Towards a Malware Detection Framework Based on Power Consumption Monitoring

Jarilyn M. Hernández Jiménez*†, Robert A. Bridges*, Jeffrey A. Nichols*, Katerina Goseva-Popstojanova†, and Stacy Prowell*

* Computational Science and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831
{hernandezjm1, bridgesra, nicholsja2, prowellsj}@ornl.gov

†Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, 26506
{jhernan7, katerina.goseva}@mail.wvu.edu

***Abstract*—As our personal, organizational, and critical infrastructure are increasingly dependent on networked computing assets, malicious software —malware—is one of the most serious national security threats. Common malware detection practices are proving insufficient, and the task poses significant challenges when faced with automatically generated and polymorphic malware, as well as rootkits, which are exceptionally hard to detect. To address these challenges, we propose an approach that uses an unavoidable consequence of malware—consumption of electrical power. The objective of this research is to determine whether malware generates a detectable signal in the power consumption of a general-purpose computer. Using unsupervised methods to analyze CPU and motherboard power data, we exhibit 87.5% true-positive, 0% false positive, 33.3% false-negative, and 100% true-negative rates when detecting the Alureon rootkit.**

I. Introduction

The current ability to protect networked assets from infection with malicious software (malware) is proving vastly insufficient, and this poses a serious national threat as compromises result in halting critical infrastructure, disclosing state secrets, and financial losses in the billions of dollars. While often our first line of defense against malware, anti-virus (AV) software and, more generally, signature-based detection methods, are simply unable to keep pace with the rate and sophistication of modern malware. By slightly changing the instructions of an existing malware sample, new malware instances, called variants, are now being generated automatically in extremely high volume (on the order of millions per day) [1]. While these variants appear to be different programs from the viewpoint of signature-based AV scanners, they exhibit similar functionality to their predecessor; consequently, new malware variants often enjoy the ability to bypass traditional detection methods until a signature for them can be identified and incorporated into detection software [2]. Moreover, modern malware is polymorphic, meaning it regularly rewrites its syntax while retaining identical functionality. Specifically, during propagation malware replaces byte sequences in its executable with completely different byte sequences that have the same net effect on the system. As a toy example, the statement, `if x == 0` can be replaced with `if x + 1 == 1`. Polymorphic malware can bypass simple heuristic detection techniques, yet authors of detection systems have attempted to address this problem by using approximate methods that are more powerful than signature matching; for example, byte frequency [3], general similarity measures [4], structural similarity [5], and behavioral analysis [6] are proposed techniques. A common weakness of these detection methods is that they operate on the machine being monitored; hence, successful attackers may disable the monitoring software or modify it to prevent detection after gaining entry to the computer [7]. This is evidenced by rootkits, which are a particularly insidious subclass of malware that achieve administrative privileges that they use to hide themselves. Consequently, rootkits are commonly associated with the establishment of advanced persistent threats and pose serious danger to our nation.

To overcome these limitations, we propose an approach to malware detection by examining the DC power consumption of the device. Because malware processes will necessarily change the power profile of the device, our hypothesis is that accurate detection of malware is possible via power profile analysis. We note that successful demonstration of this research addresses the problems of the current detection methods. Firstly, by using unsupervised learning techniques, we can detect qualitative changes in the power data without relying on signature generation—this bypasses the multitude of shortcomings associated with signature-based detection. Secondly, as neither static nor dynamic code analysis is performed, polymorphism will not be an effective evasion technique for the malware. Thirdly, by using out-of-band collection and processing of power data (i.e., on a separate machine), malware cannot hide itself from this detection. In summary, the use of power is an unavoidable consequence of malware actions, and our research task in this short paper is to prove the concept that even the most inconspicuous malware, rootkits, can be detected by using unsupervised learning on power profiles. Similar power-based malware detection research has emerged for smartphones with in-band power collection [8], medical devices with AC power and supervised techniques [9], software defined radio [10], and PFP[1] provides

This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

[1]http://pfpcyber.com/

a commercial product based on similar ideas. To the best of our knowledge, no academic research has tested the efficacy on general-purpose computers, in particular with respect to detecting rootkits.

II. Experimental Design & Data Collection

Our experimental system is a Dell OptiPlex 755, with a clean installation of Windows 7. Using a 16 channel, 250kHz data acquisition system (DAQ), we monitor current and voltage for the three rails (*3.3V*, *5V*, and *12V*) feeding the motherboard from the power supply unit (PSU), as well as the single *12V* rail to the CPU. In total, 8 channels (4 voltage channels and 4 corresponding current channels) were sampled every 10ms (.01s), and corresponding channels multiplied to obtain a 4-tuple of power for each sample. A laptop (separate computer) records power measurements, and for real-time visualization, we developed our own Visual Basic program.

To infect the experimental machine, we used the *Alureon* rootkit, which is a trojan that attempts to steal personal data by affecting network traffic. In particular, Alureon hides in the master boot record, which makes it exceptionally hard to detect [11]. We created a segregated network by connecting the laptop via wireless to the Mifi [12] and sharing that connection with the experimental machine. This design lets our chosen rootkit behave normally without any possibility of infecting the network and allowed us to monitor all of the experimental machine traffic.

To ensure repeatability, a Python script is run that executes the same sequence of events—opening ten windows of IE each with a five second delay—during three states: (1) before infection, (2) after infection, and (3) after infection plus reboot. In order to segment these sections of the power profile, we use a micro-benchmark written to stress the CPU for 5 seconds. This places markers in the power data before and after the IE section. This workflow is completed four times for each state resulting in four "clean," four "infected," and four "infected+rebooted" power profiles to be compared.

III. Results & Conclusions

Our primary goal is to prove that by only seeing "clean" power profile segment(s), we can accurately detect the infected segments. To do this, we robustly fit a Gaussian to each of the 12 data sets using the *Minimum Covariance Determinant* method with $h = .9$ [13]. This algorithm automatically discards the 10% most outlying data points before fitting the Gaussian so that anomalies will not effect the model. We use a 4-variate Gaussian to capture the correlation between the four monitored rails. Finally, we pairwise compare the Gaussians using absolute KL divergence (i.e., the information gain)—this value is large if and only if the two models are dissimilar. Our results show that the KL divergence of any two clean sets is below 0.0825. Setting a threshold of 0.1, and taking any of the "clean" data sets as a baseline, we can identify all infected data segments except "infected_1." This gives 7/8 = 87.5% true-positive, 1/3 = 33.3% false-negative, 0% false-positive and 100% true-negative rate.

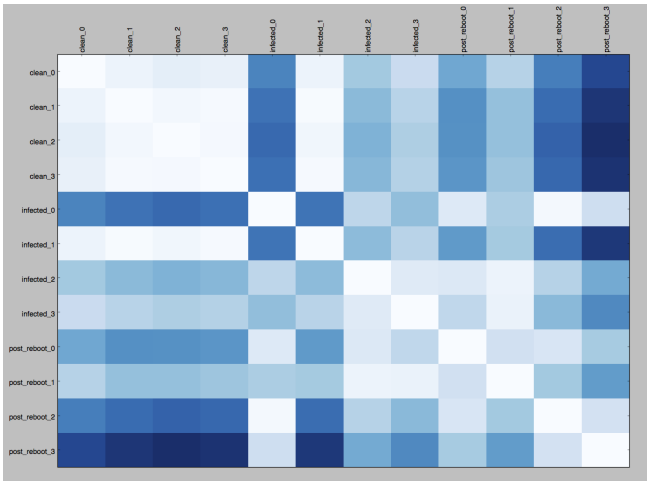

Fig. 1. Absolute KL-divergence heatmap

In conclusion, we have proved that malware, in particular rootkits, are detectable via analysis of their power profile. Future work entails testing on a variety of malware samples and investigating a workflow that can trigger rootkit actions without a priori knowledge of the malware's intent.

References

[1] Y. Ye, T. Li, Y. Chen, and Q. Jiang, "Automatic malware categorization using cluster ensemble," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 95–104.

[2] P. O'Kane, S. Sezer, and K. McLaughlin, "Obfuscation: The hidden malware," *Security & Privacy, IEEE*, vol. 9, no. 5, pp. 41–47, 2011.

[3] S. Yu, S. Zhou, and R. Yang, "Detecting malware variants by byte frequency," *Journal of Networks*, vol. 6, no. 4, pp. 638–645, Apr. 2011.

[4] P. Vinod, P. Rakesh, and G. Alphy, "Similarity measure for obfuscated malware analysis," *Information Security in Diverse Computing Environments*, p. 180, 2014.

[5] M. Narouei, M. Ahmadi, G. Giacinto, H. Takabi, and A. Sami, "Dllminer: structural mining for malware detection," *Security and Communication Networks*, vol. 8, no. 18, pp. 3311–3322, 2015.

[6] J. M. Hernández, A. Ferber, S. Prowell, and L. Hively, "Phase-space detection of cyber events," in *Proceedings of the 10th Annual Cyber and Information Security Research Conference*. ACM, 2015, p. 13.

[7] S. Sutherland. (2013) 10 evil user tricks for bypassing anti-virus. https://blog.netspi.com/10-evil-user-tricks-for-bypassing-anti-virus/. Accessed: 2015-03-12.

[8] J. Hoffmann, S. Neumann, and T. Holz, "Mobile malware detection based on energy fingerprintsa dead end?" in *Research in Attacks, Intrusions, and Defenses*. Springer, 2013, pp. 348–368.

[9] S. S. Clark, B. Ransford, A. Rahmati, S. Guineau, J. Sorber, W. Xu, K. Fu, A. Rahmati, M. Salajegheh, D. Holcomb *et al.*, "Wattsupdoc: Power side channels to nonintrusively discover untargeted malware on embedded medical devices," in *HealthTech*, 2013.

[10] C. R. A. González and J. H. Reed, "Power fingerprinting in sdr integrity assessment for security and regulatory compliance," *Analog Integrated Circuits and Signal Processing*, vol. 69, no. 2-3, pp. 307–327, 2011.

[11] M. M. P. Center. (2007) Win32/alureon. [Online]. Available: https://www.microsoft.com/security/portal/threat/encyclopedia/entry.aspx?Name=Win32%2fAlureon

[12] (2009) Mifi. https://en.wikipedia.org/wiki/MiFi. Accessed: 2016-02-13.

[13] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.