A Side-channel Framework and Microarchitectural Analysis Application: Ransomware Detection with Hardware Performance Counters

Jennie E. Hill
Department of Electrical & Computer Engineering
United States Naval Academy
Annapolis, Maryland, USA
jehill@usna.edu

ABSTRACT

Side-channels are unintended pathways through which computer systems leak information, originating from physical or microarchitectural behaviors that produce observable phenomena correlated with internal operations. Despite decades of research, the field suffers from inconsistent terminology and classification. This work proposes a structured framework to define and categorize side-channel phenomena based on leakage source and functional activity: Side-Channel Attack (SCA), Side-Channel Analysis (SCAN), and Side-Channel Defense (SCD). The utility of this taxonomy is demonstrated through a case study of microarchitectural SCAN using Hardware Performance Counters (HPCs), which capture low-level CPU events. HPCs are shown to enable passive, low-overhead monitoring for early detection of ransomware and other malware. Ongoing work explores cross-platform applicability, improved event selection, and the integration of machine learning for real-time anomaly detection, advancing the role of HPC-based side-channel analysis in practical security applications.

1 INTRODUCTION

Computers exude information, by design. They have become fully integrated into our lives because we have come to depend on the myriad services computer systems provide. Much of the information that is obtained from computers is intentional, but the physical implementation of computer hardware that provides these services leads necessarily to physical behavior on the part of an operating computer. This physical behavior has physical characteristics, many of which become channels of information leakage that can be observed by an *unintended* receiver. These "side-channels" of computer operations, such as current usage and power consumption, generation of heat and electromagnetic radiation, and events at the micro-architectural level, can be exploited to compromise the confidentiality of a system.

1.1 A Brief History of Side-channels

The identification of side-channels as avenues to gather and exploit valuable information is relatively new - it was 1996 when Paul Kocher used differences in the timing of computer performance optimizations (a function of the chosen micro-architectural implementation) to find the entire secret key of asymmetric encryption algorithms such as Diffie-Hellman, Rivest Shamir Adleman (RSA), and Digitial Signature Standard (DSS) [12], followed closely by analyzing the power consumed during cryptographic operations to extract keys from dozens of products [10]. In the 2000s the National Security Agency declassified the TEMPEST program [22], and electromagnetic signal leakage was identified as another viable side-channel [1]. In the score of years that followed, a number of additional side-channels were commonly accepted: Acoustic, optical, temperature, memory/cache, and micro-architectural; and advantages, disadvantages, and methods of exploiting and securing each are active research areas.

1.2 The Side-channel Terminology Problem

The related nomenclature, however, is highly inconsistent. Side-channel attacks are classified a number of ways: Active vs. Passive, Invasive vs. Non-invasive [18], Simple vs. Differential [26], Profiled vs. Non-Profiled [9], but the terms are often used in overlapping contexts. Some works refer to Side-channel Analysis and Constructive Side-channel Analysis while others refer to the same techniques as passive side-channel attacks and side-channel defense, respectively. Meanwhile, what actually constitutes a side-channel is also ambiguous - with multiple names used interchangeably for the same leakage vector, sometimes compounded by conflating the name of the side-channel with an attack method. "Mutual information," generally considered statistical measure of the amount of information shared between two random variables, is at times considered a metric [19] [25], a side-channel [24], an analysis method [2] [15], and an attack method [6]. There are as many as six different terms used for the similar side-channel methods related to microarchitecture/memory/cache/access/timing/transient execution, with no clear guideline or definition to distinguish between them. The highly publicized Spectre [11] and Meltdown [13] attacks of 2018 help explain this issue. In a rapidly evolving field, every novel attack that is identified sends researchers scrambling for a solution. It's not at all surprising that the terminology is inconsistent; the focus is always on reacting to the next threat.

2 A SIDE-CHANNEL FRAMEWORK

This side-channel framework is adapted from [7] and **side-channel** is defined here as:

> A *side-channel* is a pathway of information leakage resulting from the physical behavior or microarchitectural design of computer hardware, observable by an unintended receiver.

Unlike conventional communication channels, side-channels transmit information unintentionally, typically through physical

phenomena or timing variations correlated with system state or internal computation.

2.1 Classification by Leakage Source

Side-channels may be categorized based on the physical modality of leakage:

- **Power side-channels**: Exploit variations in power consumption.
- **Electromagnetic (EM) side-channels**: Measure radiated EM signals.
- **Acoustic side-channels**: Detect sounds or vibrations produced by hardware.
- **Temperature side-channels**: Monitor temperature variations.
- **Optical side-channels**: Light emission by a system (e.g., LED patterns, photon release).
- **Microarchitectural side-channels**: A broad category of side-channels based on the underlying hardware implementation of a processor, which exploit the performance-enhancing behaviors of CPU internals (e.g., caches, TLBs, branch predictors). Commonly sub-divided into timing and cache-based side-channels, as described in [20].

A single leakage modality may be accessed through multiple techniques. For example, temperature leakage may be observed via either internal sensors or external thermal imaging; both methods access the *temperature side-channel*.

2.2 Functional Categories of Side-Channel Activity

Side-channel activity can be categorized into three functional roles:

- **Side-Channel Attack (SCA)**: The exploitation of side-channel leakage to extract secrets or gain unauthorized system access.
- **Side-Channel Analysis (SCAN)**: The observation and analysis of side-channel leakage for non-invasive information gathering, monitoring, or threat detection.
- **Side-Channel Defense (SCD)**: Techniques intended to reduce side-channel leakage or to decorrelate it from sensitive internal states.

Each side-channel supports multiple approaches to Attack, Analysis, and Defense, with specific activities in these categories referred to as *methods*. These methods define how SCA, SCAN, or SCD are implemented. For example, multiple power analysis approaches (e.g., simple, differential, correlation) are used in SCA to break encryption [16]. Signal classification using hardware performance counters serves as a SCAN method for identifying anomalous system behavior [8], while cache coloring is a method of SCD aimed at reducing cache-based leakage [14].

3 CASE STUDY: MICROARCHITECTURAL SCAN USING HARDWARE PERFORMANCE COUNTERS

Modern processors are equipped with **hardware performance counters (HPCs)**, which expose low-level execution metrics including cache misses, branch mispredictions, instruction retirements, and cycles stalled. Although originally designed for performance optimization, these counters provide insight into microarchitectural behavior that aligns closely with known side-channel sources [23]. Recent research has explored the use of HPCs as a passive monitoring interface into microarchitectural side-channels. Under the proposed framework, this is categorized as a SCAN method, since it leverages side-channel leakage for detection and diagnostic purposes without interfering with normal system operations.

Multiple studies have demonstrated the effectiveness of hardware performance counters (HPCs) as a side-channel analysis (SCAN) method for identifying cache-based attacks [5, 4] and general malware behavior [21, 17]. These approaches exploit microarchitectural leakage observable through HPCs, such as cache misses, branch mispredictions, and retired instruction counts, to detect malicious execution patterns with minimal overhead.

3.1 Ongoing Research in HPC-Based Ransomware Detection

Recent advancements in ransomware detection have increasingly focused on leveraging Hardware Performance Counters (HPCs) as a robust defense mechanism. This approach capitalizes on the unique microarchitectural "fingerprints" left by ransomware during execution, offering a dynamic analysis method that is more resilient to code variations and zero-day threats than traditional signature-based systems. Researchers are employing machine learning algorithms to classify malicious behavior based on patterns observed in HPCs to achieve early detection with minimal performance overhead, crucial for mitigating the rapid encryption capabilities of modern ransomware.

Ongoing work is focused on refining the results obtained in [8], which collected data from over two hundred distinct hardware events, a significantly larger set than typically evaluated in other literature. The approach demonstrated the viability of classifying ransomware in under two seconds with over 95% accuracy, achieving this impressive performance with as few as 3 hardware event features when using Neural Network and Bagged Tree classifiers. This efficiency highlights the potential for HPC-based solutions to be integrated into endpoint security products with minimal resource consumption.

While HPC-based detection offers compelling advantages like real-time analysis and resilience against unknown threats, challenges remain. These include the need for further accuracy improvements, a deeper understanding of the causal link between hardware events and high-level software behavior, and difficulties in distinguishing ransomware embedded within seemingly benign applications.

Current research includes:

- Gaining confidence in identifying HPC events that are indicative of ransomware activity [3].
- Optimizing event selection for portability across microarchitectures and execution environments (e.g., virtualized vs. non-virtualized).
- Comparing detection performance across non-virtualized and virtualized (VM) environments.
- Expanding datasets of ransomware behavior captured through HPC-observed side-channel traces.

- Applying machine learning models to classify HPC data streams.
- Developing real-time anomaly detectors based on HPCs for embedded and cloud environments.
- Extending analysis to additional processor microarchitectures.

By providing a structured framework to understand and classify side-channel phenomena and their applications, this work clarifies existing terminology and supports more systematic research efforts. The utility of this framework is illustrated through a practical example: the use of Hardware Performance Counters (HPCs) for ransomware detection. This application is categorized as SCAN, leveraging microarchitectural side-channel leakage for passive, low-overhead threat detection. Ongoing work demonstrates that analyzing microarchitectural side-channel data obtained via HPCs may provide effective early characterization and detection of malware while imposing minimal system overhead.

ACKNOWLEDGMENTS

The views expressed in this paper are those of the author and do not reflect the official policy or position of the U.S. Naval Academy, Department of the Navy, the Department of Defense, or the U.S. Government.

REFERENCES

[1] Dakshi Agrawal, Bruce Archambeault, Josyula R Rao, and Pankaj Rohatgi. 2002. The em side—channel (s). In *International workshop on cryptographic hardware and embedded systems*. Springer, 29–45.

[2] Lejla Batina, Benedikt Gierlichs, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. 2011. Mutual information analysis: a comprehensive study. *Journal of Cryptology*, 24, 2, 269–291.

[3] Ryan Binder, Joshua Byun, Dane Brown, T Owens Walker III, and Jennie E Hill. 2025. Building confidence in hardware-based ransomware detection through hardware performance counter event correlation. In *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer, 316–322.

[4] Stefano Carnà, Serena Ferracci, Francesco Quaglia, and Alessandro Pellegrini. 2023. Fight hardware with hardware: systemwide detection and mitigation of side-channel attacks using performance counters. *Digital Threats: Research and Practice*, 4, 1, 1–24.

[5] Marco Chiappetta, Erkay Savas, and Cemal Yilmaz. 2016. Real time detection of cache-based side-channel attacks using hardware performance counters. *Applied Soft Computing*, 49, 1162–1174.

[6] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. 2008. Mutual information analysis. In *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 426–442.

[7] Jennie Hill. 2023. *Evaluation of Selected Side-Channel Analysis Methods for Ransomware Classification and Detection*. Ph.D. Dissertation. University of Maryland, College Park.

[8] Jennie E Hill, T Owens Walker, Justin A Blanco, Robert W Ives, Ryan Rakvic, and Bruce Jacob. 2024. Ransomware classification using hardware performance counters on a non-virtualized system. *IEEE Access*, 12, 63865–63884.

[9] Martin Hutle and Markus Kammerstetter. 2015. Resilience against physical attacks. In *Smart Grid Security*. Elsevier, 79–112.

[10] Paul Kocher, Joshua Jaffe, and Benjamin Jun. 1999. Differential power analysis. In *Annual international cryptology conference*. Springer, 388–397.

[11] Paul Kocher et al. 2019. Spectre attacks: exploiting speculative execution. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1–19.

[12] Paul C Kocher. 1996. Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems. In *Annual International Cryptology Conference*. Springer, 104–113.

[13] Moritz Lipp et al. 2018. Meltdown. *arXiv preprint arXiv:1801.01207*.

[14] Fangfei Liu, Hao Wu, Kenneth Mai, and Ruby B Lee. 2016. Newcache: secure cache architecture thwarting cache side-channel attacks. *IEEE Micro*, 36, 5, 8–16.

[15] Amir Moradi, Nima Mousavi, Christof Paar, and Mahmoud Salmasizadeh. 2009. A comparative study of mutual information analysis under a gaussian assumption. In *International Workshop on Information Security Applications*. Springer, 193–205.

[16] Mark Randolph and William Diehl. 2020. Power side-channel attack analysis: a review of 20 years of study for the layman. *Cryptography*, 4, 2, 15.

[17] Hossein Sayadi, Zhangying He, Hosein Mohammadi Makrani, and Houman Homayoun. 2024. Intelligent malware detection based on hardware performance counters: a comprehensive survey. In *2024 25th International Symposium on Quality Electronic Design (ISQED)*. IEEE, 1–10.

[18] François-Xavier Standaert. 2010. Introduction to side-channel attacks. In *Secure integrated circuits and systems*. Springer, 27–42.

[19] François-Xavier Standaert, Tal G Malkin, and Moti Yung. 2009. A unified framework for the analysis of side-channel key recovery attacks. In *Annual international conference on the theory and applications of cryptographic techniques*. Springer, 443–461.

[20] Jakub Szefer. 2019. Survey of microarchitectural side and covert channels, attacks, and defenses. *Journal of Hardware and Systems Security*, 3, 3, 219–234.

[21] Adrian Tang, Simha Sethumadhavan, and Salvatore J Stolfo. 2014. Unsupervised anomaly-based malware detection using hardware features. In *International workshop on recent advances in intrusion detection*. Springer, 109–129.

[22] [n. d.] Tempest: a signal problem. https://www.nsa.gov/portals/75/documents/news-features/declassified-documents/cryptologic-spectrum/tempest.pdf. Accessed: 2022-03-02. ().

[23] Leif Uhsadel, Andy Georges, and Ingrid Verbauwhede. 2008. Exploiting hardware performance counters. In *2008 5th workshop on fault diagnosis and tolerance in cryptography*. IEEE, 59–67.

[24] Nicolas Veyrat-Charvillon and François-Xavier Standaert. 2009. Mutual information analysis: how, when and why? In *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 429–443.

[25] Hailong Zhang and Yongbin Zhou. 2018. On the exact relationship between the mutual information metric and the success rate metric. *Information Sciences*, 435, 15–25.

[26] YongBin Zhou and DengGuo Feng. 2005. Side-channel attacks: ten years after its publication and the impacts on cryptographic module security testing. *Cryptology ePrint Archive*.