A Cloud-based Intrusion Detection and Response System for Mobile Phones

Amir Houmansadr, Saman A. Zonouz, and Robin Berthier
University of Illinois at Urbana-Champaign
{*ahouman2, saliari2, rgb*}*@illinois.edu*

***Abstract*—As smart mobile phones, so called *smartphones*, are getting more complex and more powerful to efficiently provide more functionalities, concerns are increasing regarding security threats against the smartphone users. Since smartphones use the same software architecture as in PCs, they are vulnerable to similar classes of security risks such as viruses, trojans, and worms [6]. In this paper, we propose a cloud-based smartphone-specific intrusion detection and response engine, which continuously performs an in-depth forensics analysis on the smartphone to detect any misbehavior. In case a misbehavior is detected, the proposed engine decides upon and takes optimal response actions to thwart the ongoing attacks. Despite the computational and storage resource limitations in smartphone devices, The engine can perform a complete and in-depth analysis on the smartphone, since all the investigations are carried out on an emulated device in a cloud environment.**

I. Introduction

Smartphones, as extremely fast-growing type of communication devices, offer more advanced computing and connectivity functionalities than contemporary mobile phones by conceptually integrating handheld computers' capabilities with phone devices. While most traditional mobile phones are able to run applications based on specific platforms such as Java ME, a smartphone usually allows the user to install and run more advanced third-party software applications. Being an "all-in-one" device, smartphones are increasingly getting attractive to a wide range of users. A recent study by ComScore Inc. indicates that over 45.5 million people in the United States owned smartphones in 2010, a 20% share of the total devices sold, and a continuous annual growth rate of 156% [1] is estimated.

Following smartphones' increasing popularity, attackers have also been interested in attacking to such platforms. In fact, a large number of smartphone malware have attempted to exploit unique vulnerabilities of smartphones. As a case in point, the smartphone virus Cabir [2] spreads and populates through the Bluetooth interface of smartphones. Another recent smartphone security study shows that trojans, using voice-recognition algorithms, can steal sensitive information that are talked through smartphones [9]. Such threats not only invade privacy and security of the smartphone users, but also manage to generate coordinated large-scale attacks on the communication infrastructures by forming botnets.

The past solutions for smartphone security encounter several limitations in practice. Many of such approaches are based on running a lightweight intrusion detection process on the smartphones [4]; these schemes fail to provide effective protection as they are constrained by the limited memory, storage, and computational resources, and battery power of the smartphone devices [3]. In addition, most of such approaches are based on detecting malware/intrusions

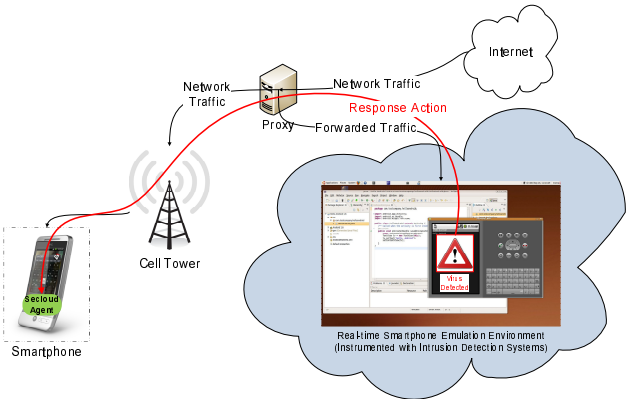

Figure 1. High-level Architecture

by looking for some specific *signatures* downloaded from a database. This, first of all, requires a large amount of storage on the device, and, secondly, such a signature-based approach is easily evaded by introducing zero-day threats. Another class of smartphone security solutions perform network-based intrusion detection [5]. Although this addresses the resource limitations affecting previously mentioned approach, their accuracy and performance is affected by their lack of knowledge on smartphone's internal activities. As another issue, none of the past proposed techniques provide automated response and recovery for the detected security threats. This is essential in order to quickly terminate the attack and restore the phone back to its normal operational mode.

II. System Design

To address the critical challenge of keeping smartphone secure, cloud-based intrusion detection has recently been introduced [7], [8]. We build on top of this concept and propose a synchronized cloud-based intrusion detection and response framework for smartphone devices. We seek the following objectives: 1) transparent operation to the users who are mostly technically unskilled; 2) light resource requirement, and 3) real-time and accurate intrusion detection and response. The proposed framework targets a practical scenario in which most smartphones cannot be equipped with heavyweight anti-malware software, but need to be protected against attacks. The proposed framework is in fact a cloud service which provides intrusion detection and response capabilities to the registered smartphones. It emulates the actual smartphone device in a virtual machine in cloud using a proxy which duplicates the in-coming traffic to the devices and forwards the traffic to the emulation platform. The real-time emulation on powerful servers allows the framework to instrument the emulated environment with a rich set of (possibly resource intensive) off-the-shelf intrusion forensics and detection

systems, which do not necessarily have to be lightweight, and perform a run-time in-depth detection analysis. The key difference with previous attempts is to replicate user input in real time. This enables our solution to have very limited bandwidth requirements while keeping the replica always synchronized. In case a misbehavior is detected, the intrusion response decides upon the best countermeasure actions and sends it to a non-intrusive software agent in the device, which is in charge of only carrying out the received actions.

Figure 1 shows the high-level architecture of the proposed framework and how its components interact with each other. A smartphone to be protected by the framework should be registered by its owner to the framework's online registration system. To register, the client should first specify his or her device, it operating system and the application list, so that the framework can instantiate an identical image of the smartphone in cloud. Additionally, the client is asked to install a very light-weight software *agent* on the smartphone that automatically would configure the proxy settings. A *proxy server* is responsible for duplicating the communication between the smartphone and the Internet and forwarding it to the *emulation environment* in cloud where the detection and forensics analyses are performed. Note that this does not disrupt the usual communication between the smartphone and the Internet. The light-weight agent on the smartphone performs three main tasks. It gathers all user and sensor inputs to the device, it sends them to the emulation environment, and it waits for potential response and recovery commands, e.g., killing the malicious application, from the emulation environment in order to take the required actions.

The real-time emulation environment is instrumented with several accurate off-the-shelf intrusion detection systems (IDSes), which currently cannot be deployed in smartphone devices due to their high resource requirements. The deployed set of detectors monitor different parts of the smartphone's software stack and perform an online and in-depth analysis to identify any malicious activity. In case a misbehavior is detected, our intrusion response engine [11] in the emulation environment solves an resource intensive game-theoretic optimization, and sends the selected optimal response action to the agent running on the smartphone device. The agent, then, can take the required actions and recover the smartphone back to its normal secure operational mode.

III. Implementation

We have implemented a working prototype of the intrusion forensics analysis engine for the Linux kernel, and we are currently working on the emulation environment. The forensics engine makes use of two sources of information: 1) set of IDSes that are deployed, and monitoring various aspects of the system; and 2) system calls which are logged by a loadable kernel module that we developed by manipulating the system call table, and in particular, replacing each system call function with a wrapper logging function. System call logs are used to create an information flow graph (see Figure 2) in which nodes are OS-level objects, e.g., processes, and directed edges represent data flows. The

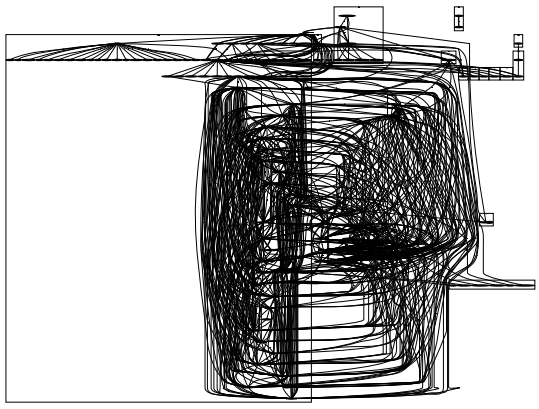

Figure 2. A sample system dependency graph

dependency graph is later augmented with triggered IDS alerts to automatically generate the *system attack graph* [10] which encodes the possible attack paths according to the provided information. The attack graph is later analyzed to identify the misbehaving smartphone application. For more information, the reader is referred to the original paper [10].

IV. Conclusions

We presented a cloud-based service to provide security and tolerance to resource limited mobile phone devices. We are currently deploying the framework on the Android-equipped HTC Droid Incredible smartphones. Our long-term plan is to later leverage the generated attack-graph to automatically decide upon response actions in an emulated smartphone environment as our intrusion response and recovery engine [11] does in computer systems.

References

[1] 2010. Windows mobile business value for mobile operators: http://download.microsoft.com/.

[2] 2010. Virus Library: http://www.viruslibrary.com/.

[3] C. Biever, 2005. Phone viruses: how bad is it?: http://www.newscientist.com/article.ns?id=dn7080.

[4] A. Boukerche and M. S. M. A. Notare. Behavior-based intrusion detection in mobile phone systems. *Jour. Paral. & Dist. Comp.*, 62(9):1476 – 1490, 2002.

[5] J. Cheng, S. H. Wong, H. Yang, and S. Lu. Smartsiren: virus detection and alert for smartphones. In *MobiSys*, pages 258–271, New York, NY, USA, 2007. ACM.

[6] J. Jamaluddin, N. Zotou, and P. Coulton. Mobile phone vulnerabilities: a new generation of malware. In *Consumer Electronics, 2004 IEEE International Symposium on*, pages 199 – 202, 2004.

[7] J. Oberheide, K. Veeraraghavan, E. Cooke, J. Flinn, and F. Jahanian. Virtualized in-cloud security services for mobile devices. In *Proceedings of the First Workshop on Virtualization in Mobile Computing*, pages 31–35. Citeseer, 2008.

[8] G. Portokalidis, P. Homburg, K. Anagnostakis, and H. Bos. Paranoid Android: versatile protection for smartphones. In *Proceedings of the 26th Annual Computer Security Applications Conference*, pages 347–356. ACM, 2010.

[9] R. Schlegel, K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X. Wang. Soundminer: A Stealthy and Context-Aware Sound Trojan for Smartphones. In *NDSS*, 2011.

[10] S. A. Zonouz, K. Joshi, and W. H. Sanders. Cost-aware systemwide intrusion defense via online forensics and on-demand detector deployment. In *CCS-SafeConfig*, pages 71 – 74, 2010.

[11] S. A. Zonouz, H. Khurana, W. Sanders, and T. Yardley. Rre: A game-theoretic intrusion response and recovery engine. In *DSN*, pages 439 –448, 2009.