EM Fingerprints: Towards Identifying Unauthorized Hardware Substitutions in the Supply Chain Jungle

Constantinos Kolias
University of Idaho
1776 Science Center Dr, Idaho Falls
ID 83402
kolias@uidaho.edu

Daniel Barbará
George Mason University
4400 University Dr, Fairfax
VA 22030
dbarbara@gmu.edu

Craig Rieger and Jacob Ulrich
Idaho National Lab
2525 Fremont Ave, Idaho Falls
ID 83402
craig.rieger@inl.gov
jacob.ulrich@inl.gov

***Abstract*—This paper proposes a system capable of branding digital device components based on the EM signals typically emitted during their normal operational cycles. Such signals contain digital artifacts that are unique, which may act as an identifier of a particular device component e.g., its CPU, or the entire device if one chooses to take into account a combination of multiple such components. In real-life scenarios, this "biometrical" fingerprinting of hardware has to be conducted only once, possibly as part of an initial device configuration process with minimum additional maintenance time and cost, by the network administrators. At a subsequent stage, devices can get "authenticated" by comparing their newly emitted signals against the preexisting database during routine checks. The experimental results attest that the proposed approach can effectively protect a network against unrecognized potentially rogue devices posing as benign or malicious substitutions of hardware components at the chip level with near-perfect accuracy. One may view the proposed system as a technical solution to verify the trustworthiness of digital parts as well as the actors involved in certain stages of the supply chain.**

I. Introduction

Modern corporate, government, military, and critical infrastructure networks consist of a myriad of digital devices typically purchased from well-trusted domestic manufacturers. Nevertheless, in practice, each one of these devices is nothing but a mere host of countless microscopic digital components produced by potentially untrusted vendors from all over the world. From an adversarial point of view, the jungle of supply chain relationships and interactions is a fertile ground of opportunities for an attacker to corrupt a system and alter its intended operations. Indeed, an attacker may operate at the design, manufacturing, distribution, or maintenance stages of a device's lifecycle. Attacks materialized by taking advantage of the complex supply chain dynamics typically aim at the addition or substitution of digital components of a device, e.g., the CPU, memory chips, or capacitors. Typically such attacks aim in (a) leakage of sensitive information, (b) persistent system access, or (c) total system failure under certain conditions.

Supply chain attacks are much more effective than traditional software-based malware because they mainly operate at the lowest level of a system, i.e., the hardware. At that level, a simple patch or a software update will not necessarily solve the problem, and complete substitution of entire product lines may be the only practical solution. Moreover, such attacks are much more stealthy because the hardware typically enjoys the complete trust of the user with even downright malicious operations often being perceived as "strange but normal". This problem is rooted deep even into the research community. Considering the example of industrial settings, existing resilience Operation Technology (OT) notional benchmarks [30] fail because they usually consider an adversary as external (or internal) entity that actively tries to undermine normal system behavior and reduce its resilience. It is apparent, that today such notions need to be updated to include the cases in which a critical component of the system lacks inherent resilience due to supply-chain compromises.

This work focuses primarily on the maintenance stages of the supply chain lifecycle. More specifically, we attempt to deal with unauthorized substitutions of entire devices with their malicious "clones" or replacements of on-board components. Such actions may be carried out by malicious insiders or members of an outsourced IT-support team, for example. We propose a set of methods, tools, and a system for providing authentication (and identification) for digital devices. The proposed approach capitalizes on the analog signals and, more specifically, the Electromagnetic (EM) signals that get naturally and involuntarily emitted by digital components during their usual operational cycles, e.g., network modules, CPUs, or other chips. Relevant works [5], [6], [9], [36], [37] have identified that such signals carry unique characteristics due to the subtle variations of the corresponding hardware components. This is the result of the minimally imprecise manufacturing processes even among products of the same production line. The system compares these analog emissions obtained at runtime with an "analog-profile", which is collected/constructed apriori during an off-line step using a two-phase Machine Learning (ML) driven process. In this way, it becomes possible to distinguish and identify each one and then technically enforce authentication and access control. Typically, in managed networks, any device has to pass from a thorough configuration step before it is allowed to join a network for the first time. The profile construction phase can be defined as an additional subprocess of the configuration step. In this way, the expected overhead is minimal. This

DOI 10.1109/SPW50608.2020.00040

approach could be implemented as a portable hand-held reader, which can interrogate the authenticity of devices inside the network during random or routine security checks.

At the lowest level, the proposed system can distinguish between various benign devices inside the protected network towards providing device-level authentication and access control. This is an often overlooked function as most modern authentication/access control methods focus on the human element of the network and neglect the device element. Secondly, it can be used to identify and reject "malicious clones" of devices, i.e., devices that look identical to benign but perform some additional operations at the hardware level under certain conditions (but not always). Finally, it can detect fine-grained unauthorized hardware alterations of a benign device, e.g., the substitution of a chip such as the CPU that could potentially perform additional operations.

The evaluation of the proposed approach was conducted upon test subjects with constrained resources in terms of processing power and memory, namely, a set of Internet-of-Things (IoT) devices. This decision was made because (a) the security of such systems is often overlooked; (b) they are more simple and as such they can be studied more easily; (c) devices of similar characteristics are Operations Technology (OT) systems and Industrial Control Systems (ICS), which are building blocks of most mission-critical networks. While we aspire that this work will set the foundations towards high-end (servers, personal computers, and smartphones) device authentication, the proposed system is immediately applicable to a large number of devices residing in industrial environments, hospitals, or the power grid.

The next section presents relevant work in the field. In section III, we present the overall system architecture and explain its advantages, limitations, and design assumptions that were made. Section IV analyzes the experimental results obtained by a proof-of-concept (PoC) implementation of the system. Section V concludes and outlines the plans for future improvements.

II. Previous Work

The side-channels formed during the normal operation of digital devices have been abused for adversarial purposes, such as to achieve leakage of private information [11], [24], the inference of cryptographic keys [10], [12], the tracking of users [7] or the unleashing surgical attacks [17], [18].

A more recent stream of research leverages on the descriptive power of these analog signals for protective purposes instead. The majority of such works are oriented towards achieving anomaly detection or verifying the control flow integrity at the software level. In this context, several alternative modalities have been considered, e.g., (a) the analysis of power consumption of the device [13], [14], [28], [29] or (b) the analysis of radiant EM [1], [4], [15], [19], [34] signals. Each of the approaches has its advantages, with the former being able to profile the behavior of the device as a whole and the latter being capable of providing a higher level of granularity to individual components of the device.

Side-channel analysis based approaches have also been used for identification and authentication purposes. Traditional means of device authentication in computer networks take place primarily in the link and application layers of the OSI stack. Such mechanisms are not appropriate for device, let alone component-level based authentication because they are tightly coupled with a particular user. For example, a device's MAC address is typically used in combination with a shared secret known by humans to derive authentication keys in layer-2 protocols such as 802.11 (WiFi) [23]. Nonetheless, shared secrets can always leak. Tag-based identification (bar code and RFID) are not effective because they are detached from the benign product, replaced or fabricated to identify a malicious.

A series of works [5], [6], [9], [36] seeks to identify immutable characteristics of devices to provide authentication. Towards this end, the authors rely on the same theoretical foundations like the ones used in this work but restrict their analysis in signals that are part of the standard wireless communication channels, e.g., WiFi. An obvious problem associated with these approaches is that they only take into account the *active* communication channels. In other words, authentication is conducted upon the signals produced voluntarily by the device and only when it attempts to communicate thus, making it less flexible and dynamic. Moreover, one may argue that this authentication is focused only on the Network Interface Card (NIC) component rather than the entire device.

In [37], researchers relied on a near field probe to capture the low-frequency emanations of various digital devices. They employed cosine distance to rank the dissimilarity of the corresponding vectors and subsequently distinguish between alternative devices with accuracy that ranged from 100% to 72% depending on the device type. Nevertheless, the particular work focuses primarily on providing device identification as an alternative to RFID technology. The potential of this method as a means for delivering device authentication, as well as its robustness against possible adversarial efforts was not explored.

A recent work [3] introduced a request-response-driven device fingerprinting technique that relies on magnetic induction (MI) signals emitted by the CPU of devices. The results indicate that the proposed system, namely DeMiCPU can achieve near-perfect (99.1% precision and recall on average), with a fingerprinting time of just 0.6 seconds. An obvious shortcoming of the particular system is that it is stimulus-based, i.e., it fingerprints the behavior of the CPU when interrogated in a specific way. This is an active way of fingerprinting the device that assumes collaboration of the subject.

Works [5], [6], [9], [36], [37] provide proof that subtle hardware differences due to the manufacturing process are carried over to the EM domain and can be an effective way for device identification. Our proposed system builds on these principles and applies sophisticated ML techniques upon EM side-channel artifacts emitted by device components of interest. This allows us to extend the concept above for purposes of authentication (not just identification), in a passive

and external way, with minimum false positives.

III. Proposed Approach

The proposed system takes advantage of the fact that during a device's normal operation, its digital components emit analog signals continuously and involuntarily [13]. Different instructions/operations at the machine level draw different amounts of current when being executed, which in turn results in the formation of EM signals with observable characteristics. As a matter of fact, these artifacts are so descriptive that researchers [8], [26], [35] have taken advantage of them to create external, instruction-level disassemblers. Interestingly, these signals bear subtle differences even if they get emanated by components of the same manufacturer, model, product line, and run the same software [37]. The discrepancies noticed in the EM spectra is the natural outcome of random hardware variations caused by imperfections of the manufacturing process. These unique variations end up distinctly modulating the signals. Therefore, in theory, it may be possible to capture and analyze EM signals with the purpose of (a) distinguishing between different devices on the network, (b) detecting modifications applied at the hardware profile of a machine. The latter might be an occurrence of malicious activity performed at the *maintenance stages* of the supply chain lifecycle, i.e., after the deployment of that device in the protected network.

Applying this approach in the realm of high-end computer systems may be a rather challenging task due to the level of architectural complexity. Indeed, it is hard to fingerprint a process of interest in servers, desktops, and laptops because the amount of random artifacts generated by all interfering processes running in parallel is expected to be high. Nevertheless, the case of IoT devices is of particular interest since they typically adopt a much simpler design. Moreover, a typical execution cycle of IoT devices happens repetitively and is comprised of a limited number of branches. Thus, exhaustively fingerprinting an operation/process in such systems is a comparatively more straightforward task. For these reasons, we have chosen to focus our efforts on the IoT device domain for providing a PoC implementation.

The system aims to capture multiple samples of signals corresponding to normal execution cycles to statistically construct a baseline/model of *normal* modes of operation. Any deviation from the baseline can be seen as an anomaly, which in this context translates to *unknown device/component* (or even *unseen operation*). Thus, the system effectively provides hardware-level device/component authentication. This type of EM-based branding draws inspiration from and resembles the biometrical authentication techniques that have been applied to humans because they are both based on immutable, inherent characteristics of their subjects.

A. System Architecture

From a 1000 foot view, the proposed system is comprised of (a) an EM sensor for capturing continuous signals from the signal source, (b) an oscilloscope for storing discrete samples of the signals in a database, and (c) an analysis engine that performs ML-based analysis upon the stored signals. Notice that the entire system is completely external to the subject device and does not assume any software being installed in it.

The component of particular interest is the analysis engine. It employs a combination of both supervised and unsupervised ML techniques operating in two layers to (a) create a model of benign devices performing legal operations, (b) to raise an alert for unrecognized devices.

At the first layer of analysis and during the system's training phase, the baseline of all normal devices and normal operations per device is created using the Local Outlier Factor (LOF) algorithm [2]. LOF is an unsupervised ML method. Unlike clustering methods, LOF attempts to discover outliers in a given set of data. According to Hawkins [16] an outlier is an observation that deviates so much from other observations as to arouse suspicion that a different mechanism generated it. LOF assigns an outlier score to each observation in the dataset. The score depends on how isolated the object is concerning the surrounding neighborhood. During the deployment phase, signals corresponding to malicious devices will be flagged as *outliers* and may be discarded/dropped by the administrators. The rest will be treated as *normal* observations, but at that point, no further details will be known concerning the identity of the device.

Therefore, full authentication, including the device identification step cannot be applied at the first stage. A second layer of analysis aims to solve this problem by incorporating supervised ML methods. More specifically, by taking advantage of a model trained independently by the K-Nearest Neighbors (KNN) algorithm [27], the system is able to classify new observations into one of the existing devices effectively.

At this point, we should make clear that we have assumed that initial fingerprints of the protected device do not contain any malicious components by default. This is a strong assumption. For further details regarding this assumption, the reader should refer to subsection III-C).

A high-level overview of the described system is given in Figure 1.

B. Threat Model

We assume that the attacker has physical access to a device and can (a) either completely replace the equipment with one that resembles the original but performs malicious software or hardware-based operations, or (b) physically alter individual hardware modules of it. While these two activities require physical access to the device, we also assume a third alternative scenario in which the attacker can modify processes running on the device stealthily and remotely to include malicious operations, say by taking advantage of software vulnerabilities (e.g., a buffer overflow) or network protocols. The latter case has been explored in our previous work [22].

C. Assumptions

The underlying assumption is that during their fingerprinting phase, devices are completely benign, i.e., they do not execute any malicious software, and the hardware modules of the

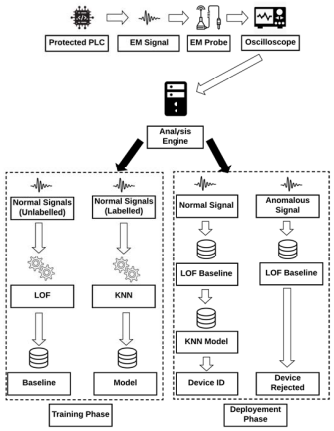


Fig. 1: System Architecture

devices are free of any malicious circuitry. This is a strong assumption because, in real-life situations, the attacker has multiple opportunities to pollute the hardware with malicious components, in the early stages of the supply chain lifecycle. Nevertheless, in this work, we focus on the maintenance stages during which a malicious insider may tamper with a device during its scheduled maintenance, and hardware compromises at earlier stages of the supply chain lifecycles will be the topic of the future research.

Another assumption is that the fingerprinting must be complete, i.e., all alternative execution paths of the device must be fingerprinted to be recognized as benign. For this reason, the experiments are restricted to a more straightforward use-case, i.e., the domain of limited-computational-resources smart-devices.

Finally, all the experiments have been conducted upon devices whose similarities are extensive, i.e., (a) devices of the same model/type manufactured by different vendors, and (b) devices of the same model/type from the same manufacturer. While the test subjects did not contain any malicious hardware components, we assume that if the system is capable of recognizing differences at that level, then it will also be capable of identifying the less subtle modifications required to achieve the malicious functionality.

D. Advantages

There are several advantages to the proposed approach:

- It can provide identification, i.e., it is capable of distinguishing between different devices as long as a fingerprint of such devices exists in the database. It can do so, even if the devices are of the same model of the same manufacturer.
- It can provide authentication, i.e., distinguish between benign devices from those that pose as such.

TABLE I: Main components of the PoC implementation of the system

Component	Purpose
Beehive's 100A EMC Probe	EM probe
Picoscope 3205a	Oscilloscope
Beehive 150A EMC Probe amplifier	Signal Amplifier
Apple MacBook Pro	Analysis Device

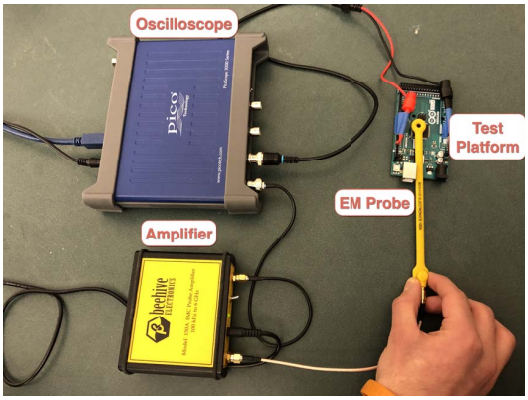


Fig. 2: Experimental setup

- The detection of unauthorized substitutions is performed externally, does not require the installation of any software in the CPU of devices, and does not interfere with its normal operation in any way.
- In contrast to a series of works relevant to side-channel based anomaly detection [20], [21], [25], [32], [33] the proposed system is not only capable of detecting software modifications, but it is also capable of tracking down hardware-based differences. This can be achieved using the same equipment and by applying minor tweaking to the algorithm parameters. The former case was the focus of our earlier works [22], while the latter is the main objective of this work. In comparison, this case is theoretically more challenging in terms of detection, as the differences in the signal morphology are much more subtle.

IV. Experimental Results

A. Experimental Setup

For a PoC implementation of the proposed system, we relied on inexpensive off-the-shelf components. For the capturing equipment, we used a Beehive 100A EMC magnetic field probe, which was placed directly on top of the CPU while it performed normal operations. For storing samples of the signals, we relied on a Picoscope 3205a oscilloscope. The evaluation was completed on an Apple MacBook Pro laptop (CPU 2.5 GHz, RAM 16 GB). All experiments were implemented as Python v3.6.1 scripts. The experimental setup can be seen in Figure 2, while a full list of the system components, including connectors and additional devices, is given in Table I.

All test subjects during the experiments were devices of a well-known prototyping IoT platform, namely, Arduino

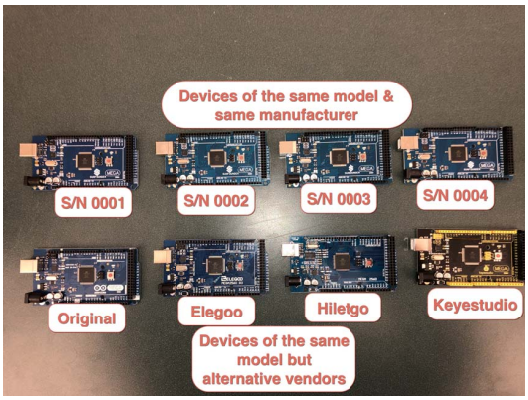


Fig. 3: Alternative devices used as test subjects

TABLE II: Technical characteristics of the chosen platform

Characteristic	Value
Operating Voltage	5V
Input Voltage	7-12V
Flash Memory	256KB
SRAM	8KB
EEPROM	4KB
Clock Speed	16Mhz

Mega 2560. The particular model is equipped with an 8-bit Atmel ATmega 2560 processor clocked in 16Mhz. The full technical characteristics of the platform are given in Table II. Depending on the experiment, devices from the original or third party manufacturers (i.e., Sunfounder, Elegoo, HiLetGo, Keyestudio) were used. All the alternative devices used as test subjects during the data-gathering phase are presented in Figure 3.

B. Dataset Description

We collected two sets of data. Dataset A contains signals from five Arduino devices of the same model (Mega 2560), but alternative manufacturers running the same code. Dataset B contains signals from four Arduino devices of the same model, i.e., Mega 2560 and of the same manufacturer. The source code of the software running on the test subjects during the data gathering is given in Listing 1. Both the datasets contain 10,000 distinct signals per device. Thus, each dataset contains 40,000, unique signals. Each one of these signals is comprised of 12,500 samples (features/dimensions). Examples of signals of four alternative devices contained in datasets A and B are provided in Figure 4 and 5, respectively.

C. Evaluation Method

We conducted experiments from subsets obtained by the two datasets. For each dataset, ten splits of 4,000 signals were used (remember each dataset contains 40,000 unique signals in total). All signals in each split were randomly selected with the non-replacement method. Three (out of four) devices were considered benign clients of the network during the training phase. Instances of the fourth device were not included in the training set; thus, these instances were treated as malicious/outliers.

TABLE III: Confusion Matrix and basic metrics corresponding to the LOF evaluation on Dataset A

		Predicted	
	Type	**Unknown Device**	**Benign Device**
Actual	**Unknown Device**	980	0
	Benign Device	63	2877
ACC		0.983	
F1		0.979	
AUC		1.0	
Duration		3.93 sec	

We evaluated the results of the first round in terms of accuracy (ACC) and F1 score and area under the curve (AUC) score and the results of the second round in terms of ACC and F1 score only.

A ROC curve is a graph of the true-positive rate (TPR) against the false positive rate (FPR) for all possible thresholds returned by the algorithm. Since two ROC may have non-standard shapes that make their comparison hard, the most common metric of comparing two ROC curves is the area under the curve (AUC). An AUC score of 1 is the optimal value, and it implies that the system yields TPR = 1 for any threshold chosen, while the FPR ranges from 0 to 1, depending on the threshold. Accuracy is defined as:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

F1 score is defined as the harmonic mean of precision (PPV) and sensitivity (TPR) as:

$$F1 = 2 * \frac{PPV * TPR}{PPV + TPR} \tag{2}$$

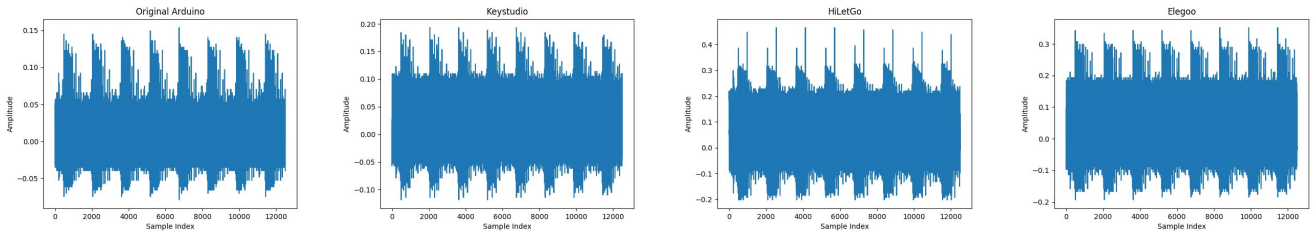
where PPV is defined as $PPV = \frac{TP}{TP+FP}$ and $TPR = \frac{TP}{TP+FN}$

We extensively experimented with the two most sensitive parameters, namely, the number of training instances as well as the number of neighbors considered by the two algorithms. By relying on merely 20 signals from each device for training purposes (creation of the baseline and training the classifier), the system yields steadily near-perfect accuracy (ACC) and F1 scores, and perfect AUC scores, for any split of any of the two datasets.

We assume that if the system can successfully discriminate between devices of such a level of similarity, then it is valid to expect the same or better predictive accuracy with devices of different CPU architectures, other hardware characteristics, or malicious modifications.

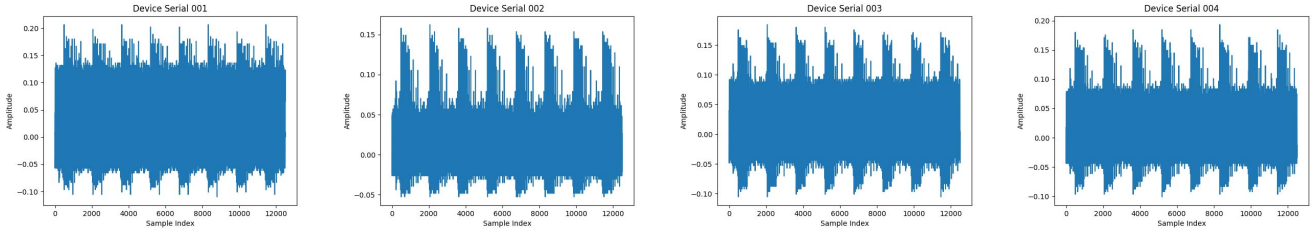
The confusion matrix (i.e., a table of TP, FP, TN, FN) along with the ACC and F1 scores achieved for the dataset A are given in Table III for the unsupervised step and in Table IV for the supervised step. The same metrics for Dataset B are presented in Tables V and VI. Notice, that all values are averaged for the ten splits considered in these datasets.

The reader may notice that the AUC score reported for the first phase of the analysis is perfect, while the corresponding



(a) Original (b) Keystudio device (c) HiLetGo device (d) Elegoo device

Fig. 4: Sample signals obtained from devices of the same model but of different manufacturers. Notice, that despite an overall similar signal phenotype, differences in the signal morphology can be observed even by the naked eye.



(a) Serial number 001 (b) Serial number 002 (c) Serial number 003 (d) Serial number 004

Fig. 5: Sample signals obtained from devices of the same model and the same manufacturer. Notice, that differences can be observed even by the naked eye.

TABLE IV: Confusion Matrix and basic metrics corresponding to the KNN evaluation on Dataset A. Notice that instances obtained by the device from vendor Keyestudio were considered anomalous/unknown and were successfully discarded during the unsupervised step of the process.

		Predicted			
	Manufacturer	**Original**	**Elegoo**	**Hiletgo**	**Keyestudio**
Actual	**Original**	980	0	0	N/A
	Elegoo	0	980	0	N/A
	Hiletgo	0	0	917	N/A
	Keyestudio	N/A	N/A	N/A	N/A
ACC		1.0			
F1		1.0			
Duration		46.88 sec			

TABLE V: Confusion Matrix and basic metrics corresponding to the LOF evaluation on Dataset B.

		Predicted	
	Type	**Unknown Device**	**Benign Device**
Actual	**Unknown Device**	980	0
	Benign Device	16	2924
ACC		0.995	
F1		0.994	
AUC		1.0	
Duration		3.21 sec	

ACC and F1 scores are less than perfect (i.e., 98.3% and 97.9% for Dataset A and 99.5% and 99.4% for Dataset B). The reason behind this ostensible discrepancy is that the ROC curve (and as a result, the AUC score) gets calculated by considering all possible thresholds returned by the LOF algorithm. On the other hand, both ACC and F1 scores were calculated according to a single threshold that was chosen automatically by the algorithm, as described in [2]. This indicates that the particular threshold was not optimal and that at least one threshold exists that yields a true positive rate (TPR) of 100% and a false positive rate (FPR) of 0%. Nevertheless, to identify this threshold, one must have apriori knowledge of the contamination degree of the dataset, which in real-life scenarios, is not possible.

TABLE VI: Confusion Matrix and basic metrics corresponding to the KNN evaluation on Dataset B. Notice that instances of device 004 were considered anomalous and were successfully discarded during the unsupervised step of the process.

		Predicted			
	Serial Number	**001**	**002**	**003**	**004**
Actual	**001**	964	0	0	N/A
	002	0	980	0	N/A
	003	0	0	980	N/A
	004	N/A	N/A	N/A	N/A
ACC		1.0			
F1		1.0			
Duration		47.15 sec			

V. Conclusions & Future Work

In this paper, we presented a method and a prototype implementation of a system that can be used for the identification and authentication of devices inside protected networks. The system leverages the EM signals emitted by the CPU (and potentially other hardware components) during its operation.

```
void loop(){
    noInterrupts();
    digitalWrite(PIN13, LOW);
    digitalWrite(PIN13, HIGH);
    for(uint i=0; i<8; i++){
        a = 0x00000000;
        b = 0xffffffff;
        a |= b;
        b = 0x00000000;
        a &= b;
        b = 0xffffffff;
        a = a^b;
    }
    interrupts();
}
```

Listing 1: Source code of the programs executed in the test subjects

It exploits the subtle differences of these signals, which are a result of minimal hardware variations caused during the manufacturing process. Such differences are random, unique, immutable, and hard to counterfeit. For this reason, we loosely describe this approach as a "device-biometrical-authentication" method. The proposed approach can be applied in high-value networks to protect from unauthorized substitutions, or modifications of hardware components of smart-devices by insiders or (theoretically) trusted external parties during the post-deployment stages of the supply chain, e.g., the maintenance cycles.

Unlike the majority of relevant works, the system relies solely on involuntarily emitted EM signals and does not assume the installation of any software. Therefore it can be applied as an external protection mechanism. Moreover, unlike visual inspection methods [31] applied for the discovery of hardware trojans, the system may be compiled even with inexpensive components and commodity equipment.

The experimental results indicate that the proposed analysis method yields near-perfect scores when used for the identification and authentication of devices. What is more, it achieves that level of accuracy with a minimal number of signals (20 signals per device suffices) for the construction of the baseline. The actual authentication stage is also extremely fast and lightweight, requiring less than a minute to evaluate a batch of nearly 4,000 signals on a modern laptop. This is enough to satisfy hundreds if not thousands of simultaneous authentication attempts.

Future improvements of the system will be oriented towards:

- Assuming early-stage device compromise - perform anomaly detection without putting trust or making any assumptions about the security of a newly deployed device on the network. The particular device may have been compromised during the early stages in the supply chain;
- Increasing the distance of the monitor - make use of alternative off-the-shelf equipment such as directional antennas that permit the capturing of signals from a greater distance and in a non-intrusive manner, i.e., through device enclosures;
- Platform complexity - extend the application of the same concept to support high-end devices such as smartphones, laptops and desktops systems;
- Integration with intrusion detection tools - integrate the proposed system with tools that detect malicious modifications at the software-level to provide holistic system protection.
- Adversarial signal transmission - stress test the effectiveness of the system under sophisticated adversarial activity, possibly crafting and transmitting signals of choice using defined radios (SDR).

This work aspires to be the first step towards a robust technical means for reinforcing the trustworthiness of the several stages and actors involved in the supply chain of digital devices.

References

[1] Nathaniel Boggs, Jimmy C Chau, and Ang Cui. Utilizing electromagnetic emanations for out-of-band detection of unknown attack code in a programmable logic controller. In *Cyber Sensing 2018*, volume 10630, page 106300D. International Society for Optics and Photonics, 2018.

[2] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pages 93–104, 2000.

[3] Yushi Cheng, Xiaoyu Ji, Juchuan Zhang, Wenyuan Xu, and Yi-Chao Chen. Demicpu: Device fingerprinting with magnetic signals radiated by cpu. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1149–1170, 2019.

[4] Shane S Clark, Benjamin Ransford, Amir Rahmati, Shane Guineau, Jacob Sorber, Wenyuan Xu, and Kevin Fu. Wattsupdoc: Power side channels to nonintrusively discover untargeted malware on embedded medical devices. In *Presented as part of the 2013 {USENIX} Workshop on Health Information Technologies*, 2013.

[5] William E Cobb, Eric W Garcia, Michael A Temple, Rusty O Baldwin, and Yong C Kim. Physical layer identification of embedded devices using rf-dna fingerprinting. In *2010-Milcom 2010 Military Communications Conference*, pages 2168–2173. IEEE, 2010.

[6] Gerald DeJean and Darko Kirovski. Rf-dna: Radio-frequency certificates of authenticity. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 346–363. Springer, 2007.

[7] Sanorita Dey, Nirupam Roy, Wenyuan Xu, Romit Roy Choudhury, and Srihari Nelakuditi. Accelprint: Imperfections of accelerometers make smartphones trackable. In *NDSS*, 2014.

[8] Thomas Eisenbarth, Christof Paar, and Björn Weghenkel. Building a side channel based disassembler. In *Transactions on computational science X*, pages 78–99. Springer, 2010.

[9] KJ Ellis and Nur Serinken. Characteristics of radio transmitter fingerprints. *Radio Science*, 36(4):585–597, 2001.

[10] Daniel Genkin, Lev Pachmanov, Itamar Pipman, and Eran Tromer. Stealing keys from pcs using a radio: Cheap electromagnetic attacks on windowed exponentiation. In *International workshop on cryptographic hardware and embedded systems*, pages 207–228. Springer, 2015.

[11] Daniel Genkin, Mihir Pattani, Roei Schuster, and Eran Tromer. Synesthesia: Detecting screen content via remote acoustic side channels. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 853–869. IEEE, 2019.

[12] Daniel Genkin, Itamar Pipman, and Eran Tromer. Get your hands off my laptop: Physical side-channel key-extraction attacks on pcs. *Journal of Cryptographic Engineering*, 5(2):95–112, 2015.

[13] Carlos Aguayo Gonzalez and Alan Hinton. Detecting malicious software execution in programmable logic controllers using power fingerprinting. In *International Conference on Critical Infrastructure Protection*, pages 15–27. Springer, 2014.

[14] Carlos R Aguayo Gonzalez and Jeffrey H Reed. Power fingerprinting in sdr & cr integrity assessment. In *MILCOM 2009-2009 IEEE Military Communications Conference*, pages 1–7. IEEE, 2009.

[15] Yi Han, Sriharsha Etigowni, Hua Liu, Saman Zonouz, and Athina Petropulu. Watch me, but don't touch me! contactless control flow monitoring via electromagnetic emanations. In *Proceedings of the 2017*

ACM SIGSAC conference on computer and communications security, pages 1095–1108, 2017.

[16] Douglas M Hawkins. *Identification of outliers*, volume 11. Springer, 1980.

[17] Mohammad A Islam, Shaolei Ren, and Adam Wierman. Exploiting a thermal side channel for power attacks in multi-tenant data centers. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1079–1094, 2017.

[18] Mohammad A Islam, Luting Yang, Kiran Ranganath, and Shaolei Ren. Why some like it loud: Timing power attacks in multi-tenant data centers using an acoustic side channel. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(1):1–33, 2018.

[19] Haider Adnan Khan, Monjur Alam, Alenka Zajic, and Milos Prvulovic. Detailed tracking of program control flow using analog side-channel signals: a promise for iot malware detection and a threat for many cryptographic implementations. In *Cyber Sensing 2018*, volume 10630, page 1063005. International Society for Optics and Photonics, 2018.

[20] Haider Adnan Khan, Nader Sehatbakhsh, Luong N Nguyen, Milos Prvulovic, and Alenka Zajić. Malware detection in embedded systems using neural network model for electromagnetic side-channel signals. *Journal of Hardware and Systems Security*, 3(4):305–318, 2019.

[21] Haider Adnan Khan, Nader Sehatbakhsh, Luong Ngoc Nguyen, Robert Locke Callan, Arie Yeredor, Milos Prvulovic, and Alenka Zajic. Idea: Intrusion detection through electromagnetic-signal analysis for critical embedded and cyber-physical systems. *IEEE Transactions on Dependable and Secure Computing*, 2019.

[22] Constantinos Kolias, RA Borrelli, Daniel Barbara, and Angelos Stavrou. Malware detection in critical infrastructures using the electromagnetic emissions of plcs. In *Transactions of the American Nuclear Society Winter Meeting 2019*, volume 121, pages 519–522, November 2019.

[23] Constantinos Kolias, Georgios Kambourakis, Angelos Stavrou, and Stefanos Gritzalis. Intrusion detection in 802.11 networks: empirical evaluation of threats and a public dataset. *IEEE Communications Surveys & Tutorials*, 18(1):184–208, 2015.

[24] Andrew Kwong, Wenyuan Xu, and Kevin Fu. Hard drive of hearing: Disks that eavesdrop with a synthesized microphone. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 905–919. IEEE, 2019.

[25] Alireza Nazari, Nader Sehatbakhsh, Monjur Alam, Alenka Zajic, and Milos Prvulovic. Eddie: Em-based detection of deviations in program execution. In *Proceedings of the 44th Annual International Symposium on Computer Architecture*, pages 333–346, 2017.

[26] Jungmin Park, Xiaolin Xu, Yier Jin, Domenic Forte, and Mark Tehranipoor. Power-based side-channel instruction-level disassembler. In *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*, pages 1–6. IEEE, 2018.

[27] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009.

[28] Jeffrey H Reed and Carlos R Aguayo Gonzalez. Enhancing smart grid cyber security using power fingerprinting: Integrity assessment and intrusion detection. In *2012 Future of Instrumentation International Workshop (FIIW) Proceedings*, pages 1–3. IEEE, 2012.

[29] Jeffrey H Reed and Carlos R Aguayo Gonzalez. Using power fingerprinting (pfp) to monitor the integrity and enhance security of computer based systems, February 16 2016. US Patent 9,262,632.

[30] Craig G Rieger. Notional examples and benchmark aspects of a resilient control system. In *2010 3rd International symposium on resilient control systems*, pages 64–71. IEEE, 2010.

[31] Andreas Schropp, Pit Boye, Andy Goldschmidt, Susanne Hönig, Robert Hoppe, Jens Patommel, Christoph Rakete, Dirk Samberg, Sandra Stephan, Sebastian Schöder, et al. Non-destructive and quantitative imaging of a nano-structured microchip by ptychographic hard x-ray scanning microscopy. *Journal of microscopy*, 241(1):9–12, 2011.

[32] Nader Sehatbakhsh, Monjur Alam, Alireza Nazari, Alenka Zajic, and Milos Prvulovic. Syndrome: Spectral analysis for anomaly detection on medical iot and embedded devices. In *2018 IEEE international symposium on hardware oriented security and trust (HOST)*, pages 1–8. IEEE, 2018.

[33] Nader Sehatbakhsh, Robert Callan, Monjur Alam, Milos Prvulovic, and Alenka Zajic. Leveraging electromagnetic emanations for iot security. In *Hardware Demo at IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2017.

[34] Nader Sehatbakhsh, Alireza Nazari, Monjur Alam, Frank Werner, Yuanda Zhu, Alenka Zajic, and Milos Prvulovic. Remote: Robust external malware detection framework by using electromagnetic signals. *IEEE Transactions on Computers*, 2019.

[35] Daehyun Strobel, Florian Bache, David Oswald, Falk Schellenberg, and Christof Paar. Scandalee: a side-channel-based disassembler using local electromagnetic emanations. In *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 139–144. IEEE, 2015.

[36] William C Suski II, Michael A Temple, Michael J Mendenhall, and Robert F Mills. Radio frequency fingerprinting commercial communication devices to enhance electronic security. *International Journal of Electronic Security and Digital Forensics*, 1(3):301–322, 2008.

[37] Chouchang Yang and Alanson P Sample. Em-id: Tag-less identification of electrical devices via electromagnetic emissions. In *2016 IEEE International Conference on RFID (RFID)*, pages 1–8. IEEE, 2016.