AN ANALYSIS OF MOBILE MALWARE AND DETECTION TECHNIQUES

Author:
Aswathy Dinesh
Aswathy.dinesh@tufts.edu

Supervisor:
Ming Chow

Abstract

Day by day the number of smartphone users is increasing rapidly, along with smartphone usage mobile malware attacks are also growing. Malware is malicious software used to disrupt, gather information, or gain access to a computer system or mobile device. Malware developers make use of third–party application to inject malicious content into smartphone and compromise phone security. Malware detectors are the primary tools to fight against these malwares. The success of malware detectors are based on techniques it uses. The primary objective of this report is to focus on various malware detection techniques along with their strengths, limitations and help users assess the risk imposed by malware.

Keywords
Malware, Smartphones, iPhone, Android, Threats.

Contents

1. Introduction

Smartphone usage has been rapidly increasing and it is increasingly becoming a sophisticated device. This increasing popularity makes the attackers more attracted to these devices. Smartphone use is now not just limited to personal conversation but has expanded to financial transactions, internet banking and for storing personal data. This has made smartphones more vulnerable to malware attacks and a target for information and identity theft.

Researchers from Kaspersky Lab first found the malware called Cabire, for mobile phone in 2004[1]. After that number of malware increased largely along with the popularity of the smartphones. This paper discusses about mobile malwares and analysis of different mobile malware detection techniques.

2. To the Community

This paper is primarily aimed at an average smartphone user. Today smartphone uses ranges from communication, internet banking, online shopping, pictures, games, home security and fitness. There is hardly an area which does not involve smartphone. This in turn has resulted in an average user supplying or storing a lot of information using their phone, but in most instances these users aren't aware whether the data they are providing through their devices is secure or not. The attackers take advantage of these factors and personal details or credentials from

users. This paper gives an idea to the user about different types of malwares, their impact and common detection techniques.

3. Action Items

I will start the discussion by briefly summarizing about mobile malware, types and symptoms in phone in section 3.1, in section 3.2 I discuss about the malware creators and what they want. In section 3.3 I discuss about malware detectors. Finally in section 3.4 I will take a look at various detection techniques for mobile devices followed by conclusion.

3.1 What Is Mobile Malware?

Malicious software that is designed specifically to target a mobile device, such as a tablet or smartphone in order to damage or disrupt the device. Mobile malware first emerged as early as 2004 targeting the Symbian OS [1], since then the number of malwares increased rapidly along with popularity of smartphones. Theses nefarious program either install themselves or are installed on the device by unwitting mobile users, and then perform functions without user knowledge or permission. They can be distributed through the internet via mobile browser, downloaded from stores or even installed via device messaging functions. The insidious objectives of mobile malware range from spying to key logging, from text messaging to phishing, from unwanted marketing to outright fraud [2].

Mobile Malware can broadly be classified into four types [2]:

3.1.1 Spyware and Adware

Spyware secretly gather confidential information about the mobile users and relays this data to third party. In some cases these may be advertisers or marketing data firms, which is why spyware is sometimes referred to as “adware”. Spyware uses the victim’s mobile connection to relay personal information such as contacts, location, message habits, and browser history and user preferences or downloads. Spyware also gathers devices information such as OS versions, product ID, International Mobile Equipment Identity number, and International Mobile Subscriber Identity number which can be used for future attacks.

3.1.2 Trojans and Viruses

Mobile Trojans infect user devices by attaching themselves to seemingly harmless or legitimate programs, Trojans are installed with the app and then carry out malicious actions. Trojans are closely related to mobile viruses. Malicious parties can use mobile viruses to root the device and gain access to files and flash memory.

3.1.3 Phishing Apps

Just like desktop computing, attackers are creating mobile phishing apps that look like legitimate services but may steal sensitive information and credentials to perform financial fraud. One such recent example was a fake security app for Facebook, which claimed to secure user’s Facebook account but infact stole user’s information for identity theft.

3.1.4 Bot Processes

Mobile malware is getting more sophisticated with programs that operate in the background on the user devices, concealing themselves and lying in wait for certain behaviors like an online banking session to strike. Hidden processes can execute completely invisible to the user, run executable or contact botmasters for new instructions.

Mobile Malware Symptoms:

These types of mobile malware differ greatly in how they spread and infect devices; they all can produce similar symptoms. Signs of malware infection can include unwanted behaviors and degradation of device performance. Mobile malware can reduce battery life or processing power, hijack the browser, send unauthorized SMS message and freeze the device entirely.

3.2 Who are the Malware Creators?

Malware writers go by variety of names. Some of the most popular names are black hats, hackers, and crackers. There are essentially two phases in the lifecycle of software during which malware is inserted. These are pre-release phase and post-release phase. An internal threat or insider is a one type of hacker capable of inserting malware into software before its release to the end-user. An insider is a trusted developer, typically within the organization of software to be deployed to its end-users. All other persons or organizations that take on the hacker role insert malware during the post-release phase, when the software is available for its intended audience.

3.3 Malware Detector

The malware detector attempts to help protect the system by detecting malicious behavior. The malware detector performs its protection through the manifested malware detection technique(s).

Malware detectors take two inputs. One input is its knowledge of the malicious behavior; this knowledge comes from the learning phase. The other input is the program under inspection. Once the malware detector has the knowledge of what it considers malicious behavior and the program under inspection, it can employ its detection techniques to decide if the program is malicious or benign. Although Intrusion Detection System (IDS) and malware detectors are sometimes used synonymously, a malware detector is usually only a component of complete IDS.

3.4 Malware Detection Techniques

In this section, I analyze various mobile malware detection techniques from various research papers.

3.4.1 Static Analysis

Static analysis is a quick, inexpensive approach to finding malicious characteristics or bad code segments in an application without executing them. These techniques are used in a preliminary analysis, when suspicious applications are first evaluated to detect any security threats.

System Call: The mobile application is first dissembled using tools like IDA pro. The tool is used to extract the System calls made by the application and then passed to Centroid Machine to perform anomaly detection and classify application based on the malicious activities.

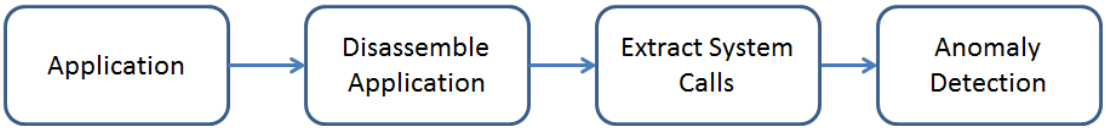

Figure 1: System Call

Static Taint Analysis: This analysis is performed using PiOS , a technique proposed for performing static taint analysis on iOS application binaries. The PiOS tool uses Static Analysis to check if the application accesses sensitive information and transmits it over the network. PiOS first creates a control flow graph from the application binaries. The analysis considers paths originating from sensitive sources, such as the address 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 sync without notifying the user and thus causing privacy leaks.

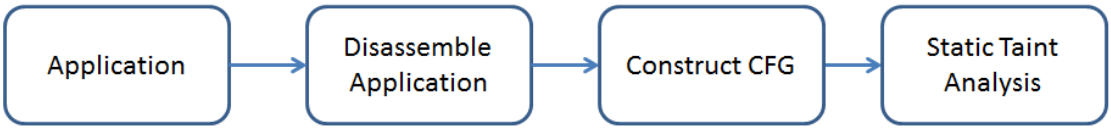

Figure 2: Static Taint Analysis

Source code analysis: This malware detection technique was proposed for Android. This approach uses ded, a Dalvik decompiler to generate java source code from the application's installation image and then use Fortify SCA, a static code analysis suite, to evaluate the recovered source code.

Figure 3: Source Code Analysis

3.4.2 Dynamic analysis

Dynamically monitoring the behavior of mobile application in an isolated environment is termed as Dynamic or Behavioral Analysis. Researchers primarily use dynamic analysis in taint tracking or system call tracing.

TaintDroid provides system-wide dynamic taint tracking for Android [5]. 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 data that 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.

TaintDroid might suffer from false negative and false positive results; in addition, it focuses solely on dataflow and doesn't consider other vulnerabilities.

The Android Application Sandbox (AASandbox) system is another technique which offers two-step analysis for an android application [5]. A mobile application passes to AASandbox, where it performs static and dynamic analysis in offline mode. Static analysis disables the application image binary and uses the disassembled code to search for suspicious patterns. Dynamic analysis executes the binary in an Android emulator and logs the system calls.

3.4.3 Application Permission analysis:

Applications need some permission to access certain data. At the time of installation, Android platform asks the user to grant or deny permissions based on the activities the application can perform.

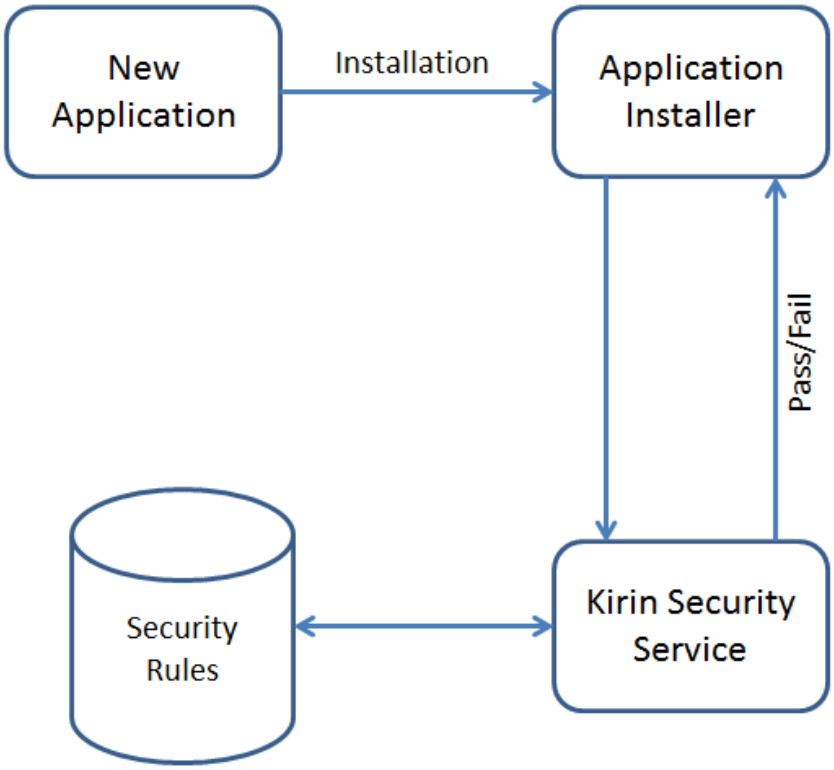

Figure 4: Application Permission Analysis

Figure 4 shows Kirin security service for Android platform. Kirin performs a permission check on the application during installation. When a user installs an application, Kirin extracts its security configuration and checks them against the security policy rule that it already has. If an application fails to pass all the security policy rules, Kirin can either delete it or alert the user.

3.4.4 Cloud-Based Detection

In this scheme a lightweight client application monitors the system calls in the device and sends it to the server in cloud to detect malicious behavior.

Paranoid Android is a cloud-based malware protection technique that moves security analysis and computations to a remote server that hosts multiple replicas of mobile phones running on emulators [8]. A tracer, located in the smartphone, records all the necessary information required to reply to the mobile application's execution. The tracer transmits the recorded information to the cloud-based replier, which replays the execution in the emulator. The replier can deploy several security checks, such as dynamic malware analysis, memory scanners, system call anomaly detection, and commercial antivirus scanning from the cloud's ample resources.

Crowdroid is a behavior-based mobile malware detection technique for Android [9]. Crowdroid is a lightweight client application that monitors system calls invoked by the target mobile application, preprocesses the calls, and sends them to cloud where a clustering technique helps determine whether the application is malicious. Increased use of Crowdroid results in improved malware detection but

using the approach initially might cause false positives, as the sample size is still very small.

3.4.5 Battery Life Monitoring

As malicious application tend to use most of the battery capacity. An interesting methodology VirusMeter was proposed by Liu et al. [5] to detect energy consumption and detect malware. VirusMeter detects anomalous behavior by abnormal power consumption. The idea behind this approach is any malicious activity would consume more battery. VirusMeter monitors the activities in the phone and uses APIs provided by the mobile platform to collect the remaining battery capacity. Based on the collected data it computes how much the application can consume battery and compares it with the power model. If there is a difference and if it exceeds the threshold then it raises alarm.

4. Conclusion

In a world where everything from front door to bank account can be controlled from with a smartphone it is essential that an individual's personal information is not easily compromised. Proliferation of smartphones has brought along with it sophisticated malwares which are challenging to detect.

To some extent a user can take some simple steps like using a security software in conjunction with not clicking on suspicious links or providing personal information on suspicious sites or apps, checking for SSL certificate when entering

information on financial sites although not fool proof this will guard user from compromising his personal information.

More importantly user should be careful enough rather than blindly downloading an application.

5. References

[1] Trend Micro. "A Brief History of Mobile Malware". http://countermeasures.trendmicro.eu/wp-content/uploads/2012/02/History-of-Mobile-Malware.pdf

[2] Dupaul, N. "Common Mobile Malware Types: Cybersecurity 101", Oct 2013. http://www.veracode.com/blog/2013/10/common-mobile-malware-types-cybersecurity-101/

[3] Idika, N. Mathur, A. "A Survey of Malware Detection Techniques", http://cyberunited.com/wp-content/uploads/2013/03/A-Survey-of-Malware-Detection-Techniques.pdf

[4] Mohata, V. Dakhane, V. Pardhi, R. "Mobile Malware Detection Techniques" http://www.ijcset.com/docs/IJCSET13-04-04-094.pdf

[5] Mahinthan Chandramohan, Hee Beng Kuan Than. "Detection of Mobile Malware in the Wild" http://www.academia.edu/1335332/PrePrint_Detection_of_Mobile_Malware_in_the_Wild

[6] Liu, L. G., Zhang, Y, X., Chen. S. "VirusMeter: Preventing your cellphone from spies" In Proceedings of RAID, volume 5758 of Lecture Notes in Computer Science, 2009.

[7] Portokalidis, G. et al., "Paranoid Android: Versatile Protection for Smartphones," Proc. Ann. Computer Security Applications Conf. (ACSAC 10) ACM, 2010, pp. 347-356.

[8] Burguera, U. Zurutuza, and S. Nadjm-Tehrani, "Crowdroid: Behavior-Based Malware Detection System for Android," Proc. ACM Workshop Security and Privacy in Mobile Devices (SPMD 11), ACM, 2011, pp. 15-26.

[9] Ramu, S. "Mobile Malware Evolution, Detection and Defence", April 2012.