Detection of Threats in Mobile Apps: A Survey

Ms. Supriya Mandhare
Atharva College of Engineering,
Malad, Mumbai 95, India

Ms. Snehal Kathale
Atharva College of Engineering,
Malad, Mumbai 95, India

Ms. Chanda Chouhan
Atharva College of Engineering,
Malad, Mumbai 95, India

Ms.Supriya Solaskar
Xavier Institute of Engineering
Mahim Causeway,

***Abstract*— Smartphones are gaining popularity, creating new application areas as their capabilities increase in terms of computational power, sensors and communication. Emerging new features of mobile devices give opportunity to new threats. The most major threat of Android users is malware infection via Android application markets. General countermeasures to smartphone malwares are currently limited to signature-based antivirus scanners which efficiently detect known malwares, but they have serious shortcomings with new and unknown malwares creating a window of opportunity for attackers. Many methods are available to detect malware in mobile applications. But this doesn't offer a one-stop-shop solution to all types of problems. This paper includes survey of static and dynamic methods used for detecting threats in mobile apps.**

Keywords*— *Smartphone, Mobile OS, Threats

I. INTRODUCTION

A smart phone is mobile phone which is capable of doing much more things as compare to traditional phones. In the past, mobile phones were mostly about making phone calls. They had a number pad, a digital phone book and a pick-up/hang-up button and not much more. They can run programs and games, access the internet, send email and much more [1].

The smartphone operating system (OS) movement has grown to include competitors such as Google, Microsoft, Apple, Symbian, and Palm. Although these operating system platforms have come a long way since their inception, none of these companies provide an OS that is ideal for all users. They claim that their platforms perform the best in all endeavors and will certainly not advertise any weakness with their systems. This makes it difficult for end users to know which platform is best suited for their need. To address this problem, we perform a comprehensive analysis of each mobile operating system in order to identify its strengths and weaknesses. From these results, we can determine what phone is best suited for third party development, gaming, business applications, and multimedia. A smart phone consists of following operating systems:

Android OS:
Android is a new mobile device operating systems. It is first announced by Open Handset Alliance (OHA).

This modern mobile operating system is built on top of a Linux kernel version 2.6. A Linux kernel provides the interfaces for upper level to access low level hardware control functionalities. On the other hand, all Android applications, including core applications, are developed in Java, the most popular cross- platform programming language. Each Java program is run as a single process on Linux with its own instance of a Dalvik Virtual Machines (DVM), which is an optimized java virtual machine for Mobile Device[3]

SYMBIAN OS:
The Symbian OS is the operating system developed and sold by Symbian Ltd. The OS is used primarily by Nokia with its S60 user interface and by Sony Ericsson with its UIQ user interface. The Symbian OS was designed specifically for mobil e devices. It has very small memory footprint and lo w power consumption. It is an open OS, enabling third party developers to write and install applications independently from the device manufacturers.

WINDOWS MOBILE :
Windows Phone is a proprietary mobile operating system developed by Microsoft. Windows Phone introduced a new design language, previously called Metro UI, but later renamed to simply Modern.Software development for the Windows Mobile OS is done using Visual C++ making use of Microsofts' .NET framework. The SDK is setup to work using Visual Studio as the Integrated Development Environment (IDE).

II LITERATURE SURVEY

2.2 Mobile Threat [2]
Like viruses and spyware that can infect your PC, there are a variety of security threats that can affect mobile devices. We divide these mobile threats into several categories: application-based threats, web-based threats, network-based threats and physical threats.

2.2.1 Application-Based Threats [2]:Downloadable applications can present many types of security issues for mobile devices. "Malicious

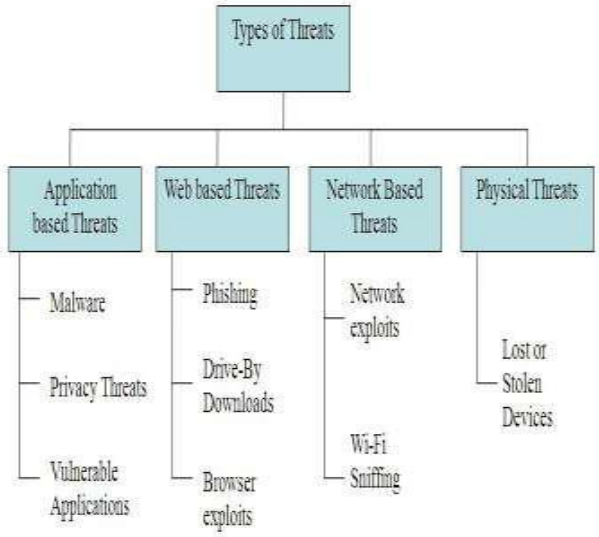

Fig. 1: Types of Threats

apps" may look fine on a download site, but they are specifically designed to commit fraud. Even some legitimate software can be exploited for fraudulent purposes. Application-based threats generally fit into one or more of the following categories:

Malware is software that performs malicious actions while installed on your phone. Without your knowledge, malware can make charges to your phone bill, send unsolicited messages to your contact list, or give an attacker control over your device.

Privacy Threats may be caused by applications that are not necessarily malicious, but gather or use sensitive information (e.g., location, contact lists, personally identifiable information) than is necessary to perform their function.

Vulnerable Applications are apps that contain flaws which can be exploited for malicious purposes. Such vulnerabilities allow an attacker to access sensitive information, perform undesirable actions, stop a service from functioning correctly, or download apps to your device without your knowledge.

2.2.2 Web-based Threats [2]
Because mobile devices are constantly connected to the Internet and frequently used to access web-based services, web-based threats pose persistent issues for mobile devices:

Phishing Scams use email, text messages, Facebook, and Twitter to send you links to websites that are designed to trick you into providing information like passwords or account numbers. Often these messages and sites are very different to distinguish from those of your bank or other legitimate sources.

Drive-By Downloads can automatically download an application when you visit a web page. In some cases, you must take action to open the downloaded application, while in other cases the application can start automatically.

Browser exploits take advantage of vulnerabilities in your mobile web browser or software launched by the browser such as a Flash player, PDF reader, or image viewer. Simply by visiting an unsafe web page, you can trigger a browser exploit that can install malware or perform other actions on your device.

2.2.3 Network Threats [2]
Mobile devices typically support cellular networks as well as local wireless networks (WiFi, Bluetooth). Both of these types of networks can host different classes of threats:

Network exploits take advantage of flaws in the mobile operating system or other software that operates on local or cellular networks. Once connected, they can install malware on your phone without your knowledge.

Wi-Fi Sniffing intercepts data as it is traveling through the air between the device and the Wi-Fi access point. Many applications and web pages do not use proper security measures, sending unencrypted data across the network that can be easily read by someone who is grabbing data as it travels.

2.2.4 Physical Threats [2]
Mobile devices are small, valuable and we carry them everywhere with us, so their physical security is also an important consideration.

Lost or Stolen Devices are one of the most prevalent mobile threats. The mobile device is valuable not

only because the hardware itself can be re-sold on the black market, but more importantly because of the sensitive personal and organization information it may contain.

2.3 Malware Detection Techniques

Malware has had a tremendous impact on the world as we know it. The rising number of computer security incidents suggests that malware is an epidemic.[3]

This Malware can also be termed as all kind of intrusions that is disastrous to the computer software and hardware system. Malware writer creates malware for different reasons and purposes ranging from challenges to economic gain, destruction to retaliation among others. Its growth is highly alarming in volume and its rate of expansion cannot be overlooked due to its damages. [4]

The task of detecting malware can be categorized into analysis, classification, detection and eventual containment of malware. Several classification techniques have been used in order to classify malware according to their instances and this has made it possible to recognize the type and activities of a malware and new variant. Analysis of malware has to do with identifying the instances of malware by different classification schemes using the attributes of known malware characteristics. Malware detection has to do with the quick detection and validation of any instance of malware in order to prevent further damage to the system. The last part of the job is containment of the malware, which involves effort at stopping escalation and preventing further damages to the system. A commercial antivirus uses signature based technique where the database must be regularly updated in order to possess the latest virus data detection mechanisms. However, the zero-day malicious exploit malware cannot be detected by antivirus, based on signature-based scanner, but the use of statistical binary content analysis of file to detect anomalous file segments. Toward this end, malware detection technique has been classified according to the following:

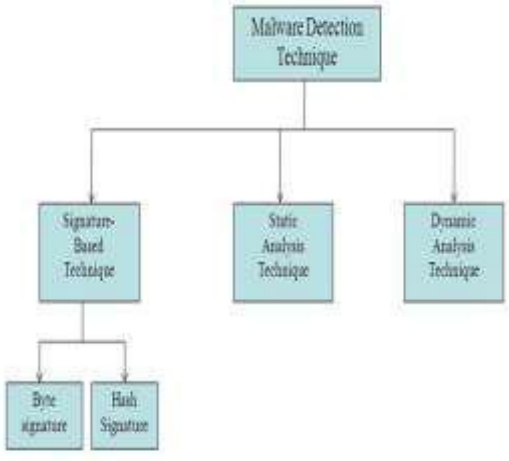

Fig.2: Malware Detection Techniques

2.3.1 Signature-based Technique:

Commercial antivirus scanners look for signatures which are typically a sequence of bytes within the malware code to declare that the program scanned is malicious in nature. Basically there are three kinds of malwares: basic, polymorphic, metamorphic malwares. In basic malware the program entry point is changed such that the control is transferred to malicious payload. Detection is relatively if the signature can be found for the viral code [5]. A pattern-matching approach commercial antivirus is an example of signature based malware detection where the scanner scans for a sequence of byte within a program code to identify and report a malicious code. This approach to malware detection adopts a syntactic level of code instructions in order to detect malware by analyzing the code during program compilation. This technique usually covers complete program code and within a short period of time.

2.3.2 Static Analysis [7]

Finding malicious characteristics or bad code segments in an application without executing them. The technique illustrated in Figure 3(a) is widely used in a preliminary analysis, when suspicious applications are first evaluated to detect any obvious security threats. This technique uses IDA Pro to disassemble the mobile application and extract system calls (feature extraction). It then use Centroid Machine, a lightweight clustering mechanism, to classify the mobile application as either malicious or benign (anomaly detection) [7].Static analysis is a quick, inexpensive approach to.Interactivity culminates in a built-in programming language and open plugin architecture.

Figure 3(b) a technique proposed for performing static taint analysis on iOS application binaries.

It disassembles the mobile application and constructs a control flow graph (CFG). The analysis considers paths originating from sensitive sources, such as theaddress book, current GPS coordinates, keyboard cache, unique device ID, and other phone-related information. Dataflow analysis checks for any sensitive data transmitted from the source to synch without notifying the user and thus causing privacy leaks. It can only detect privacy leaks within a single application, and it fails if two or more applications are transitively chained together [7].

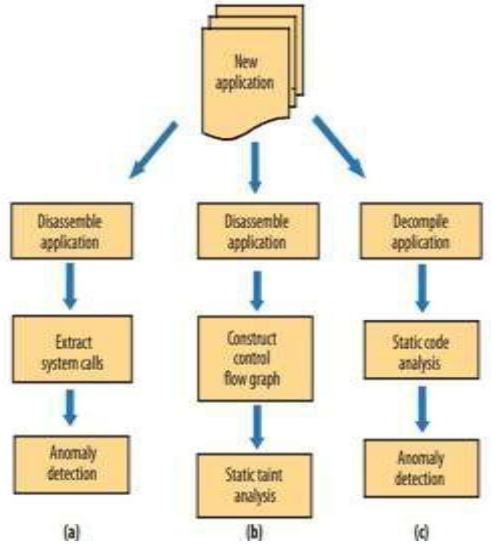

Fig.3: Static analysis Based on (a)System Calls(b)Taint (c)Source Code

2.3.3 Dynamic Analysis [7]

Unlike static analysis, dynamic analysis involves executing the mobile application in an isolated environment, such as a virtual machine or emulator, so that researchers can monitor the application's dynamic behavior. Researchers primarily use dynamic analysis in taint tracking or system call tracing [7].

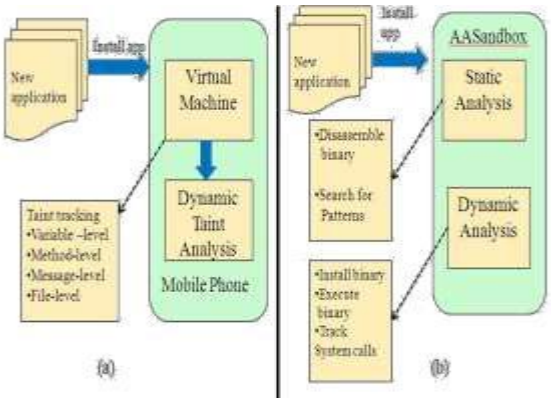

Fig.4: Dynamic Analysis Based on (a) System wide (b)Sandbox -Based

TaintDroid provides system-wide dynamic taint tracking. As Figure 4(a) shows, the mobile application passes to the Dalvik virtual machine to perform four granularities of taint propagation: variable, method, message, and file-level. Taint tracking marks any ambiguous datathat originates from sensitive sources, such as location, microphone, camera, and other phone identifiers. This technique modifies the native library loader to ensure that all the native libraries are called from the virtual machine, thus preventing untrusted applications from executing native methods directly. Finally, dynamic analysis screens impacted data for any potentially sensitive data leaks before it leaves the system at the network interface—a taint sink [7].

The Android Application Sandbox (AASandbox) system offers two-step analysis for Android applications. As Figure 4(b) shows, the mobile application passes to AASandbox, where it performs static and dynamic analyses in offline mode. Static analysis disassembles the applicationimage binary and uses the disassembled code to search for any suspicious patterns Dynamic analysis executes the binary in an Android emulator and logs the system calls [7].

CONCLUSION

The growing popularity and sophistication of smartphones, such as the devices based on Android have also increased concerns about the privacy of their users. To address these concerns, smartphone OS designers have been using different security models to protect the security and privacy of users. This paper includes techniques used for detecting malwares on smart phones and classified broadly into signature-based detection technique, static analysis technique, and dynamic analysis technique.

REFERENCES

[1] Yong Wang, Kevin Streffet, Sonell Raman, "Smartphone security Challenges" ,pp. 52 58,Vol. 45, Issue: 12 ,IEEE 2012

[2] "A Survey of Malware Detection Techniques", [Online] Available:http://cyberunited.com/wp-content/uploads/2013/03/A-Survey-of-Malware-Detection-Techniques.pdf

[3] Vinit B. Mohata , Dhananjay M. Dakhane , RavindraL.Pardhi ,"Mobile Malware Detection

[4] Techniques", International Journal of Computer Science & Engineering Technology (IJCSET),pp.2239-2235,vol.04,April 2013

[5] AbdelfattahAmamra, ChamseddineTalhi, and Jean-Marc Robert, "SmartphonMalware Detection: From a Survey Towards Taxonomy ", 7th International Conference on Malicious and Unwanted Software (MALWARE), pp.79-86, IEEE 2012

[6] Mahinthan Chandrmohan and HeeBengKuan Tan, "Detection of Mobile Malware in the Wild", pp.65 - 71 ,Vol.45, Issue: 9 IEEE Computer Society, September 2012

[7] Yong Wang, Kevin Streffet,Sonell Raman year , "Smartphone security Challenges",Vol. 45 , Issue: 12 , pp. 52 - 58 ,IEEE Transaction, 2012

[8] Charlie Miller, "Mobile Attacks and Defense"vol.9,pp. 68 - 70 IEEE JULY/AUGUST, 2011.