Article Type: Feature

Nowhere to Hide: Monitoring Side-channels for Supply Chain Resiliency

Domenic Forte, *University of Florida, Gainesville, FL, 32611, USA*
Ben Amaba, *Sonatype, Inc., Gainesville, FL 32608, USA*
Cate Richards, *Sonatype, Inc., Fulton, MD 20759, USA*
Jeff Daniels, *Lockheed Martin Corp., Fort Worth, TX, 76108, USA*

Side-channels are nonfunctional characteristics of a program or hardware (HW), such as power consumption, electromagnetic radiation (EM), temperature, timing, or memory consumption, that allow one to infer information about the program, software (SW), or HW. Often, attackers take advantage of these side-channels to uncover secrets from cryptographic systems, web applications, and more. However, in the right hands, side-channel analysis can also be used for anomaly detection where it has several advantages over traditional solutions.

An anomaly is defined as “an occurrence that is different from what is standard, normal, or expected.”[1] Anomalies are caused by the activation of faults. Faults arise due to failure, malfunction, or quality degradation which may be unintentional (i.e., due to design weaknesses, errors, or bugs) or intentional (i.e., due to malware). Anomalies that disrupt normal system operations, exceed safety thresholds, or violate operational constraints can pose operational risks and compromise system reliability. For example, a navigational anomaly from a SW bug caused the 1998 Mars Climate Orbiter mission to fail, resulting in a loss of approximately $125M.[2] Some anomalies can lead to leakage of sensitive data such as 2017’s Equifax data breach.[3]

Rise of Supply Chain Attacks

Third-party, open-source, or commercial-off-the-shelf (COTS) SW and HW are increasingly vulnerable to supply chain attacks and responsible for faults. SW supply chain attacks inject malicious code (i.e, malware) while HW supply chain attacks (i.e., hardware Trojans or HTs[4]) compromise physical components, chips, and electronics or the intellectual property (IP) modules integrated into them. For instance, in 2018, the US Department of Defense started to embrace open-source SW and architectures. Within 2 years, 90% of enterprise systems were composed of open-source SW and the next year witnessed a 742% increase in cyberattacks against them.[5] Examples of attacks on critical infrastructure include the BlackEnergy power grid hack, the Colonial Pipeline cyber incident, and Chernovite’s Pipedream.[6] Nation-state threat actors are not only targeting political and military assets but positioning themselves across civilian infrastructure in order to incite public chaos and panic. A particular risk is the so-called Living off the Land (LOTL) attack, which exploits existing tools within an environment rather than installing new code or scripts. This allows it to evade traditional security measures, enabling hackers to dwell undetected in the victim’s environment for extended periods.[7]

With such infrastructure being increasingly targeted, supply chain vulnerabilities have become a public safety issue and serious efforts are now being put in place to hold SW and HW makers liable. The Cross Sector Cybersecurity Platform Goals by the Cybersecurity and Infrastructure Security Agency (CISA) and the U.S. National Institute of Standards and Technology (NIST) recommend the use of Software Bill of Materials (SBOMs) to document all third-party components used for a given application.[8] SBOMs can support software composition analysis (SCA), periodic scans for newly discovered vulnerabilities, and management policies to address and resolve zero-day exploits before they impact critical systems. Hardware assurance can be achieved in an analogous way.

All that said, BOMs and offline scans are not a panacea for all supply chain attacks.[8] They are only as good as their data, the quality of which can vary by source and method of collection, such as APIs, SDKs, PDKs, and reverse engineering tools. For example, the most common source of information is the National Vulnerability Database (NVD) by NIST. Since February 12, 2024, infrequent NVD updates have left over 2,500

XXXX-XXX © 2024 IEEE
Digital Object Identifier 10.1109/XXX.0000.0000000

vulnerabilities without essential metadata.[9] In addition, security is a “moving target” where malware/HTs that perform the same action can change their appearance to hide from static analyzers, and zero-day vulnerabilities may remain hidden for years. In monitoring the edge, particularly during a critical zero-day situation, determining system status poses ongoing challenges.

Traditional Solutions and Limitations

SW-based malware detection approaches may be divided into two classes: signature-based (or static) vs. behavior-based (or dynamic).[10] Signature-based detection is the most often utilized approach in anti-virus programs. It identifies malware by offline code analysis that finds structures from a predefined malware signature database. Since signatures are from known malware, signature-based methods may be evaded by either constructing new malware or utilizing obfuscation, poylmorphism, or mutation to modify the signatures of existing malware. The signatureless LOTL attacks will also elude them.

Behavior-based approaches execute code within a sandboxed, emulated, or controlled setting where features are logged. For example, a code might be labeled as malware if it modifies host files or registry keys, establishes suspicious network connections, etc. By monitoring behavior, they can identify known malware families as well as potential zero-day attacks and polymorphic variants. However, time complexity and high false positive rate are their weaknesses. False positives can cause catastrophic and life-threatening failures in critical infrastructure. For instance, a false positive for an extreme weather event in a smart grid could result in a shutdown that deprives hundreds of thousands of people of energy, while also compromising vital healthcare services.[6]

SW-based malware detection approaches are also incapable of detecting HTs, which are located in HW rather than SW or program code. These limitations underscore the urgent need for a more integrated HW-SW solutions that address the potential susceptibility of numerous organizations to malicious actors.

The Case for Real-time HW Monitoring

Continuous real-time monitoring and response are desired to identify and mitigate zero-day flaws when they get exploited in the real-world. In this regard, external monitors that measure unintended side-channel leakages are the most promising.[11] First and foremost, unlike malware/HT signatures, operational anomalies are inescapable and cannot be hidden from side-channel measurements. Second, external side-channel monitors can be viewed as separate, trusted, and upgradable “add-ons” to critical systems. Unlike traditional techniques like scanning, sandboxing, and hardware-support, external monitors do not impose burdensome performance, resource, or power constraints on the computational system being monitored.[12] This allows them to be easily integrated into legacy systems that do not possess security infrastructure – a feature important because many critical systems are long-lived. Further, separation from the critical system reduces the possibility that the same supply chain vulnerability affecting the critical system can bypass the external monitor. In this manner, external monitors can assure that a critical system performs its intended function throughout its life cycle.

The Main Challenges and Proposed Solution

Traditionally, side-channel analysis has relied on large, sophisticated instruments such as oscilloscopes and PCs which are large, expensive, and infeasible to deploy outside of a laboratory environment. To resolve this, researchers at the University of Florida developed RASC (short for, remote access to side-channels) an external miniature platform that provides in situ monitoring of a critical system to detect anomalies. Early demonstrations of RASC were able to detect malware in a course-grained manner as well as extract cryptographic keys using power and EM side-channels.[13–14]

In this article, we describe how RASC can perform side-channel disassembly in real-time for fine-grained malware detection. RASC specifically targets anomalies that cause a change in arbitrary instructions and/or sequences of instructions being executed on a processor at run time. In this manner, RASC can effectively overcome the limitations and complement traditional SW-based detection approaches. Further, by operating at run time, RASC can trigger more precise incident response and recovery at the time of attack, thereby avoiding the safety and reliability risks to critical systems as quickly as possible. The rest of this article describes RASC’s capabilities, use cases, prototype, initial results for disassembly and malware detection, and thoughts on future directions.

RASC OVERVIEW

Basic Anatomy of RASC

A critical system’s PCB and RASC are shown on the top left-hand side of Figure 1. RASC would be placed on top of the critical system component under monitor (CSCUM, area highlighted in red). RASC consists of two printed circuit boards (PCBs). The top PCB con-

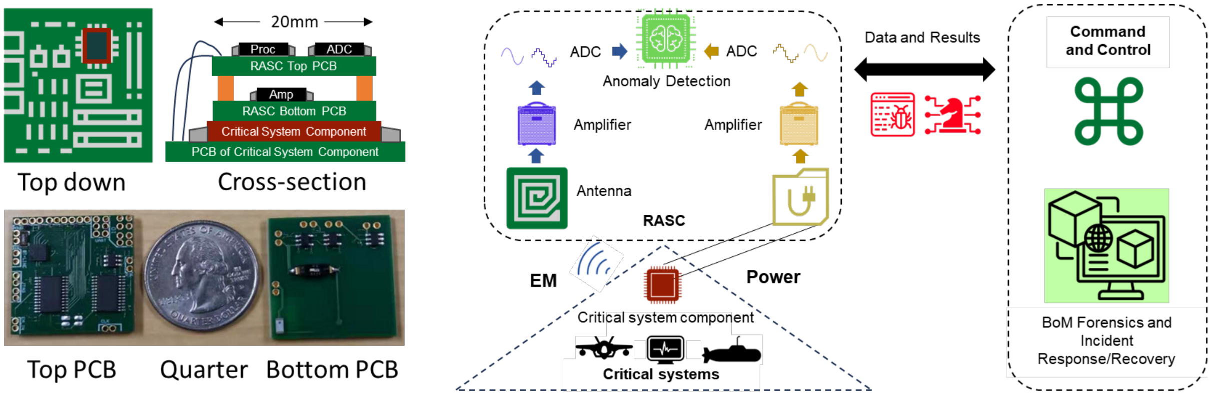

FIGURE 1. (Left) RASC's positioning, anatomy, and dimensions. RASC prototype is about the size of a quarter; (Right) Concept of operation (CONOP). RASC measures power and EM signals of a critical system component under monitor (CSCUM), processes them, performs classification and sends results to an admin or digital twin for forensics and incident response.

tains analog-to-digital converters (ADCs) for digitizing EM and power traces measured from the CSCUM and a field-programmable gate array (FPGA) or microcontroller (MCU) for data processing. This board is also connected to the power supply lines of the CSCUM to measure its power consumption side-channel. The bottom PCB has a magnetic probe/antenna and amplifiers to boost the EM signals emanating from the CSCUM. Depending on the operational environment, RASC could also include clock, communication, and other chips. The bottom left-hand side of Figure 1 compares the size of our RASCv2 prototype's PCBs to a US quarter.

RASC's HW and SW can be trusted and assured. In contrast to the CSCUM, RASC is cheap to manufacture – $100s to $1000s at low volume depending on desired spec – and can be done so onshore by a trusted party. RASC consists of basic commercial-of-the-shelf (COTS) components that can be purchased directly from original component manufacturers (OCMs). Alternatively, a custom ASIC could be developed to implement several RASC functions into a single chip and further reduce RASC's size – this ASIC could be verified as HT-free via reverse engineering. RASC's signal processing, classification, and/or communication SW is simple enough to develop and verify in-house.

Concept of Operation (CONOP)

RASC's CONOP is shown on the right-hand side of Figure 1. RASC simultaneously measures the power and EM of the CSCUM, amplifies these analog signals, and converts them to digital signals. Using these side-channels, its processing unit can perform coarse-grained or fine-grained anomaly detection in real time. In the former approach, the side-channel traces are typically analyzed in the frequency domain where deviations in periodic program activities (loops) can be detected[12]. On the other hand, fine-grained approaches use side-channels to reconstruct the executing code (i.e., disassembly[15]) and compare them to the expected execution. Thus, they are even capable of detecting anomalies in program execution that preserve loop iteration time.

RASC's monitoring capability can provide resilience to supply chain vulnerabilities both before and after identification of zero-day vulnerabilities. In the first case, RASC can continuously monitor the CSCUM and alert administrators, command and control (C&C), and other security personnel when anomalies are detected. An anomaly could indicate either a fault or exploitation of an unknown vulnerability. Unlike traditional information technology (IT) systems, the availability requirements of critical infrastructure forbid their downtime and, thus, limit the amount of time available for forensics and incident response. To investigate anomalies found by RASC, we recommend the use of digital twins (DTs). DTs model real-world physical products, systems, or processes, and regularly synchronize with them. A DT can use disassembly results from RASC to simulate CSCUM dynamics, perform what-if analyses of the malware/Trojan's origin, trigger, or intended payload[1], and provide a test bed for interventions such as containment, eradication, and recovery.[16]

In the second case, RASC can be "awoken" to monitor the CSCUM when a zero-day vulnerability is found in its BOM. Then, if an anomaly is detected by RASC that C&C links to a zero-day exploit, a pre-

prepared response can be triggered immediately as a remedy. This establishes capabilities to further close security gaps by tying the supply chain vulnerability management and continuous monitoring capabilities with the edge and disconnected environments.

Types of Anomalies Detected by RASC

Through side-channel based disassembly, RASC can detect anomalies that alter the instructions executed by a critical system's processor from what is expected from the SBOM. First and foremost, malware injected through buffer overflow attacks, DLL injection, and similar techniques can overwrite arbitrary code or jump to specific locations in memory to achieve their malicious objectives. When such code is executed, RASC can identify each instruction using side-channel analysis and compare the recovered instructions to the authentic SBOM code. Divergence even by a single instruction could be detected in this case.

In addition, RASC can handle return-oriented programming (ROP), LOTL, and HT threats that do not even inject new code into memory as long as they alter program control flow. For example, the A2 HT was developed at the University of Michigan in 2016.[17] The A2's analog trigger is unlikely to be detected by RASC due to its small side-channel footprint. However, its payload – which changes the mode of a CPU from user to supervisor – could result in "privileged “instructions being executed without the system calls, interrupts, and exceptions typically required to do so. By detecting the execution of privileged instructions and the instructions preceding them through side-channels, RASC can identify the unauthorized control flow and therefore the A2's payload activation.

Key RASC Specifications

The most important parameters impacting RASC's anomaly detection capabilities are (in order of importance): (1) *Sampling rate* refers to the number of side-channel samples per unit time. Empirically, we have found that the sampling should be 20-100× the CSCUM's clock rate, ensuring there is enough temporal information for side-channel disassembly; (2) *Sample resolution* refers to the number of bits per sample after ADC conversion. Lower sample resolution of power and EM side-channels may result in a loss of information, potentially making it more challenging to capture subtle variations that could be indicative of class differences and/or anomalies; (3) *Processing capabilities* affect RASC's ability to handle large volumes of measurement data, sophisticated feature extraction and classification algorithms, and adaptation to non-stationary environments. Parallel processing in an ASIC or FPGA can significantly accelerate computations compared to an MCU.

RASC's specs should be chosen based on the complexity of the CSCUM (e.g., clock rate, pipeline depth, or operating system) and the accuracy required for anomaly detection. The higher the sampling rate, sample resolution, and processing capabilities, the higher RASC's cost and power requirements.

RASC PROTOTYPE RESULTS AND DISCUSSION

Prototype and Setup

RASCv3 was fabricated according to the specs shown in Table 1 and used to perform side-channel disassembly and malware detection in real time. Decimal2float, ASCII, and ADConverter benchmarks from AVR-ASM-Tutorial.net were implemented on an *Arduino UNO*, which acted as the CSCUM. Power and EM traces were collected by RASC. Beforehand, profiling was performed to determine linear coefficients to combine the two channels according to their mutual information[14]. Feature dimensionality reduction was performed using the method of minimum redundancy and maximum relevance (mRMR)[18]. A hierarchical classifier[11] was developed using the Quadratic Discriminant Analysis (QDA) algorithm in Xilinx Vivado and loaded onto RASC to classify traces into instructions. Further, to improve accuracy of malware detection, hidden Markov models (HMMs) were utilized.[19]

Disassembly Results

As part of the hierarchical classifier, the entire AVR instruction set was divided into 8 groups of instructions. The plot in Figure 2 displays the group recognition rate according to the number of features used by the QDA classifier. The dashed magenta and red lines compare traditional principal component analysis (PCA) feature selection with mRMR feature selection assuming the use of both power and EM features. PCA compresses all measurement samples into a smaller number of features, but this is expensive to perform in real time. The more efficient mRMR method uses only the subset of features and achieves a similar recognition rate as PCA. Both can obtain a 100% group recognition rate in training with approximately 100 features.

The magenta and black lines in Figure 2 utilize mRMR feature selection with only EM and only power measurements, respectively. Compared to the combined measurements (red), their recognition rates are lower. Further, power and EM recognition rates

TABLE 1. RASCv3 Prototype Specifications

Parameter	Quantity or Range	Chip Used
Cost*	≈ $400	N/A
PCB Dimensions	$(3.8\text{cm})^2$	
Voltage	[-1V, 1V]	ADI LT2242
Speed	250MS/s	
Resolution	1mV, 12 bits	
Amp Gain	10	TI OPA657
Amp Bandwidth	1.6GHz	
Processing	160MHz	Xilinx Artix-7 XC7A100T
Memory	1.6MB	
Clock	167MHz	TI CDCE906

* Cost is specified for low volume purchases of chips and fabrication. It would be lower at higher volumes.

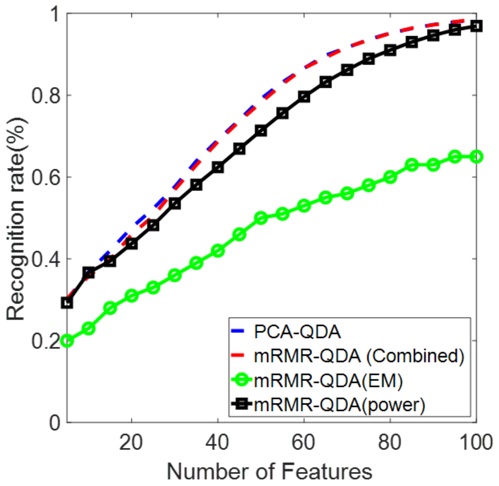

FIGURE 2. Opcode group recognition rate as function of number of features for combined power and EM measurements, for only power measurements, and for only EM measurements. PCA is used for feature selection in the former case while mRMR is used for all three cases.

saturate at 97% and 82% recognition, respectively, even after increasing the features far beyond 100 (not shown). This demonstrates that power and EM contribute unique information that can improve classification accuracy while lowering classifier implementation complexity (i.e., number of features).

The mRMR-QDA (Combined) classifier was implemented in RASCv3 and used to perform disassembly in real time. That is, for each instruction executed by the CSCUM, the opcode was identified from the power and EM trace measurements. The same algorithms were also duplicated in an offline setup where a traditional oscilloscope with 5Gs/s and 16-bit resolution was used to collect power and EM traces. The testing accuracy for three benchmarks for offline and real-time setups is shown in the second and third columns of Table 2. In all three cases, the recognition rate is higher than 80% with the offline setup obtaining about 8.5% improvement over RASC. While not perfect, these results are nonetheless impressive. In contrast to side-channel attacks where 100s of traces may be used for key extraction, disassembly is performed here with a single trace for each instruction. If more than one trace was available (e.g., different iterations of an instruction within a loop), the signal-to-noise ratio (SNR) and recognition accuracy could be increased.

Malware Detection Results

The mRMR-QDA (Combined) classifier was used along with an HMM to distinguish the standard benchmarks from hijacked benchmarks that were modified with malware to alter their control flow. The fourth and fifth columns of Table 2 display percentage of times that non-malicious and malicious codes were correctly classified. For Decimal2float and ADConverter benchmarks, the malware-free and malware codes are perfectly distinguished. As soon as the control flow changes, the malware is detected. Although the disassembly opcode recognition errors for all 3 benchmarks are similar, the HMM could not distinguish between malware-free and malware codes all the time for the ASCII benchmark resulting in a 1% false positive rate. Nevertheless, these results represent a substantial improvement over the ad hoc malware detection approach that only utilized power measurements in prior work[13] – see the sixth and seventh columns in Table 2.

Comparison with Alternatives

The approach taken by RASC bares similarity to both signature- and behavioral-based malware detection approaches while also complementing them. Like signature-based approaches, its main criteria for classification is disassembly. However, RASC dynamically performs disassembly via side-channels. While this more akin to behavior-based approaches, it avoids time complexity because monitoring occurs post-deployment and in real time rather than in a controlled environment. Further, the instructions disassembled by RASC can be combined with DTs to alleviate false positives.

TABLE 2. Disassembly and malware detection testing results. ‘Disassembly Recognition’ columns refer to the percentage rate that opcodes of instructions in the benchmark code are correctly classified using ‘Offline’ (i.e., traditional oscilloscope) and ’Real time’ (i.e., RASC) measurement setups. The ‘Pass’ and ‘Fail’ sub-columns report the percentage rate that a code was detected as non-malware and malware, respectively.

Benchmarks	Disassembly Recognition (%)		Proposed HMM Malware Detection		Prior Malware Detection[13]	
	Offline	Real time	Pass (%)	Fail (%)	Pass (%)	Fail (%)
Decimal2float	91	82	100	0	83	17
Decimal2float (hijack)	N/A	N/A	0	100	0	100
ASCII	89	81	99	1	97	3
ASCII (hijack)	N/A	N/A	0	100	0	100
ADconverter	87	81	100	0	60	40
ADconverter (hijack)	N/A	N/A	0	100	3	97

RASC is not the first approach to side-channel based anomaly detection. Examples where power has been used include the work of Liu et al.[19] and Eisenbarth et al.[15] while examples where EM was used include Zajic et al.[12] RASC is a spiritual successor of these that expands their scope and usefulness by (1) combining power and EM modalities to improve accuracy of disassembly and malware detection, (2) performing side-channel data collection and analysis in situ rather than in a laboratory setting, and (3) performing classification and communicating results to security admins in real time.

KEY POINTS AND PATH FORWARD

Real-time monitoring via side-channels can improve the safety and reliability of critical systems by detecting software and hardware supply chain attacks and supporting incident response and recovery. Such a capability was demonstrated above using the RASCv3 prototype which was able to achieve malware detection accuracy with 100% accuracy and 1% false positives in the worst case. These results are promising but further work is needed.

First, a salient point from the disassembly recognition results is that sampling rate and resolution are not the main limitations. The offline approach only obtains about 8.5% improvement over RASC using the same classification algorithms. This implies that the incorporation of state-of-the-art classification approaches, such as deep learning (DL), are more important for improving classification accuracy. Doing so is challenging due to the resource constraints of real-time monitoring systems. Thus, the path forward requires either more efficient DL algorithms (e.g., obtained through hardware-aware neural architectural search[20]) or more computing resources (e.g., adding a DSP and/or ASIC to RASC) for on-board processing.

Second, few (if any) side-channel monitoring solutions, including RASC, have been tested on more complex CSCUMs, such as 32-bit CPUs and SoCs. The SNR of such targets will be much lower than prior targets due to their multiple cores, larger caches, pipelined execution units, sophisticated instruction sets, higher clock frequencies, and advanced power management schemes (e.g., dynamic voltage and frequency scaling). Addressing the added complexity will require other upgrades to RASC’s key specifications, including sampling rate and resolution, along with more advanced classification algorithms.

Finally, for situations where RASC’s specs and algorithms have already reached their practical limits, such as monitoring low-cost IoT edge devices, other ways to improve detection capabilities should be explored. For example, more complex processing or classification could be migrated in full or in part to a cloud environment using RASC’s wireless communication capabilities. Such tradeoffs should be investigated and the workload between RASC and cloud can be optimized to meet objectives along with real-time constraints. This article also offered the new perspective of using DTs in conjunction with side-channel based disassembly data and I/O logs to reduce false positives and to improve diagnosis capabilities.

ACKNOWLEDGMENTS

The authors would like to thank Yunkai Bai for providing experimental results and figures. This work was partially supported by Army Research Office (ARO) under Grant W911NF-19-1-0102.

REFERENCES

1. E. C. Balta, M. Pease, J. Moyne, K. Barton, and D. M. Tilbury, “Digital twin-based cyber-attack detection framework for cyber-physical manufacturing systems,” *IEEE Transactions on Automation Science and Engineering*, 2023.
2. J. Birkinshaw, J. Bessant, and R. Delbridge, “Finding, forming, and performing: Creating networks for discontinuous innovation,” *California management review*, vol. 49, no. 3, pp. 67–84, 2007.
3. Y. Zou, A. H. Mhaidli, A. McCall, and F. Schaub, “" i’ve got nothing to lose": Consumers’ risk perceptions and protective actions after the equifax data breach,” in *Fourteenth Symposium on Usable Privacy and security (soups 2018)*, 2018, pp. 197–216.
4. M. Tehranipoor and F. Koushanfar, “A survey of hardware trojan taxonomy and detection,” *IEEE Design & Test of Computers*, vol. 27, no. 1, pp. 10–25, 2010.
5. “9th Annual State of the Software Supply Chain.” [Online]. Available: https://www.sonatype.com/state-of-the-software-supply-chain/introduction
6. M. I. Malik, A. Ibrahim, P. Hannay, and L. F. Sikos, “Developing resilient cyber-physical systems: A review of state-of-the-art malware detection approaches, gaps, and future directions,” *Computers*, vol. 12, no. 4, p. 79, 2023.
7. “FBI director Christopher Wray testifies on China’s growing cyber threat against U.S. — 1/31/24,” Jan 2024. [Online]. Available: https://www.youtube.com/watch?v=W-MpWmGg5Kw
8. D. Dooley, M. Ashley, N. Solomon, S. Sawyerr, K. Junčytė, and B. Washington, “The role of SBOMs in software supply chain security,” May 2023. [Online]. Available: https://devops.com/the-role-of-sboms-in-software-supply-chain-security/
9. K. Poireault, “NIST NVD Disruption Sees CVE Enrichment on Hold — infosecurity-magazine.com,” https://www.infosecurity-magazine.com/news/nist-vulnerability-database/, [Accessed 02-04-2024].
10. A. Souri and R. Hosseini, “A state-of-the-art survey of malware detection approaches using data mining techniques,” *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1–22, 2018.
11. J. Park, F. Rahman, A. Vassilev, D. Forte, and M. Tehranipoor, “Leveraging side-channel information for disassembly and security,” *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 16, no. 1, pp. 1–21, 2019.
12. A. Zajic and M. Prvulovic, “Using analog side-channels for malware detection,” in *Understanding Analog Side Channels Using Cryptography Algorithms*. Springer, 2023, pp. 151–209.
13. Y. Bai, A. Stern, J. Park, M. Tehranipoor, and D. Forte, “RASCv2: Enabling remote access to side-channels for mission critical and IoT systems,” *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 27, no. 6, pp. 1–25, 2022.
14. Y. Bai, J. Park, M. Tehranipoor, and D. Forte, “Dual channel EM/power attack using mutual information and its real-time implementation,” in *2023 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. IEEE, 2023, pp. 133–143.
15. T. Eisenbarth, C. Paar, and B. Weghenkel, “Building a side channel based disassembler,” *Transactions on Computational Science X: Special Issue on Security in Computing, Part I*, pp. 78–99, 2010.
16. D. Allison, P. Smith, and K. Mclaughlin, “Digital twin-enhanced incident response for cyber-physical systems,” in *Proceedings of the 18th International Conference on Availability, Reliability and Security*, 2023, pp. 1–10.
17. K. Yang, M. Hicks, Q. Dong, T. Austin, and D. Sylvester, “A2: Analog malicious hardware,” in *2016 IEEE symposium on security and privacy (SP)*. IEEE, 2016, pp. 18–37.
18. H. Peng, F. Long, and C. Ding, “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.
19. Y. Liu, L. Wei, Z. Zhou, K. Zhang, W. Xu, and Q. Xu, “On code execution tracking via power side-channel,” in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 1019–1031.
20. L. Sekanina, “Neural architecture search and hardware accelerator co-search: A survey,” *IEEE access*, vol. 9, pp. 151 337–151 362, 2021.

Domenic Forte is a professor in the electrical and computer engineering (ECE) department at the University of Florida. His current research interests include microelectronics supply chain security, security-aware electronic design automation, and hardware security primitives. He received his Ph.D. in electrical engineering from University of Maryland, College Park. He is currently a senior member of both the IEEE and the ACM. Contact him at dforte@ece.ufl.edu.

Ben Amaba is a federal business developer at Sonatype. He is active in artificial intelligence, data sciences, automation, cybersecurity, and open-source software. He holds a Ph.D. degree in industrial & systems engineering from the University of Miami. He is a Fellow with the IISE, a licensed PE, and a board member position at Operations Sciences Institute (OSI), the University of Miami, University of Central Florida (UCF), National Society of Professional Engineers (NSPE), and other organizations. Contact him at bamaba@sonatype.com.

Cate Richards is the Director of Federal Programs at Sonatype and a technical advisor to the NSPE Software Professional Certification Task Force. Previously, she was with the Synopsys' Software Integrity Group and spent 20 years at IBM. She received her MBA in business administration from Rollins College and can be contacted at crichards@sonatype.com.

Jeff Daniels is the Technology Office and Automation Program Director at the Lockheed Martin Corporation. He is an advisory board member at the OSI and on the board of directors at UCF. He holds a Ph.D. in digital communications from Indiana State University. He can be reached at jeff.daniels@lmco.com.