ORNL/TM-2022/2732

Heartbeat: Detecting Malware by Periodic Power Signal Injection and Monitoring

Stacy J. Prowell
Joel Dawson
Ali Passian

October 2022

OAK RIDGE National Laboratory

ORNL IS MANAGED BY UT-BATTELLE LLC FOR THE US DEPARTMENT OF ENERGY

ORNL/TM-2022/2732

Cyber Resilience and Intelligence Division

HEARTBEAT: DETECTING MALWARE BY PERIODIC POWER SIGNAL INJECTION AND MONITORING

Stacy J. Prowell
Joel Dawson
Ali Passian

October 2022

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831
managed by
UT-BATTELLE LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

EXECUTIVE SUMMARY

Rootkits and other stealthy malware attempt to conceal their presence on a computer by making changes to the host computer's operating environment. ORNL's Heartbeat technology detects these changes, and thus the malware itself. Heartbeat operates by directly monitoring the DC power consumption of the computer while a set of operations, the "heartbeat," is executed periodically. These operations exercise parts of the operating system that are common targets of malware tampering. The power consumption during these heartbeat events is monitored and then compared to a previously learned baseline, with any significant deviation detected and analyzed. This technology has been tested and validated in a laboratory environment, and ORNL is currently seeking a deployment partner to allow for further in-context development and testing of this technology.

1. MOTIVATION

Traditional malware detection techniques face three critical challenges.

- **Increases in polymorphic malware.** Signature-based detection is the most common technique used to discover malware because it is *fast* and *reliable* with *low false-positive rates*, but it depends on up-to-date signatures for *low false-negative rates.* Polymorphic malware evades signature-based systems by modifying its static files during the installation phase, which greatly increases the likelihood that signature-based antimalware software will flag it as benign [1].

- **Deployment of "fileless" malware.** Fileless malware establishes its presence in memory but does not write itself to traditional persistent storage [2]. This means that the malware may not survive reboot, but also that it leaves very little or no forensic evidence to be discovered by traditional scanning methods. This type of malware is especially concerning for critical systems – like power grid or industrial safety control systems – that require constant uptime and are almost never rebooted.

- **Rise of UEFI and "bootkit" malware.** Very recently new "firmware" malware has been discovered that writes itself to the flash memory of the computer hardware itself, thus executing "underneath" the operating system [3][4][5]. This malware can persist across reboots and can even survive re-installs and is very difficult to detect.

These techniques can be combined, with polymorphic techniques creating a "moving target" for detection, fileless malware techniques used to prevent discovery through traditional on-device files and forensic traces, and bootkit malware to enable persistence for otherwise fileless malware systems.

2. APPROACH

2.1 ANTI-DETECTION LEAVES A TRACE

Unlike other software on a computer, rootkit malware takes actions to hide its presence from detection systems [6]. Some of these are *active* measures where the malware attempts to modify components of the operating system (or even lower-level structures – see "bootkit" malware above) to present false information to malware scanners and thus evade detection. With antimalware controls effectively

defanged, a rootkit can take further steps to establish persistence and enable the installation of other malware executables.

All software, malicious or not, executes code to accomplish its mission, and *unavoidably* consumes power and time. By injecting itself into heavily used system utilities and structures, a rootkit induces a power- and time-consumption overhead that is unexpected and unusual, given normal operations. ORNL's Heartbeat technology isolates and analyzes the power and timing deviation of carefully chosen operating system functions and structures in a tamper- and noise-resistant manner, revealing the presence of malicious OS changes to a remote server or SIEM.

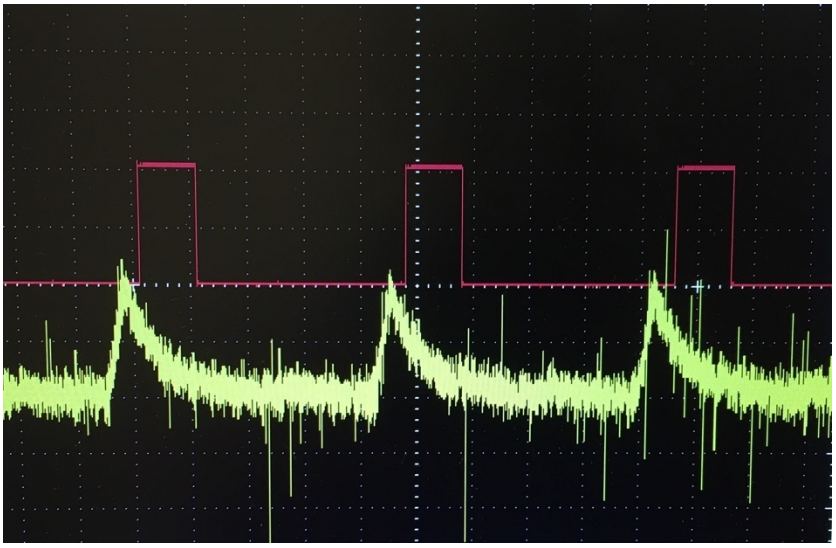

Figure 1 Heartbeat signal (yellow) and lock-in amplifier guide pulses (red)

2.2 HEARTBEAT APPROACH

There have been several attempts to detect malware using power monitoring. These approaches have focused on examining the very noisy power consumption signal to discover anomalous consumption and highlight it. They have met with limited success [7]. ORNL's Heartbeat approach improves on these attempts by taking advantage of several key insights:

- Rootkits create changes in OS behavior that are definable and malware-specific [8]
- By running them periodically with a small program – creating an observable "heartbeat" – targeted system functions can be instrumented in a lightweight, low-privilege manner [9]
- Users can isolate and measure the heartbeat signal from raw power measurements using lock-in amplification [10]

This approach has both advantages and disadvantages. Significant advantages are the following.

- Heartbeat does not rely on scanning files or memory but focuses on detecting an unavoidable consequence of the changes in the operating system used by malware to hide itself. This approach works with existing malware detection approaches and is complementary to them.

- Heartbeat does not depend on *how* a threat is installed, so it works for "zero-day" malware.

- While Heartbeat runs a small amount of code on the machine being monitored, it does so as an *unprivileged user*, so it does not increase the attack surface of the device.

- Heartbeat's detection occurs by monitoring the DC power consumption of a device and is thus *logically isolated* from the device itself. That is, malware operating on the device has no connection to the detection system and no channel to enable it to tamper with detection.

- Because the detection occurs off-device, it can be used to take local actions, including implementing local network filtering and device isolation *on detection*.

Disadvantages of this approach are the following.

- Additional hardware may be required. Some systems may come with power monitoring that can be used by Heartbeat, but for many a specialized power supply or an in-circuit addition of hardware may be required to capture the power signal.

- Additional network connectivity may be required. The detection is performed by a separate single-board computer (SBC) that must itself be connected to the network.

- Additional testing in the relevant deployment environment is needed to better characterize the detection capabilities of the Heartbeat system.

The requirements for additional hardware are low; ORNL is currently investigating a proof-of-concept approach that relies on the use of an Arduino, Raspberry Pi Zero, or other low-cost, low-power device to implement the necessary monitoring.

3. TECHNICAL DETAILS

3.1 INSTALLATION

The Heartbeat technology consists of four primary components:

1. The heartbeat injection code
2. A power sensor (current or voltage)
3. A function generator
4. A lock-in amplifier

The injection code is a small unprivileged software program that executes the heartbeat routine at a fixed interval. The last three are hardware devices that must be deployed into or near the power supply of the monitored system. While all four of these components are necessary for Heartbeat's functioning, we believe that one or more hardware elements can be miniaturized and combined onto a single, low-cost PCB or off-the-shelf computer, which will simplify the hardware installation. The specific nature of this miniaturization effort depends on the target installation.

3.2 OPERATION

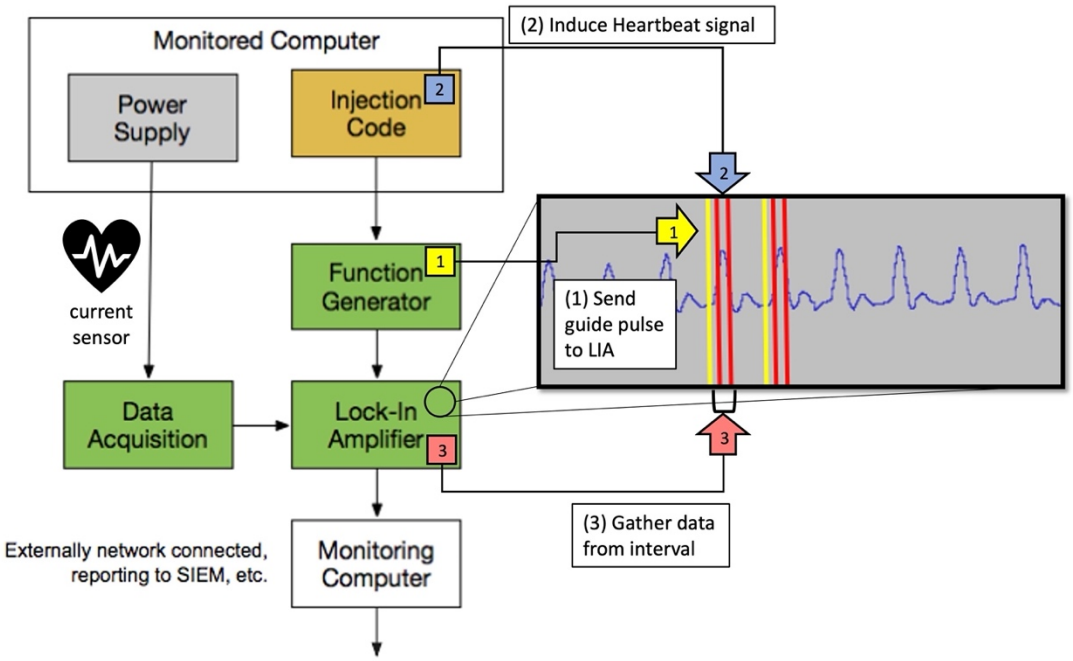

Figure 2 Heartbeat components and operational workflow

Fig. 2 shows a high-level model deployment and workflow for Heartbeat in a generic computer system [8][9][10]. Hearbeat's operation can be described in the following steps:

- **Step 1:** Prior to executing the heartbeat instructions, the Heartbeat code signals the function generator to send a synchronization pulse to the lock-in amplifier.
- **Step2:** The Heartbeat code then runs the specified exercises on the monitored computer, which causes a transient "spike" in its electrical behavior.
- **Step 3:** This spike is automatically read and conditioned by the lock-in amplifier, which then sends the data to a nearby server or SIEM.

By only sampling during a tight interval, the lock-in amplifier filters out most of the power consumption noise that comes with this data source, greatly improving the signal-to-noise ratio of the collected data.

Choosing *which* system functions and structures to run during each heartbeat is critically important to Heartbeat's accuracy and should be informed by the monitored device's architecture and use case, the user's security policies, and knowledge about malware tactics and objectives. Much of this process can be automated during installation.

4. CURRENT STATUS AND POTENTIAL DEPLOYMENT

Heartbeat is currently evaluated at a Technology Readiness Level (TRL) of 4, indicating "Component and/or system validation in laboratory environment" [11]. Testing and development of Heartbeat have taken place in the laboratory environment targeting malware running on a commodity Windows computer. Custom power sensors are placed between the power supply and the motherboard connector,

and “benchtop” hardware (lock-in amplifier, data acquisition board, function generator) used for testing. A number of malware samples have been tested, including Black Energy [12], Sofacy.A [13], SofacyCarberp [14], Greenbug [15], and ZeroAccess [16]. In thie environment, Heartbeat successfully detected all malware samples and was able to distinguish them from the uninfected state.

ORNL is currently seeking a deployment partner to allow for further in-context development and testing of this technology. The specific deployment environment will have a significant impact on how Heartbeat should be developed and deployed. For example, a Linux host will require different heartbeat injection code than a Windows host. Hardware placement and monitoring may be simpler for certain devices and deployments than others. Once a deployment environment is chosen, ORNL is confident Heartbeat can be rapidly evolved and deployed to protect systems in that environment.

REFERENCES

[1] Alrzini, Joma, and Diane Pennington. “A Review of Polymorphic Malware Detection Techniques.” In *International Conference on Interdisciplinary Computer Science and Engineering (ICICSE2020)*. Virtual Event, 2020. https://strathprints.strath.ac.uk/73493/.

[2] Sudhakar, and Sushil Kumar. “An Emerging Threat Fileless Malware: A Survey and Research Challenges.” *Cybersecurity* 3, no. 1 (January 14, 2020): 1. https://doi.org/10.1186/s42400-019-0043-x.

[3] “LoJax, Software S0397 | MITRE ATT&CK®.” Accessed October 5, 2022. https://attack.mitre.org/software/S0397/.

[4] “MoonBounce: The Dark Side of UEFI Firmware.” Accessed October 5, 2022. https://securelist.com/moonbounce-the-dark-side-of-uefi-firmware/105468/.

[5] ComputerWeekly.com. “MosaicRegressor APT Campaign Using Rare Malware Variant.” Accessed October 5, 2022. https://www.computerweekly.com/news/252490098/MosaicRegressor-APT-campaign-using-rare-malware-variant.

[6] Eresheim, Sebastian, Robert Luh, and Sebastian Schrittwieser. “The Evolution of Process Hiding Techniques in Malware - Current Threats and Possible Countermeasures.” *Journal of Information Processing* 25 (2017): 866–74. https://doi.org/10.2197/ipsjjip.25.866.

[7] Reed, Jeffrey H., and Carlos R. Aguayo Gonzalez. Using power fingerprinting (PFP) to monitor the integrity and enhance security of computer based systems. United States US9262632B2, filed November 3, 2011, and issued February 16, 2016. https://patents.google.com/patent/US9262632B2/en.

[8] Prowell, Stacy J., and Christopher T. Rathgeb. Statistical fingerprinting for malware detection and classification. United States US9135440B2, filed July 31, 2013, and issued September 15, 2015. https://patents.google.com/patent/US9135440B2/en?q=prowell&inventor=rathgeb&oq=prowell+rathgeb.

[9] Prowell, Stacy J., Jeffrey A. Nichols, and Jarilyn M. Hernandez Jimenez. System and method for monitoring power consumption to detect malware. United States US10685118B2, filed May 15, 2018, and issued June 16, 2020. https://patents.google.com/patent/US10685118B2/en?inventor=stacy+prowell&oq=stacy+prowell.

[10] Dawson, Joel, and Ali Passian. Rootkit detection system. United States US11074345B2, filed May 30, 2019, and issued July 27, 2021.

https://patents.google.com/patent/US11074345B2/en?q=joel+dawson&inventor=ali+passian&oq=joel+dawson+ali+passian.
[11] Frank, Melvin. “Technology Readiness Assessment Guide — DOE Directives, Guidance, and Delegations.” Directive. Accessed October 5, 2022. https://www.directives.doe.gov/directives-documents/400-series/0413.3-EGuide-04-admchg1.
[12] usa.kaspersky.com. “BlackEnergy APT Attacks in Ukraine,” January 13, 2021. https://usa.kaspersky.com/resource-center/threats/blackenergy.
[13] “Justice Department Announces Actions to Disrupt Advanced Persistent Threat 28 Botnet of Infected Routers and Network Storage Devices,” May 23, 2018. https://www.justice.gov/opa/pr/justice-department-announces-actions-disrupt-advanced-persistent-threat-28-botnet-infected.
[14] Lee, Bryan. “Sofacy Attacks Multiple Government Entities.” *Unit 42* (blog), February 28, 2018. https://unit42.paloaltonetworks.com/unit42-sofacy-attacks-multiple-government-entities/.
[15] “Sophisticated Espionage Group Turns Attention to Telecom Providers in South Asia.” Accessed October 5, 2022. https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/greenbug-espionage-telco-south-asia.
[16] Naked Security. “The ZeroAccess Rootkit,” April 4, 2012. https://nakedsecurity.sophos.com/zeroaccess2/.