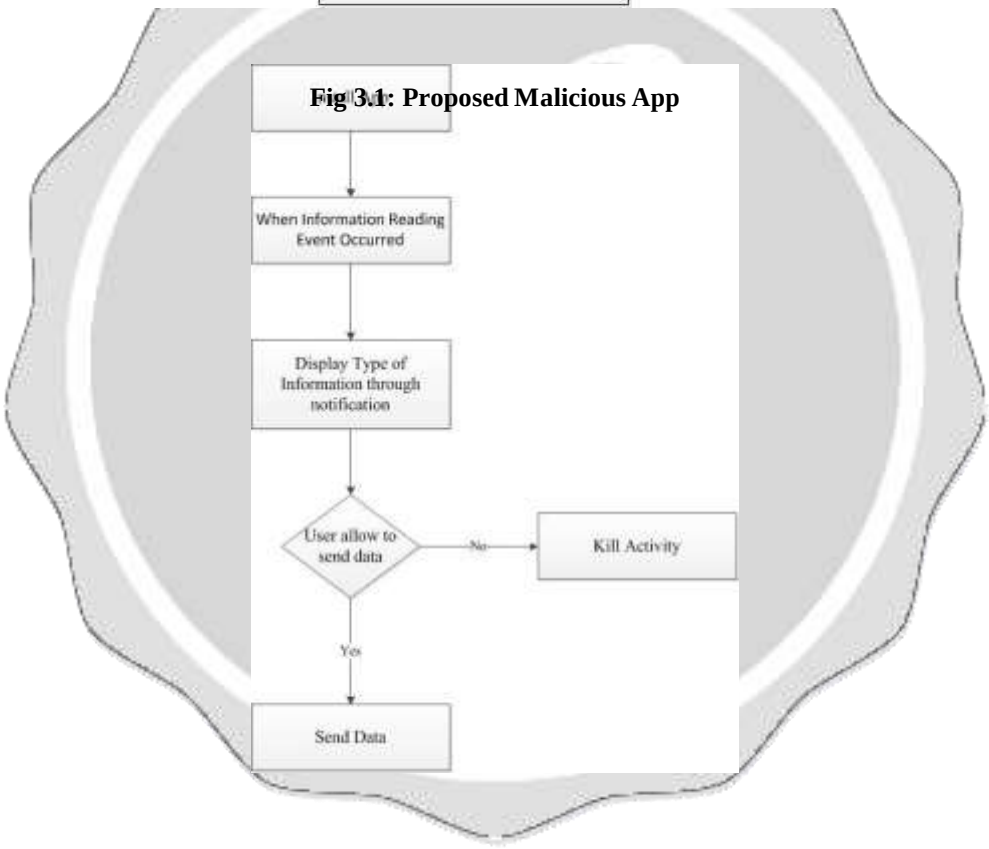


Fig 3.2: Proposed Monitor App

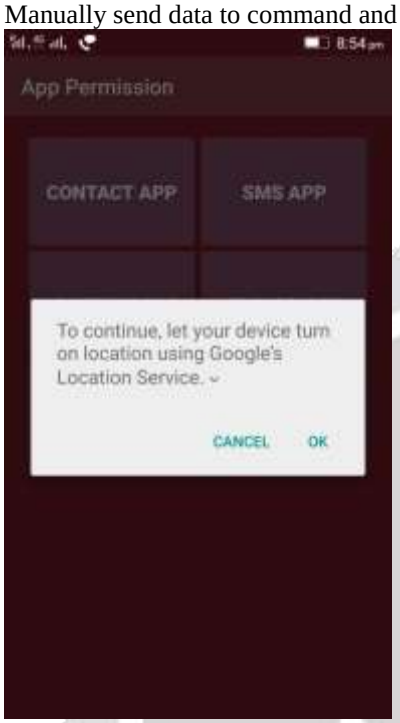
Firstly, an application made with some attractive features which leads to downloads by smartphone users. In the background such application containing some malicious codes, which is capable of reading user's confidential information stored on their cell phone such as the contacts of users, SMS messages, location of user, and so on. These information must be visible to only owner of the smartphone and should not be manipulated by any third party application. These application basically collect confidential information from the device and send it to some command & control server available in wireless network.

Second application is monitor application that monitor all the events that read some data from the device. Whenever some background activity or background service read the confidential data of the user it display type of information being read and trying to send to some command & control server available in the wireless network. This app does not allow such activity and notify user about it. If user allow such task in some special cases, when user needs automatic backup or something, it allow to send data, else user does not allow and app kill such activity.

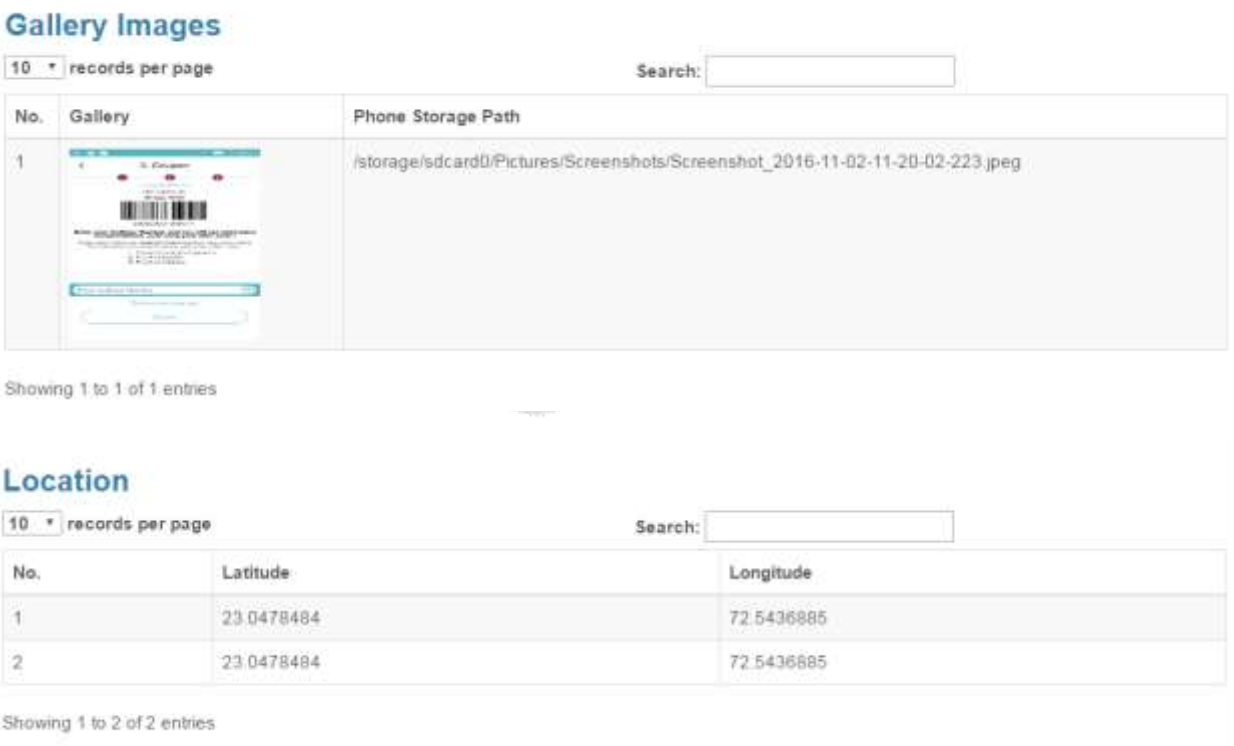
4. Implementation

❖ Implementation scenario:

1. Install AppPermission application that containing malicious code.
2. Manually send data to command and control server.



3. Data uploaded to server.



4. Install MonitorApp application that monitors the activity by malware.

5. After Installation whenever data is being uploaded to server, it ask user's permission whether it allow uploading or not.

6. If user allows, data will be uploaded.
7. If user denies, data will not be allowed to send out to android environment.

4. Conclusion

The current smartphone security models encourage components and procedures controlling the installation and execution of outsider applications. All things being equal, the adequacy of the received security components is by all accounts questionable. Their capacity to ensure the clients' information is enormous concerned. The proposed model prove an unique mechanism that allow user to look at the data that are being read by some mechanism or some malicious code in the application, and allow to forfend the task performed by malicious code exist in several application by notify the user about the type of information and thus it can be unmask such task before it perform.

5. References

1. Liu, Jinxin, Hao Wu, and Huabin Wang. "A detection method for malicious codes in Android apps." In *Wireless Communications, Networking and Mobile Computing (WiCOM 2014), 10th International Conference on*, pp. 514-519. IET, 2014

2. Babu, Anoop Joseph, Rahul Raveendranath, Venkiteswaran Rajamani, and Soumya Kanti Datta. "Dissecting SMS malwares in android." In *Contemporary Computing and Informatics (IC3I), 2014 International Conference on*, pp. 1065-1069. IEEE, 2014

3. Radoniaina Andriatsimandefitra, Val´erie Viet Triem Tong, "Detection and Identification of Android Malware Based on Information Flow Monitoring" *In 2nd International Conference on Cyber Security and Cloud Computing*, Pg: 200-203, IEEE, 2015, DOI:10.1109/CSCloud.2015.27

4. Wooguil Pak, Youngrok Cha, Sunki Yeo, "Detecting and tracing leaked private phone number data in Android smartphones",*31st International Conference On Information Networking(ICOIN)*, Pg:503-508, IEEE, 2015, DOI:10.1109/ICOIN.2015.7057956

5. Kotkar, Chetan, and Pravin Game. "Prevention mechanism for prohibiting SMS malware attack on android smartphone." In *2015 Annual IEEE India Conference (INDICON)*, pp. 1-5. IEEE, 2015

6. https://developer.android.com/guide/platform/index.html?hl=nn accessed on dec 2 at 2.00 pm

7. https://www.google.co.in/imgres?imgurl=https%3A%2F%2F4.bp.blogspot.cm%2FgJgio2rWQGA%2FV8WpJZ9HXcI%2FAAAAAAAAACSg%2F24jB0Qbej7sg_volhVsb4G092_UEfTwCLcB%2Fs1600%2Fandroid_nougat_new_android_addition.jpg&imgrefurl=http%3A%2F%2Fshaourhaider.blogspot.com%2F2016%2F08%2Fhavenougatandroidn.html&docid=s4UZwWGRXjCU5M&tbnid=8xxv3dJemrUdM%3A&vet=1&w=729&h=400&bih=662&biw=1366&ved=0ahUKEwiwoYLt5uHQAhWCGpQKHf0hAIUQMwgiKAYwBg&iact=mrc&uact=8 accessed on dec 2, 1:00pm

8. https://www.tutorialspoint.com/android/android_overview.htm accessed on dec 2, 1.30 pm