Digital Investigation 10 (2013) S96–S104

Contents lists available at SciVerse ScienceDirect

Digital Investigation

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

Automated identification of installed malicious Android applications

Mark Guido*, Jared Ondricek, Justin Grover, David Wilburn, Thanh Nguyen, Andrew Hunt

The MITRE Corporation, 7515 Colshire Drive, Mclean, VA 22102, USA

Keywords:
Android
Mobile forensics
Periodic
Enterprise
Monitoring

A B S T R A C T

Increasingly, Android smartphones are becoming more pervasive within the government and industry, despite the limited ways to detect malicious applications installed to these phones' operating systems. Although enterprise security mechanisms are being developed for use on Android devices, these methods cannot detect previously unknown malicious applications. As more sensitive enterprise information becomes available and accessible on these smartphones, the risk of data loss inherently increases. A malicious application's actions could potentially leave sensitive data exposed with little recourse. Without an effective corporate monitoring solution in place for these mobile devices, organizations will continue to lack the ability to determine when a compromise has occurred. This paper presents research that applies traditional digital forensic techniques to remotely monitor and audit Android smartphones. The smartphone sends changed file system data to a remote server, allowing for expensive forensic processing and the offline application of traditional tools and techniques rarely applied to the mobile environment. The research aims at ascertaining new ways of identifying malicious Android applications and ultimately attempts to improve the state of enterprise smartphone monitoring. An on-phone client, server, database, and analysis framework was developed and tested using real mobile malware. The results are promising that the developed detection techniques identify changes to important system partitions; recognize file system changes, including file deletions; and find persistence and triggering mechanisms in newly installed applications. It is believed that these detection techniques should be performed by enterprises to identify malicious applications affecting their phone infrastructure.

1. Introduction

Android malware is increasingly becoming a problem for both enterprise and individual users alike. The number of malicious applications targeting the Android operating system is dramatically increasing. Kaspersky Labs reported that the total number of Android malware samples tripled compared to the previous quarters during the first six months of 2012.[1]

To combat Android malware, a current industry approach is to install a mobile virus scanner on a phone, much like an administrator would do on an enterprise laptop or desktop. These virus scanners typically run as applications inside the Dalvik virtual machine and compare newly installed applications against a known repository of malware signatures. This "blacklisting" technique has known weaknesses that can be exploited by malware distributors (Vidas et al, 2011b). A zero-day malicious application could surreptitiously escalate privileges by modifying critical system files and altering the phone's behavior, thus rendering virus-scanning engines blind to attacks.

Because smartphones are mobile and often operate outside the confines of the enterprise, network level

* Corresponding author.
E-mail address: mguido@mitre.org (M. Guido).

[1] Y. Namestnikov. (2012, Aug. 8). "IT Threat Evolution: Q2 2012," *SecureList*. Kaspersky Lab ZAO. Available: http://www.securelist.com/en/analysis/204792239/IT_Threat_Evolution_Q2_2012.

1742-2876/$ – see front matter © 2013 The MITRE Corporation. Published by Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.diin.2013.06.011

monitoring approaches are typically not effective. Therefore, it is necessary to have robust methods of monitoring the devices themselves for potentially malicious activities. Monitoring the usage pattern and stored data of an enterprise smartphone, whether performed by security personnel or malicious actors, can reveal unique information about users and the organizations they work for.

1.1. Project background

This research applies traditional forensic techniques to monitor and audit Android smartphones. The initial focus was on the enterprise use case, described in Section 1.2, where phone users inadvertently or intentionally install malicious applications. The focus of the research was to ascertain new ways of identifying malicious Android applications when they are installed or as they deliver their malicious payloads. Android phones are an appropriate target for this research because they are increasingly being selected for enterprise deployments.

1.2. Enterprise use case

The use case selected for this effort was detecting the unintentional or deliberate installation of malicious applications on Android smartphones associated with an enterprise; these smartphones may have been distributed to the enterprise user population and may have access to resources on the enterprise network. Enterprise users are assumed to have no privileged access on these phones, and part of this research is to keep enterprise users from obtaining privileged access. The malicious applications may lower the security posture of both the smartphone and the enterprise network to which it connects.

2. Related work

While the research incorporates several commonly used forensic tools and techniques, the methodology of pairing these tools and techniques into one system that would operate on the enterprise is unique. The computer forensics field is trending primarily toward live forensic analysis, as seen in traditional tools and systems such as Encase Enterprise and AccessData Enterprise. This research also falls under this category and is believed to be the first system of its kind for smartphones. Mobile forensics, considered "live" because of its reliance on the target smartphone's running kernel, typically uses commercially available mobile phone kits that take a one-time logical or physical image of a target phone (Lessard and Kessler, 2010). If these kits do not support certain target phones, specialized, albeit inconsistent, methods are used. Vidas, Zhang, and Christin (Vidas et al, 2011a) discussed these inconsistencies and provided a generalized method for forensic acquisition on Android phones where investigators have no prior knowledge of their contents. The research team drew upon these findings during the system design process.

Utilizing a running kernel to perform live forensic acquisitions in the enterprise can be problematic. It is theoretically possible for a malicious user or a malicious application to modify the running kernel to hide certain actions and skew the system's measurements. Weinstein (Weinstein, 2012) proposed a method of trusting the kernel, where achieving a practical amount of trust in a system's measurements would be possible.

The methodology used to send small updates of data from the phone over-the-air (OTA) is considered fairly novel, primarily because the use case was unique (Section 1.2). The system described herein was implemented for the enterprise and does not take into account evidentiary issues that may restrict some of the techniques used during a forensic acquisition. Teleporter, a previous capability that documented a technique for obtaining hard-drive images over limited bandwidth connections, was identified (Watkins et al., 2009). The research team analyzed their methods and incorporated a similar hashing technique, but the overall system design differed significantly from Teleporter.

The market for mobile forensic tools and techniques has grown as more mobile devices with capabilities rivaling laptops and desktops have flooded the market. Mobile operating systems now support and use file systems that traditionally have been found in laptops and desktops. This fact is significant that it allows the system to incorporate offline traditional forensic techniques when analyzing smartphones. Fairbanks (Fairbanks, 2012) researched the fourth extended file system (ext4) for digital forensics and incorporated support of the ext4 file system into The Sleuth Kit (TSK), a popular open-source forensic tool set that is easily incorporated into analytical systems.[2] The research team implemented Fairbanks' work and used it to identify added, modified, and deleted files in the ext4 file system found on many recent Android phones.

Another technique incorporated into the system was using the forensic tool `fwalk` to produce Digital Forensics Extensible Markup Language (DFXML) (Garfinkel, 2011). This use of DFXML allowed system processes to pass data into the detector architecture and analyze it in a standard way.

The Android Malware Genome Project (AMGP) was another related work effort that helped the research. In addition to being a source for the malware we used for experimentation, the classification work (Zhou and Jiang, 2012) on over 1200 malware samples influenced the system's development direction. These classifications inspired a detector that looked for malware that, when installed and executed, would establish persistence on a phone to survive a reboot. The research team theorized that malware could alternatively use a modified system.img, boot.img, or bootloader to install persistence mechanisms. One research effort that used a modified boot.img to exploit Universal Serial Bus (USB) connectivity between a USB client and a USB host (Wang and Stavrou, 2010) was identified and lent credence to the theory.

3. Periodic mobile forensics

The system performs periodic scans of an Android phone's block devices, identifying changes to specific bit

[2] B. Carrier *The Sleuth Kit* Available: http://www.sleuthkit.org/.