Digital Investigation 14 (2015) 17–35

Contents lists available at ScienceDirect

Digital Investigation

journal homepage: www.elsevier.com/locate/diin

Andro-AutoPsy: Anti-malware system based on similarity matching of malware and malware creator-centric information

Jae-wook Jang [a], Hyunjae Kang [b], Jiyoung Woo [a], Aziz Mohaisen [c], Huy Kang Kim [a, *]

[a] Graduate School of Information Security, Korea University, Republic of Korea
[b] Enterprise Risk Services, Deloitte Anjin LLC, Republic of Korea
[c] Computer Science and Engineering Department, State University of New York at Buffalo (SUNY Buffalo), USA

ARTICLE INFO

Article history:
Received 12 October 2014
Received in revised form 24 June 2015
Accepted 27 June 2015
Available online 16 July 2015

Keywords:
Similarity matching
Profiling
Android malware
Malware classification
Certificate

ABSTRACT

Mobile security threats have recently emerged because of the fast growth in mobile technologies and the essential role that mobile devices play in our daily lives. For that, and to particularly address threats associated with malware, various techniques are developed in the literature, including ones that utilize static, dynamic, on-device, off-device, and hybrid approaches for identifying, classifying, and defend against mobile threats. Those techniques fail at times, and succeed at other times, while creating a trade-off of performance and operation. In this paper, we contribute to the mobile security defense posture by introducing Andro-AutoPsy, an anti-malware system based on similarity matching of malware-centric and malware creator-centric information. Using Andro-AutoPsy, we detect and classify malware samples into similar subgroups by exploiting the profiles extracted from integrated footprints, which are implicitly equivalent to distinct characteristics. The experimental results demonstrate that Andro-AutoPsy is scalable, performs precisely in detecting and classifying malware with low false positives and false negatives, and is capable of identifying zero-day mobile malware.

Introduction

The explosive growth in the number of mobile devices running the Android platform has attracted the attention of malware creators because a vast amount of private information (e.g., contacts, short messages, and e-mails) is usually stored on these devices. The availability of this information in many mass-market mobile devices renders them a desirable target for malware creators, making the security of mobile devices one of the most important, yet challenging, areas of research.

Mobile as well as traditional malware analysis for detection and classification falls into two broad types: dynamic and static. Dynamic analysis aims to provide methods for effectively and efficiently extracting the unique patterns of each malware family based on its behavior. If malware programs (or samples) behave in a unique way under a specific condition, this type of analysis fails to detect them because it does not recognize their intended malicious behavior. Dynamic analysis techniques, on the other hand, analyze malware on an emulator or a mobile device without the human interaction, providing autonomous installation and execution. On one hand, this type of analysis has several limitations with respect to

* Corresponding author. Tel.: +82 2 3290 4898.
E-mail addresses: changkr@korea.ac.kr (J.-w. Jang), janetk1004@gmail.com (H. Kang), jywoo@korea.ac.kr (J. Woo), mohaisen@buffalo.edu (A. Mohaisen), cenda@korea.ac.kr (H.K. Kim).

http://dx.doi.org/10.1016/j.diin.2015.06.002
1742-2876/© 2015 Elsevier Ltd. All rights reserved.

analyzing malware embedding updates, drive-by downloads, or C&C (Command & Control) attacks (Zhou and Jiang, 2012). Furthermore, for capturing the unique behavior of malware accurately, dynamic analysis needs to roll back the emulator or mobile device into its clean state whenever the analysis of a piece of malware is complete. On the other hand, using static analysis, strings of bytes associated with malware samples are discovered through reverse engineering and used as a signature for identifying malware. In spite of effective characteristics, static techniques are often prone to high false positive rates because of the evolution in the code basis and code repackaging often associated with malware. Static analysis techniques require more efforts in reverse engineering to generate reliable and meaningful signatures. Furthermore, recent and new malware families have utilized embedding obfuscation techniques, such as proguard, which change the calling order or the method names of variables and functions, and thus hinder the transformation of the malware itself into source code, making such defenses ineffective.

Despite the efforts of antivirus (AV) vendors, the amount of malware is increasing exponentially. According to a report by McAfee, 2.47 million new pieces of mobile malware and a total of 3.73 million pieces of malware appeared in 2013 (McAfee, 2013). Between the end of 2012 and 2013, the total amount of malware increased by nearly 200%. To address this trend, AV vendors analyze a large number of malware samples every day in order to prevent their widespread dissemination and to guide users on disinfection and risk management by classifying malware into broad families. However, malware-centric analysis, including both static and dynamic analysis, is limited, and is not keeping pace with the trends increasing numbers of malware and their families. Existing malware analysis method focuses on codes and functions of malware. In particular, static analysis takes a long time to parse meaningful code patterns in disassembled or decompiled codes, and dynamic analysis requires an irritating amount of analyst-guided pre-analysis time for operations such as roll-back of emulator or mobile devices.

The added contents reads as follows: The “Trojan horse defense” surfaced in 2003 in several cybercrime cases brought in the United Kingdom is a good example of how our work can relate to the digital investigation community; the incident pertains to the claim that a malware creator ran malicious codes on victim's device without the device owner's consent (Brenner et al., 2004). The defense attributed actions to malware and has presented a challenging issue to the forensics community: the accurate assessment and investigation to determine whether a person is innocent or guilty. Technically, investigators needed to determine if a system was compromised, and if so what are the implications of such compromise and what unapproved activity was the device doing. To this end, our proposed method incorporates malware creator information as well as malware-centric information to attribute malware. Our method identifies direct evidence of malicious behaviors of malware creators to detect malware created by them and analyzes the malicious behavior and attacker's intent.

To overcome the drawbacks inherent in previous malware-centric methods and help investigators answer these questions, we propose a novel and feature-rich anti-malware system based on profiling, called Andro-AutoPsy. Our system is a novel hybrid malware detection and classification method based on similarity matching of profiles. Our proposed profiling system, which comprises mobile devices and a remote server, is analogous to criminal profiling. In the real world, criminal profiling, also known as offender profiling, is a methodology that is intended for helping investigators accurately predict and profile the characteristics of unknown criminal subjects or offenders (Kocsis, 2009; Nykodym et al., 2005; Rogers, 2003). We adopt criminal profiling methodology in the malware analysis domain. In order to respond to malware more efficiently and effectively, malware analysts need to check the target of an attack, since it reflects the attack's intent of the malware creator. Such process tries to achieve the end goals by answering the following questions. 1) What do malware creators want to obtain? 2) How do malware creators attack the victim? 3) What do malware creators need for an attack? By answering these questions, analysts can understand malware creators' attack pattern.

For understanding the intent of malware creator, we exploit integrated footprints, including opcodes in *.smali*, meta-data in *Androidmanifest.xml*, and the serial number of a certificate as feature vectors for malware characterization. We observe that a) malware samples have unique malicious behavior patterns and characteristics, b) the malicious behavior of malware samples is determined by operation codes (opcodes) and requires a particular permission set, and c) such an opcode set influences the behavior of the malware. To operate our system at scale, we represent malware characteristics using Bayer's profiling as described in Bayer et al. (2009). We prepare a representative profile that combines multiple features. For that, and for each malware family, we characterize it by integrated footprints using static analysis features. Then, by comparing the profiles, we detect and classify malware samples into similar groups.

Contribution:

1. We propose an “Integrated Malware Analysis System” which considers malware-centric information as well as malware creator-centric information. Using the serial number of a certificate simplifies the process of malware detection and classification.
2. We demonstrate the operational relevance of our system. Our system enables AV vendors to react to many species of malicious samples by quickly and efficiently conducting similarity matching between these and previously detected samples. Our system facilitates the detection of new malware, including existing malware's variants and zero-day exploits. This is further highlighted through in-depth experiments using real-world malware samples. Our system implements an efficient malware detection and classification method. Despite using static analysis, it requires only 72 s/MB to detect and classify malware into similar groups.