Special Session: Emerging (Un-)Reliability Based Security Threats and Mitigations for Embedded Systems

Hussam Amrouch
Karlsruhe Institute of Technology
Haid-und-Neu-Str. 7
Karlsruhe 76131, Germany

Prashanth Krishnamurthy
NYU Tandon School of Engineering
6 MetroTech Center
Brooklyn, NY 11201, USA

Naman Patel
NYU Tandon School of Engineering
6 MetroTech Center
Brooklyn, NY 11201, USA

Jörg Henkel
Karlsruhe Institute of Technology
Haid-und-Neu-Str. 7
Karlsruhe 76131, Germany

Ramesh Karri
NYU Tandon School of Engineering
6 MetroTech Center
Brooklyn, NY 11201, USA

Farshad Khorrami
NYU Tandon School of Engineering
6 MetroTech Center
Brooklyn, NY 11201, USA

ABSTRACT

This paper addresses two reliability-based security threats and mitigations for embedded systems namely, aging and thermal side channels. Device aging can be used as a hardware attack vector by using voltage scaling or specially crafted instruction sequences to violate embedded processor guard bands. Short-term aging effects can be utilized to cause transient degradation of the embedded device without leaving any trace of the attack. (Thermal) side channels can be used as an attack vector and as a defense. Specifically, thermal side channels are an effective and secure way to remotely monitor code execution on an embedded processor and/or to possibly leak information. Although various algorithmic means to detect anomaly are available, machine learning tools are effective for anomaly detection. We will show such utilization of deep learning networks in conjunction with thermal side channels to detect code injection/modification representing anomaly.

KEYWORDS

Embedded Systems; Cyber-Physical Systems; Reliability; Long-Term Aging; Short-Term Aging; Side Channels; Thermal Measurements; Infrared Images.

ACM Reference format:
Hussam Amrouch, Prashanth Krishnamurthy, Naman Patel, Jörg Henkel, Ramesh Karri, and Farshad Khorrami. 2017. Special Session: Emerging (Un-)Reliability Based Security Threats and Mitigations for Embedded Systems. In *Proceedings of ES Week, Seoul, South Korea, Oct. 2017 (ES WEEK'17),* 10 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Modern embedded systems (ES), industrial control systems (ICS), and other cyber-physical systems (CPS) are becoming complex interconnected systems of heterogeneous hardware and software components such as sensors, actuators, controllers, physical systems/ processes that are controlled or monitored, computational nodes, and communication interfaces and protocols. Increasing network connectivity and remote programmability of embedded devices in CPS is increasing the attack surface. At the same time, these capabilities simplify deployment and maintenance of CPS that are geographically spread. One can appreciate the potential widespread and possibly long-lasting impact of attacks on embedded systems especially when an attacker uses knowledge of the process dynamics characteristics to craft process-aware attacks to maximize process impact or to elude detection or both. The complexity and connectivity of embedded devices in CPS necessitates robust cyber-security techniques [20, 21, 42, 45, 53]. There have been several publicized attacks on CPS over the past few years [2, 3, 16, 18, 27, 47, 48, 50, 62, 69]. The number of incidents that the ICS Cyber Emergency Response Team (ICS-CERT) received and responded to in the US has increased from 245 in 2014 to 295 in 2015 [2, 3].

While cyber-security for embedded systems is a broad research area spanning hardware, firmware, and software, both in terms of threats and mitigations, this paper addresses two specific directions. First, the paper addresses the security impact of device aging - both long-term aging and short-term aging. The second direction considered in this paper is thermal side channels for remote monitoring of an embedded system (or possibly creating leaks).

When a device is aged either by running specific instruction sequences or by varying the device supply voltage or the clock, it can create temporal violations of the device guard band. Aging is relevant to embedded systems both as a threat and as a mitigation. On one hand, aging degrades an embedded system impacting the physical processes in the CPS. Malicious aging can be utilized to launch a warranty attack (by a malicious consumer who wishes to wear out a device to misuse the warranty) or planned obsolescence (by a manufacturer who wishes to prematurely degrade a device to force users to replace/upgrade the device). On the other hand, aging can be used as a hardware-level signature of the system to detect counterfeit/ damaged/ compromised embedded devices.

When an embedded device executes code, the underlying physical processes on the device result in analog emissions called side channels. Side channel *modalities* include thermal, power, electromagnetic (EM), magnetic, and acoustic and reflect, in general, information about the characteristics of the code being executed

ES WEEK'17, Seoul, South Korea

DOI: 10.1145/nnnnnnn.nnnnnnn

with different temporal scales and resolutions depending on the side channel modalities. While side channels have been extensively studied in the context of information leakage, i.e., remotely inferring properties about code executing on a device with/without assistance of a malicious code resident on the device, we consider here the possibility of using side channels to remotely monitor the device to continuously verify that the device is operating as intended (e.g., to detect code modifications due to cyber-attacks, etc.). In particular, we consider the thermal side channel of an embedded device and show that thermal images can be used to remotely extract information on device activity patterns.

This paper is organized as follows: Device and circuit aging and its relevance to embedded systems security is addressed in Section 2. Side channels of embedded devices are discussed in Section 3 including, in particular, the thermal side channel. Some concluding remarks are provided in Section 4.

2 AGING IN CIRCUITS

Aging is one of the major concerns in the current and emerging CMOS technology where displacing just few atoms inside transistors due to aging phenomena may degrade the functionality of circuit. Negative and Positive Bias Temperature Instability (NBTI and PBTI) are the most prominent phenomena. In general, BTI increases delays of pMOS and nMOS devices and hence circuits become slower over time [10]. To ensure the correct functioning of a circuit, guard bands (i.e., safety margins) are included in order to compensate for and overcome any such delay increases during the projected lifetime of the device. In advanced technologies, larger guard bands are necessary since only a few defects within a transistor can degrade its functionality.

A guard band is an over-design of the circuit to tolerate degradations to sustain reliable operation during its projected lifetime. A *timing* guard band can be implemented by adding extra time on top of the maximum delay of the circuit as shown in Eq. 1 [12]:

$$t_{clk.period} = t_{delay}(critical\_path) + t_{GB}. \quad (1)$$

To avoid the increase in overall power and hence on-chip power density and temperature, timing guard bands are employed [11]. From a reliability perspective, circuit designers ponder two aging-related questions: *a) how can one accurately estimate guard bands?* and b) *what is the smallest, yet reliable guard band?*

2.1 Long-Term Aging

Classic long-term aging considers BTI-based mechanisms that only increases the threshold voltage of transistors (V_{th}) which leads to a *gradual increase in the circuit's delay*. It occurs when traps are generated at the Si-SiO2 interface when a negative voltage is applied to a PMOS device [14]. (N)BTI increases the magnitude of threshold voltage (V_{th}) of the PMOS transistor under stress and hence degrades the delay through it. At the circuit level, this manifests as circuit timing and functional failures [5, 43, 66].

Figure 1 shows the threshold voltage drift of a PMOS transistor (at an operating temperature of 80°C) that is continuously under stress for 6 months (blue) as well as a transistor that is under stress and recovery every other month (red). In practice, a PMOS transistor experiences stress (when the transistor is on, i.e., when a negative voltage is applied to its gate) and a recovery (when a positive voltage is applied to the gate of the transistor). The impact of BTI on circuit performance has become severe, especially after the introduction of high-k gate dielectrics since the 45 nm technology node [73]. For long-term aging, we use the model from [73].

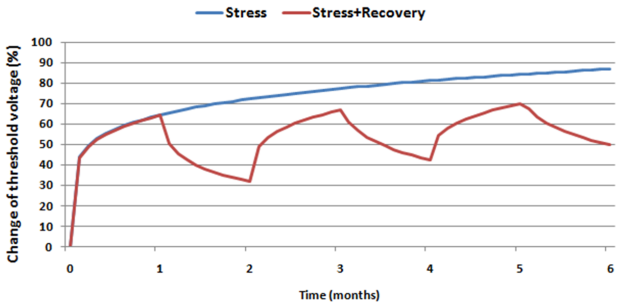

Figure 1: Long-term aging: % change in threshold voltage of a PMOS transistor over time due to BTI [39].

However, other considerations have been noticed recently. The different types of defects (i.e., interface and oxide traps) generated by aging-related phenomena interact with the applied electric fields in the transistors and manifest as different degradations. Besides increasing V_{th}, aging degrades other device parameters including carrier mobility (μ), transconductance (g_m), drain current (I_d), subthreshold slope (SS), and gate-drain capacitance C_{gd} [9][1].

In embedded systems that do not switch frequently between high and low operating voltages (i.e., chips that do not employ Dynamic Voltage and Frequency Scaling (DVFS)), considering BTI-based aging may be sufficient.

2.2 Limitations of Long-Term Aging Models

Considering the impact of aging only on V_{th} underestimates the overall impact of aging on circuit delay. Figure 2 confirms this with our aging analysis for the Berkeley Out-of-Order Machine (BOOM). Considering only V_{th} underestimates the impact of aging on timing guard bands by about 22%. Neglecting impact of aging on carrier mobility underestimates the guard bands by 11% [7][2].

Besides accurately modeling all physical origins of aging, considering the operating conditions of the embedded system is a pre-requisite to improve the accuracy of estimates for the guard bands. Such operating conditions include voltage, temperature, and *duty cycle* (i.e., % of time the transistor is under stress).

The operating conditions stimulate the aging mechanisms [63]. We used different hardware and software approaches to study how workloads running on the system accelerate aging-induced degradation [8]. The workloads determine how a chip ages and how long the guard band can hold before timing violations start to occur. As a result of aging-induced degradations, transistors gradually slow-down over their lifetime.

[1] based on our measurements on devices at the 45nm technology node.
[2] Software Download: The short-term aging models, aging-aware cell libraries, reliability framework, etc. are publicly available at [36].

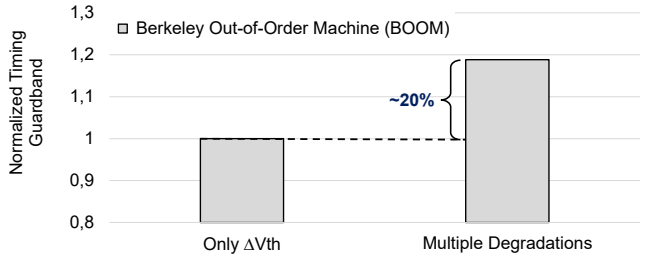

Figure 2: ΔV_{th} is one of many factors to consider when investigating the impact of aging on circuit delay.

Further, to accurately estimate the guard band in advanced technology nodes, Random Telegraph Noise (RTN) needs to be considered as well. This is because while BTI is the dominant aging phenomenon in CMOS at high voltages (e.g., 1.2V) [41], RTN is the dominant aging phenomenon at lower voltages (e.g., 0.7V) [71]. In [71], we reported the first comprehensive model that jointly considers BTI and RTN-based aging allowing designers to accurately assess the impact of long-term and short-term aging-induced degradation across a wide range of voltages ($V_{dd} \in [0.4V : 2.1V]$).

2.3 Short-Term Aging [70]

Besides the *long-term* reliability mechanisms which *gradually* increase the delay in circuits, there has been a paradigm shift in our understanding that shows aging-induces *short-term* reliability degradation as well [70]. The reason behind this is integration of voltage regulators that support *ultra-fast voltage switching* (sub-μs) in Intel Haswell [17] and other chips. While ultra-fast voltage switching reduces the overhead of voltage switching, it considerably accelerates aging. Every time the voltage switches from high V_{dd} to low V_{dd}, the circuit becomes sensitive to aging degradation [70]. While this increase in sensitivity to aging follows the voltage changes instantaneously, the recovery from degradation takes time.

When the circuit operates at high V_{dd}, higher degradations accumulate. When the voltage rapidly switches (i.e., within $\leq 1\mu s$) to a lower V_{dd}, the higher sensitivity at the lower V_{dd} combined with the high degradations (accumulated at the previous high V_{dd}) causes a *temporal violation* of the employed guard band. Such a violation is transient because at the lower V_{dd}, recovery mechanisms kick-in healing the accumulated aging. Figure 3 demonstrates such guard band violations using SPICE simulations that employ a physics-based aging model that accounts for voltage dynamics [31]. Not every high-to-low V_{dd} switching causes a transient error. The two pre-requisites for short-term aging induced transient errors are a) the circuits spends sufficient time at the high V_{dd} to accumulate enough degradations and b) the voltage switches to a sufficiently low V_{dd} to amplify the impact of the degradation.

If the stress is not long enough or voltage switches barely to a lower V_{dd}, transient errors may not occur. Short-term aging does not occur when switching from low V_{dd} to high V_{dd}. This is because degradations at the low V_{dd} are weak and the resiliency to degradation effects is large at high V_{dd}.

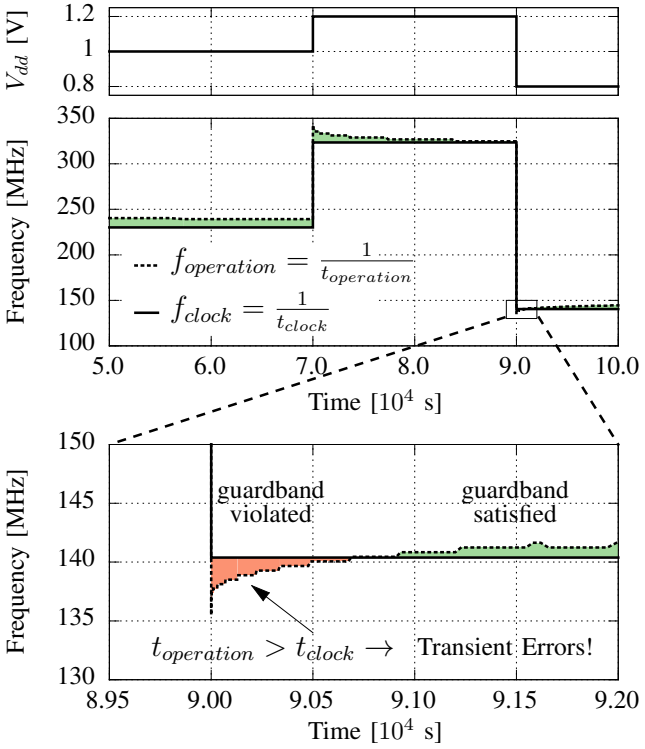

Figure 3: Aging-related degradation at high V_{dd} plus ultra-fast voltage switching cause short-term aging which results in transient errors by temporarily violating the timing guard band [70].

In emerging embedded systems that switch frequently between high and low operating voltages (i.e., chips that employ DVFS) in order to meet performance and power constraints, it is a prerequisite to consider BTI- and RTN-based aging effects.

2.4 Malicious Aging [39]

Deliberately accelerating aging degradation can undermine embedded systems and is an emerging security threat. An adversary may maliciously accelerate the aging of an IC (MAGIC) and shorten its useful life as shown in Fig. 4. MAGIC denies service to the IC user and in turn may cause catastrophic failure of the system [39]. MAGIC exploits the fact that the circuit delay is input dependent and that it exhibits its worst-case delay for specific input patterns [77]. MAGIC attack identifies such input patterns and constructs a malicious program that generates such patterns. Executing this malicious program on embedded devices such as mobile phones, tablets, and PCs accelerates IC failure and can cause the chip to fail sooner than expected, i.e., shortens the chip's normal lifetime. We used the long-term aging models [73] to demonstrate MAGIC. One can envision at least two types of MAGIC attacks [39].

Warranty Attack: A consumer C purchases a device manufactured by company X. The warranty period for this device is W months. Consumer C uses the device for a while but when the device is still under warranty, a physical damage occurs (e.g., scratch on the LCD). C wants to get a new device but the warranty does

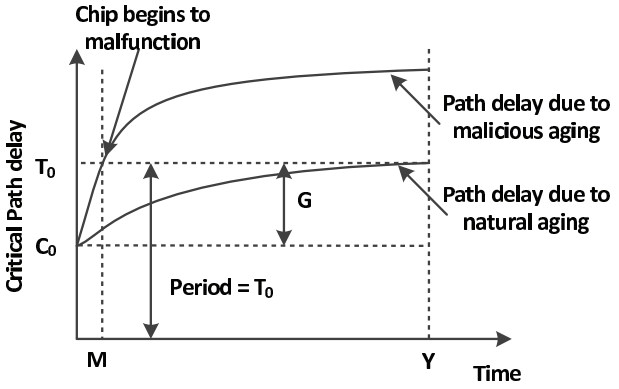

Figure 4: MAGIC violates guard bands by aging devices [39].

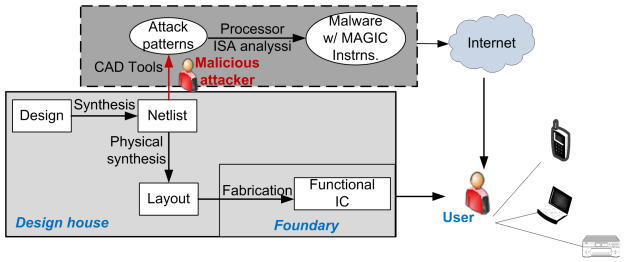

Figure 5: The MAGIC flow is shown in the dotted box. The attacker identifies the critical path, the input patterns which maximally stress the gates in the path, instructions that generate the input patterns and builds a MAGIC program. When the MAGIC program is run, processor ages and fails prematurely [39].

not cover physical damage. C downloads the MAGIC program and the OS from forums such as Cyanogenmod [1], executes the MAGIC program to intentionally wear-out the device and returns the device to X to get a new one. In the warranty attack, the user attempts to brick the device. The warranty attack may not inflict a considerable financial loss to the victim manufacturer, but may hurt its reputation. In the warranty attack, MAGIC is created by an *expert-attacker* and launched by a *non-expert malicious user*. The expert-attacker is usually an insider with access to the processor netlist and can use VLSI CAD tools. The expert-attacker creates a MAGIC program and distributes it to users to bring disrepute to the manufacturer by wearing out even a few devices. On the other hand, the goal of the novice user[3] is to damage his device before the warranty period expires and exchange it for a new one.

Planned Obsolescence: A malicious manufacturer M slows down the previously sold devices in order to nudge (force) its customers to buy a recently released device. M sends a patch to its customers before releasing a new device. Installing such patch slows the older devices forcing the users to buy the new device [55, 59, 65, 67, 78]. Planned obsolescence financially benefits the manufacturer by nudging its customers to upgrade to the latest device. The manufacturing company wears out the device. Planned obsolescence makes the device stop working properly but malicious aging is hidden from the user, i.e., pretending that the device has aged normally and hiding that aging is due to MAGIC.

Figure 5 shows the design flow (solid box) and the MAGIC attack launched within this flow (dotted box). The processor is synthesized and netlist and layout are generated. The layout is sent for fabrication. After IC testing, fault-free ICs are shipped. The warranty attack can be launched as follows:

Step 1: A *malicious insider* in the design house obtains the processor netlist.
Step 2: The attacker identifies the critical path in the processor and creates input patterns that place this path under BTI stress. The attacker analyzes the processor Instruction Set Architecture (ISA) and crafts instructions to create a MAGIC program to generate the above patterns.

[3]We assume the non-expert user has a hacked Operating System (OS) installed and has root access to his device.

Step 3: The attacker uploads the MAGIC program to a website and the non-experts download and execute it on their processors.

MAGIC [39] was demonstrated on the OpenSPARC T1 processor [56]. The degradation was evaluated using the long-term BTI-based aging model from [73]. The execute stage (E-stage) in the processor was maliciously aged. When the MAGIC program was executed, the performance of the E- stage degraded by 10.92%, 13.25%, and 16.8% after one, two, and six months, respectively, bypassing guard band and other protections, causing the processor to fail. MAGIC patterns can be generated for any pipeline stage. For Open SPARC, we chose E-stage as it has the critical paths.

Post-manufacture critical path may differ from design-time critical path due to process variations. After manufacturing, when the MAGIC program is executed, the design-time critical path will age and become longer than the manufacture-time critical path. We observed that the top 10 longest paths in the E-stage of the OpenSPARC were within 2% of each other. The change in threshold voltage, and in turn the critical path delay, is affected by the temperature. Thus, temperature is another knob for the attacker.

2.5 What are the Security Implications of Short-term Aging ?

Short-term aging manifests as a temporal violation of the guard band resulting in transient errors (i.e., timing violations by increasing the path delay). We have and are investigating important questions along the following directions: a) short-term aging attacks be launched in a controlled manner to transiently undermine the security of on-chip systems at specific instances and for specific durations, b) short-term aging attacks to accelerate warranty and obsolescence attacks that were demonstrated using long-term aging, c) short-term aging as a standalone attack vector; if the temporal violation of the guard band is large enough, the on-chip system becomes unstable because of the unsustainable clock frequency leading to errors/crashes in the software due to the induced transient timing errors, and d) short-term aging as a trigger for a Trojan that is inserted in the chip (e.g., by a rogue in the foundry) and lays dormant until short-term aging effects exceed a threshold and then activates its malicious behavior.

3 SIDE CHANNELS [57]

Various analog side channel modalities including thermal, power, electromagnetic (EM), magnetic, and acoustic are relevant to embedded devices and have been heavily studied in the literature [4, 13, 15, 19, 23, 25, 26, 28, 29, 32–35, 37, 38, 40, 51, 52, 54, 60, 61, 64, 68, 72]. Various efforts have addressed side channels such as electromagnetic (EM) [22, 26, 28, 29, 32, 54, 72], acoustic [22, 30, 37, 49], thermal [35, 38], magnetic [15], and power [24, 61]. Side channels leak information from air-gapped devices by running malicious code on the device so as to create signatures in the side channels which when monitored can yield retrieve sensitive information. A computer infected with malicious code that excites specific radio frequency signal patterns using the graphics card can leak information to a mobile phone with a radio FM receiver[34].

The majority of the prior works have addressed side channels in the context of emission patterns from digital devices and information leakage through these analog side channels. Information leakage can be viewed as a special case of monitoring wherein specifically crafted code on the device generates sequences of activity patterns (of the processor load, clock, memory bus, peripherals, etc.) that can be decoded by an air-gapped receiver to extract messages (e.g., a sequence of bits) sent by the code resident on the device [44]. Multiple side channel modalities are applicable for remote monitoring of the device state (e.g., whether the code execution on the device matches nominal expected patterns or exhibits anomalies).

EM signals enable high-bandwidth monitoring of embedded processors (up to several GHz) and code execution patterns within the device by observing the signal frequency content and temporal patterns. The EM signals generated by the components of the embedded processor – including the CPU, GPU, memory, clocks, data storage components, voltage regulators, and input/output modules and associated analog/digital circuitry and wiring – vary depending on their usage, computation, and communication patterns (e.g., use of system memory bus to exchange data between CPU and memory). Actuators such as motors also generate distinctive EM signal patterns during their operation. From high-bandwidth EM signal measurements, machine learning techniques can provide a high-level of detail of the device activity. EM signals generated by components and operations in an embedded processor can be differentiated by their frequency content and temporal patterns yielding discernible signatures of events during code execution. To capture noisy EM signals over a wide range of frequencies and under cluttered conditions, combinations of multiple receiver/antenna pairs (e.g., helical, microstrip, Vivaldi) and antenna arrangements can be used with optimized configurations for specific types of devices. Multi-antenna geometric arrangements and polarizations can enable robust data acquisition in noisy environments where multiple devices and other EM sources may be present.

Power measurements provide aggregate readings reflecting the activity of the embedded processor including CPU and GPU usage patterns. Power measurements from CPS peripherals (such as sensors and actuators) yield information on device activity. Over a longer time scale, thermal measurements provide readings corresponding to CPS device activity. Thermal signatures can differentiate among components in a system (e.g., between two processors in a multi-processor system). Magnetic signals provide a low-bandwidth device signature that can be used in conjunction with EM signal measurements to detect hardware-level modifications (e.g., unauthorized hardware changes or tampering), especially in close proximity to the target device. Actuators (e.g., motors) generate distinctive acoustic signals correlated to their operational states (e.g., RPM of a motor). Various physical processes in a microcontroller generate an acoustic signal (e.g., vibrations of electronic components in the power regulation circuitry), albeit outside the human auditory range.

These analog side channels provide somewhat overlapping, but complementary, sources of information about the state of the monitored device. When multiple side channels are monitored, these side channel signals can be sampled at different sampling rates depending on the sensing modality and then time-synchronized. Fusion of multiple information streams enables robust, remote-awareness of the state of the monitored device including the device characteristics, code modifications, and in general, real-time analysis of the code execution state and control flow within the device.

These analog side channels can be complemented by on-processor digital side channels that measure special-purpose registers (e.g., Hardware Performance Counters or HPCs). Real-time, on-device monitoring using digital side channels have been studied [74–76]. These methods can be used for signature-based detection of malware and detection of device code-specific HPC pattern deviations. HPCs are special-purpose registers built into modern processors (e.g., Intel x86, ARM, MIPS, and PowerPC). The NumChecker [74, 75] and ConFirm [76] demonstrates that HPCs can detect malicious firmware and software modifications [74–76]. For example, HPC-based monitoring can detect kernel rootkits by analyzing the system call behavior of unmodified and modified code blocks [74, 75].

Unlike on-device monitoring, the proposed approach uses remote monitoring of analog emissions across an air-gap. While prior work primarily used side channels as an information leakage mechanism and considers security vulnerabilities of analog emissions from an air-gapped device, the proposed approach *uses* these emissions for real-time monitoring. In particular, we consider the thermal side channel using an infrared camera and show that the code execution on the device can be remotely monitored using sequences of thermal images.

3.1 Thermal Side Channels

In this paper, the thermal side channel is considered as a representative remote sensing modality. The high-resolution thermal imaging testbed shown in Figure 6 is used for remote thermal monitoring of a multi-core Intel processor. In order to keep the processor operating without packaging and heat sinks, a stable and controlled source of cooling is provided by a thermoelectric Peltier element that dissipates the heat generated from the chip from the back side. This setup enables the thermal camera to capture the IR radiations

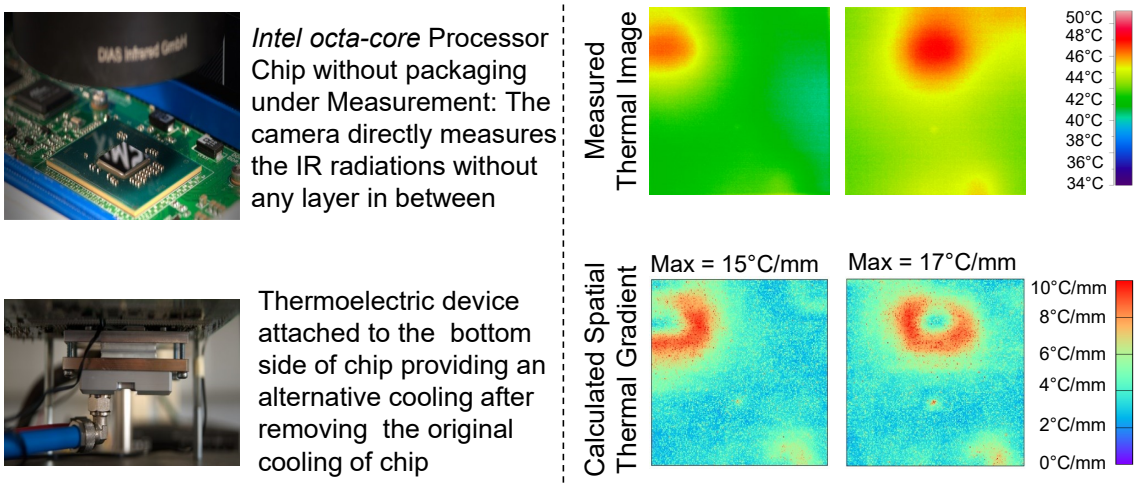

Figure 6: Thermal monitoring setup and examples of thermal images of an Intel 8-core chip. [6]

emitted from the chip directly without any intervening layers interfering with the radiations. Furthermore, the cooling mechanism from the back side can be calibrated by changing the power to the Peltier device to mimic the behavior of the original cooling of the chip using heat sinks, packaging, etc.

The thermal side channel monitoring approach described below in this section considers CPS applications wherein embedded devices run periodic computations. For instance, a CPS device implementing a control algorithm (e.g., to control [46] motors and other electromechanical systems) typically displays a relatively well-structured temporal behavior as a repeated sequence of sensor reading, sensor data processing, control algorithm computation, and actuator writing steps as shown in Figure 7. Other CPS-relevant applications (e.g., aggregating data from sensors, fusing data from sensors to provide a situational awareness to a human operator, etc.) have similar periodic code structures. The periodicity of CPS code results in well-defined periodic characteristics of (thermal) side channel emissions from the device. Hence, by observing the characteristics and the temporal patterns of side channel emissions, deviations in the embedded device behavior during code execution can be detected.

The temporal patterns in the thermal imagery generated due to the code running on the processor can be used to identify the changes in code using machine learning. A simple approach will extract low-dimensional features such as temperature variations in each processor core, maximum, minimum or average temperatures in the region of each processor core, frequency-domain features like periodicity of the measured signal from thermal images. The time-series of such low-dimensional feature data can be used to detect changes from the "nominal" behavior using a one-class Support Vector Machine classifier. Furthermore, an end-to-end machine learning approach can be used to automatically learn subtle spatial and temporal patterns of thermal images obviating the need for manual feature extraction. One can automatically and implicitly learn feature representations optimized for the device computational activity estimation and anomaly detection.

Similar to typical embedded controller code in CPS devices, a code comprising of periodic iterations of a time period of relatively high computational activity (activity time) followed by a fixed time period of low activity (sleep time) is considered. An instantiation of this code structure is characterized by a loop time period T and an activity time period Δ. The loop time period is the sampling time or iteration time in an embedded controller code in a CPS device while the activity time period Δ is the amount of time required for the computations performed in an iteration of the loop. The sampling time T is typically a fixed quantity that is chosen depending on the task being performed by the CPS device while the activity time period Δ depends on the computations being done within each sampling period. We consider a fixed period T and a variable activity time $\Delta \in (0, T)$ and pose the machine learning problem as estimation of Δ, given a time sequence of thermal heat maps over a sliding window of time. From the estimated time-series of activity times Δ, an anomaly detection algorithm probabilistically determines whether the estimated activity times correspond to expected values for the device based on the observation that a cyber-attack that removes, adds, or modifies code in a CPS device will result in a modification of the activity time. The overall machine learning methodology for computational activity time estimation and thereby anomaly detection is shown in Figure 8.

For simplicity and to focus on machine learning based activity time estimation, we consider a scenario in which the activity time Δ is a nominally fixed during normal operation, Δ could vary during normal operation depending on, for example, input data to the CPS device. The methodology can extend to such a case by characterizing ranges of temporal patterns of Δ instead of a single fixed nominal Δ and basing anomaly detection on evaluation of the deviation between the machine learning based estimated activity times and the expected ranges or temporal patterns of activity time.

In our end-to-end machine learning based framework, a sequence of thermal heat maps over a sliding window of time (defined here to be 0.5 s, corresponding to 25 consecutive images since the thermal imager provides 50 frames per second) of the microprocessor over a time window are used as the input to a convolution neural

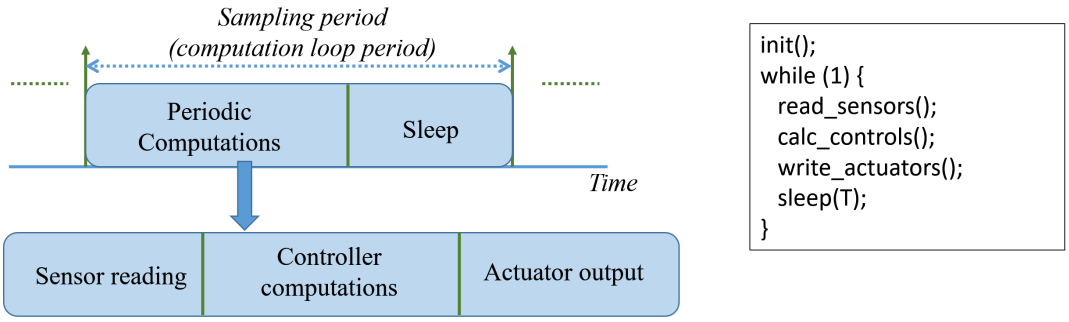

Figure 7: The periodic code structure in a CPS device comprises of periodically repeating computations interspersed with sleep times, e.g., a loop of sensor reading, control algorithm calculations, and actuator writing with a fixed sampling time (a code snippet is shown on the right). [57]

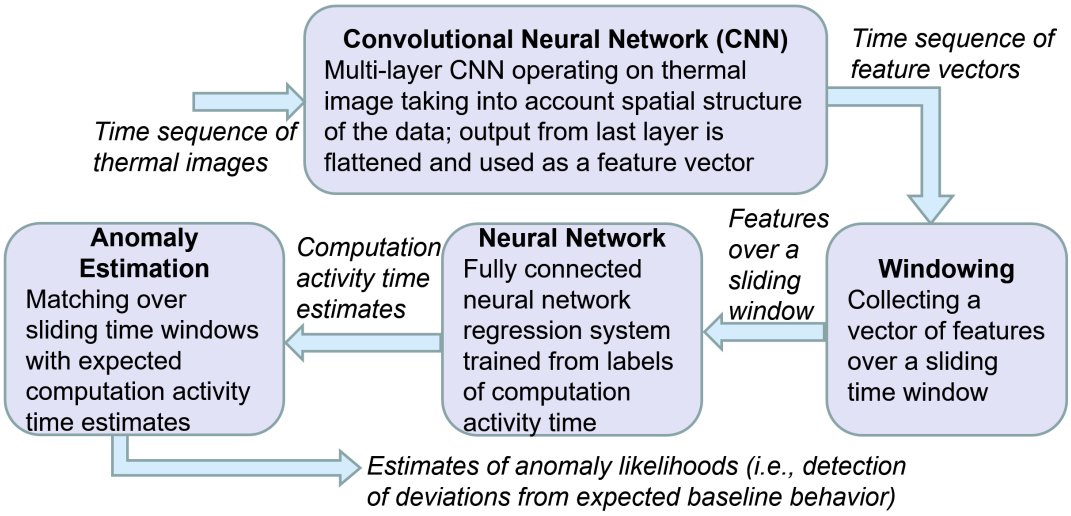

Figure 8: Machine learning based methodology to estimate the CPS device computational activity time and for anomaly detection from a sequence of thermal heat maps. [57]

network (introduced in [57]) to predict the activity time. The use of a high-speed thermal imager will enable more precise activity time prediction due to finer temporal granularity.

The proposed neural network architecture has five convolutional blocks, each comprising of a spatial convolution layer, a rectified linear unit (ReLU) layer and a max-pooling layer. The number of convolutional kernels in each block are 16, 32, 32, 64 and 64. The size of each convolutional kernel in all the blocks is 3x3 with a stride of 1 and the size of each max-pooling kernel is 2x2 with a stride of 2. The weights of all the convolution blocks are shared over all the images in the specified time window. The output of the last convolutional block for all the images in the specified time window are flattened and combined. The concatenated output is passed through three fully connected neural network with a ReLU non-linearity to output feature vectors of size 1024, 128 and 32 respectively. The feature vector of size 32 is passed through a fully connected neural network to predict the activity time. The weights of the network were optimized using Adaptive moment estimation optimizer with a Huber loss function.

In order to generate the thermal heat map dataset for our end-to-end learning system, code for various configurations of T and Δ was implemented using the algorithmic structure shown in Figure 7. The code performs floating-point calculations over the specified time periods. For each value of Δ, sliding time windows are defined for the collected thermal data set with a stride of 5 frames, i.e., time windows comprised of frames 1 to 25, frames 6 to 30, etc. The set of normalized gray-scale thermal images in a time window is input to the end-to-end learning system which predicts the corresponding value of Δ for that data set. The overall thermal dataset was split into training and validation dataset with a ratio of 75:25. Extraneous off-die parts of the overall acquired thermal image are cropped out of the overall heat map to obtain a heat map of 270x270 pixels.

The accuracy of estimation of Δ was evaluated on the testing data set. The estimation of Δ for testing data sets with actual Δ value of 0.1 s is shown in Figure 9. In this figure, the time series of estimates of Δ for sliding time windows of thermal image sequences (with a stride of 5 frames as discussed above; hence, a new estimate of Δ after every 0.1 s since the thermal imager provides 50 frames

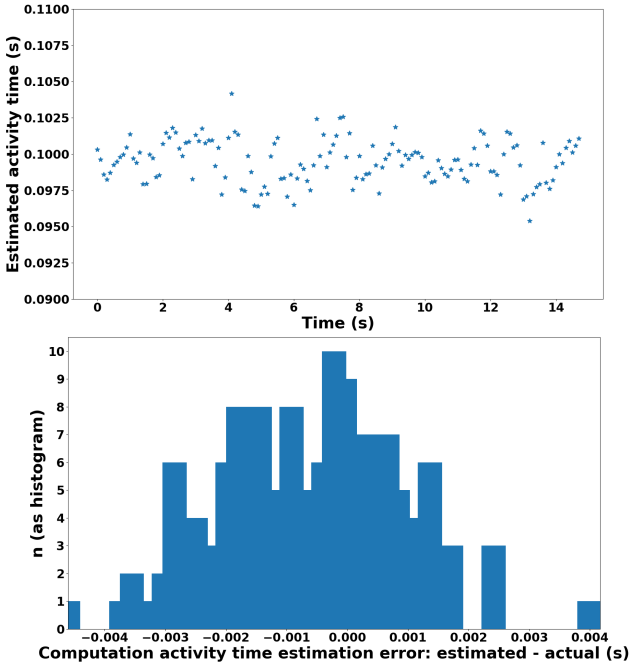

Figure 9: Estimates of computation time from sliding time windows of thermal heat maps collected with computational activity time $\Delta = 0.1$ s. The top figure shows the time series of estimates of Δ from successive sliding time windows of heat maps and the bottom figure shows the histogram of the errors (predicted - actual) in the estimated values of Δ. The histogram shows that the estimated values of Δ are centered around the correct value of 0.1 s with a Gaussian distribution of errors around this correct value. [57]

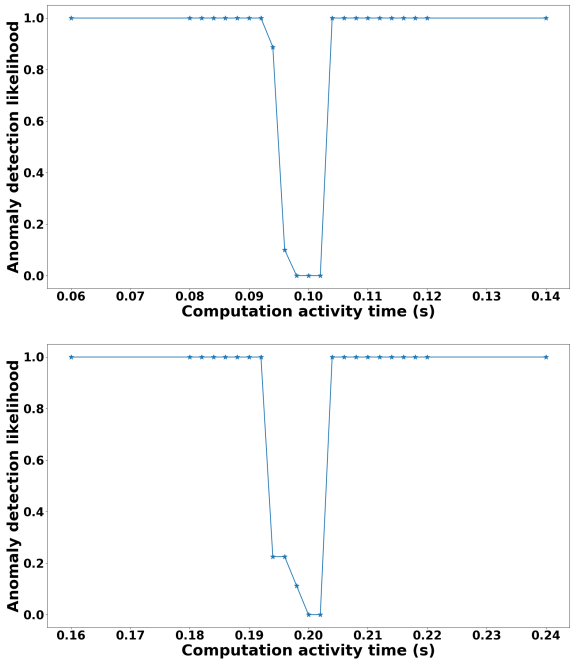

Figure 10: Probability of classifying a time sequence of heat maps as anomalous from a sliding window analysis of machine learning based estimates of Δ) for a range of actual Δ and expected Δ of 0.1 s (top figure) and 0.2 s (bottom figure). An increase in the computation time by as little as 4 ms is detected with 100% accuracy. This is noteworthy since the thermal imager provides 20 ms temporal granularity (since it only supports 50 frames per second). [57]

per second) and histogram of the estimation error (estimated Δ - actual Δ) are shown. For the collected data sets with Δ of 0.1 s and 0.2 s, the means of the absolute values of the estimation errors are 1.26 ms and 1.86 ms, respectively, and the standard deviations are 1.48 ms and 2.49 ms, respectively.

Based on the estimation of Δ from sequences of thermal images, anomalies (i.e., changes to the running code) are detected by probabilistically matching sequences of estimated Δ values over sliding time windows of thermal images against the nominal Δ, or more generally, expected ranges or temporal variation patterns of Δ. In the simplest case wherein the nominal Δ is a constant Δ_{nom}, an anomaly is detected if in a time sequence Δ_t of estimated Δ values over a time window (set here to 2 s, i.e., 20 consecutive estimates of Δ_t), a sufficient percentage (set to 90%) of Δ_t are different from Δ_{nom} by more than a threshold (specified here as 0.0015 s) and if the mean of the estimates Δ_t over the considered time window is different from Δ_{nom} by more than a threshold (also 0.0015 s).

The results of our anomaly detection algorithm on data sets collected with Δ settings in the ranges around 0.1 s and 0.2 s are shown in Figure 10. The anomaly detection likelihoods correspond to the percentages of time windows in these data sets that the anomaly detection algorithm declared as anomalous when comparing against the nominal values of 0.1 s and 0.2 s, respectively. There are no false positives and variations in Δ by around 4 ms increase or 8 ms decrease are detected as anomalous with 100% accuracy (i.e., without false negatives). It is noteworthy that both the estimation of Δ and the detection of variations of Δ provide temporal granularities superior to the 0.02 s (i.e., 50 frames per second) sampling period of the thermal imager. This indicates that the machine learning system can use the fine-grain variations in temperature and the spatial and temporal patterns to accurately estimate the computation activity.

Changes in periodic code structures can be robustly detected using the high-resolution thermal imaging data. While we use an infrared thermal camera in our experiments, the algorithmic approaches can operate on scalar temperature measurement streams as long as they are of sufficient thermal signal and temporal resolution. The technique can be effectively used with on-processor temperature measurements if the processor-integrated temperature sensors provide better resolution than the 1 degree Celsius typically provided by the on-chip, integrated sensors (on-chip monitoring using typical integrated temperature sensors and processor fan was considered in [58]). To find the regions of the thermal image (corresponding to discrete locations of a small set of on-chip, thermal sensors) that are of most utility for estimating processor activity, the machine learning system can be modified to include a sparsity-inducing component to automatically learn salient parts of the image. A masking matrix approach was utilized in [57] for this purpose. Salient parts of the thermal image were learned in

terms of a masking matrix whose weights are learned through back-propagation in an end-to-end manner along with the weights of the proposed network architecture. The sum of absolute values of the weights was utilized as a sparsity-inducing regularization component in the machine learning loss function. It was seen in [57] that when retrained with this modified loss function, a small fraction of the overall image was sufficient to estimate Δ without any appreciable loss of accuracy. Thus, it can be inferred that integrating high-resolution high-sampling-rate temperature sensors into processors at strategic locations (which may physically correspond to power circuitry, cores, caches, etc.) can enable accurate estimation of computation times and robust anomaly detection.

4 CONCLUSION

While this paper addressed aging of and analog side channels in embedded systems, there are connections/ synergies between these two directions that are relevant to embedded systems security. Side channels can be used to monitor for aging effects, both in the context of detecting aging-based attacks and also in the context of facilitating device integrity testing using short-term aging as a signature for the monitored embedded device. Short-term aging can be utilized to cause patterns of transient changes on the device to leak information via side channels. On a device with a pre-loaded malicious code, short-term aging can be used as a trigger to create transient effects that wake up a Trojan on the device that then executes the malicious code to leak information via side channels from the device or the CPS physical process. On the flip side, side channel monitoring can be used to detect execution of such malicious code. Aging effects (especially short-term aging) naturally disappear following an attack due to the "recovery" intrinsic to the aging mechanism. Subsequent forensic analysis to retrace the attack is difficult if not impossible. This is unique to aging and unlike other attack vectors that may leave a forensic trail.

5 ACKNOWLEDGEMENTS

The work is supported in part by a US-German travel supplement to NSF grant 1319841, ONR grants N00014-15-12182 and N-00014-17-12006, Boeing, and the German Research Foundation (DFG) as part of priority program "Dependable Embedded Systems" (SPP 1500 – http://spp1500.itec.kit.edu/).

REFERENCES

[1] 2014. Cyanogenmod forum. http://forum.cyanogenmod.com/ (last accessed 10 Feb., 2014). (2014).

[2] 2014. ICS-CERT year in review – 2014. [Online]: https://ics-cert.us-cert.gov/sites/default/files/Annual_Reports/Year_in_Review_FY2014_Final.pdf. (2014).

[3] 2015. NCCIC/ICS-CERT Year in Review – 2015. [Online]: https://ics-cert.us-cert.gov/sites/default/files/Annual_Reports/Year_in_Review_FY2015_Final_S508C.pdf. (2015).

[4] D. Agrawal, B. Selcuk, K.Deniz, P. Rohatgi, and B. Sunar. 2007. Trojan detection using IC fingerprinting. In *IEEE Symposium on Security and Privacy*. 296–310.

[5] M. A. Alam, K. Haldun, V. Dhanoop, and M. Souvik. 2007. A comprehensive model for PMOS NBTI degradation: Recent progress. *Microelectronics Reliability* 47, 6 (June 2007), 853–862. DOI: http://dx.doi.org/10.1109/MDT.2003.1232254

[6] H. Amrouch and J. Henkel. 2015. Lucid infrared thermography of thermally-constrained processors. In *IEEE/ACM Symposium on Low Power Electronics and Design (ISLPED)*. 347–352.

[7] H. Amrouch, B. Khaleghi, A. Gerstlauer, and J. Henkel. 2016. Reliability-aware Design to Suppress Aging. In *IEEE/ACM Design Automation Conference*. 1–6.

[8] H. Amrouch, J. Martin-Martinez, V. van Santen, M. Moras, R. Rodriguez, M. Nafria, and J. Henkel. 2015. Connecting the physical and application level towards grasping aging effects. In *IEEE Reliability Physics Symposium (IRPS)*. 3.D.1.1–3.D.1.8.

[9] H. Amrouch, S. Mishra, V. van Santen, S. Mahapatra, and J. Henkel. 2017. Impact of BTI on dynamic and static power: From the physical to circuit level. In *IEEE Reliability Physics Symposium (IRPS)*. CR–3.1–CR–3.6.

[10] H. Amrouch, V. van Santen, T. Ebi, V. Wenzel, and J. Henkel. 2014. Towards Interdependencies of Aging Mechanisms. In *IEEE/ACM Conference on Computer-Aided Design*. 478–485.

[11] H. Amrouch, V. M. van Santen, and J. Henkel. 2017. Interdependencies of Degradation Effects and Their Impact on Computing. *IEEE Design & Test* 34, 3 (June 2017), 59–67.

[12] S. Arasu, M. Nourani, J. M. Carulli, and V. K. Reddy. 2016. Controlling Aging in Timing-Critical Paths. *IEEE Design & Test* 33, 4 (Aug 2016), 82–91.

[13] J. Balasch, B. Gierlichs, and I. Verbauwhede. 2015. Electromagnetic circuit fingerprints for Hardware Trojan detection. In *IEEE International Symposium on Electromagnetic Compatibility*. 246–251.

[14] S. Bhardwaj, W. Wang, R. Vattikonda, Y. Cao, and S. Vrudhula. 2006. Predictive modeling of the NBTI effect for reliable design. In *IEEE Custom Integrated Circuits Conference*. 189–192. DOI: http://dx.doi.org/10.1109/CICC.2006.320885

[15] S. Biedermann, S. Katzenbeisser, and J. Szefer. 2015. Hard Drive Side-Channel Attacks using Smartphone Magnetic Field Sensors. In *International Conference in Financial Cryptography and Data Security*. Springer, 489–496.

[16] C. Blask. 2011. ICS Cybersecurity: Water, water everywhere. [Online]: http://www.infosecisland.com/blogview/18281-ICS-Cybersecurity-Water-Water-Everywhere.html. (Nov 2011).

[17] E. Burton, G. Schrom, F. Paillet, J. Douglas, W. J. Lambert, K. Radhakrishnan, and M. Hill. 2014. FIVRfi!?Fully integrated voltage regulators on 4th generation Intel® Coreffl SoCs. In *IEEE Applied Power Electronics Conference and Exposition (APEC)*. 432–439.

[18] E. Byres and J. Lowe. 2004. The myths and facts behind cyber security risks for industrial control systems. In *The VDE Kongress*, Vol. 116. 213–218.

[19] R. Callan, A. Zaji, and M. Prvulovic. 2014. A practical methodology for measuring the side-channel signal available to the attacker for instruction-level events. In *IEEE/ACM International Symposium on Microarchitecture*. 242–254.

[20] A. Cárdenas, S. Amin, and S. Sastry. 2008. Research Challenges for the Security of Control Systems. In *3rd USENIX workshop on Hot Topics in Security*.

[21] A. Cárdenas, S. Amin, B. Sinopoli, A Giani, A. Perrig, and S Sastry. 2009. Challenges for securing cyber physical systems. In *Workshop on future directions in cyber-physical systems security*.

[22] B. Carrara and C. Adams. 2014. On acoustic covert channels between air-gapped systems. In *International Symposium of Foundations and Practice of Security*. Springer, 3–16.

[23] S. Chari, J. R. Rao, and P. Rohatgi. 2002. Template attacks. In *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 13–28.

[24] S. S. Clark, B. Ransford, A. Rahmati, S. Guineau, J. Sorber, K. Fu, and W. Xu. 2013. WattsUpDoc: power side channels to non-intrusively discover un-targeted malware on embedded medical devices. In *USENIX Conference on Safety, Security, Privacy and Interoperability of Health Information Technologies*. 9–9.

[25] A. Dakshi, R. Josyula, R. Pankaj, and S. Kai. 2005. Templates as master keys. In *International Workshop on Cryptographic Hardware and Embedded Systems (CHES)*. Springer, 15–29.

[26] F. Debeer, M. Witteman, B. Gedrojc, and Yijun S. Riscure. 2011. Practical Electro-Magnetic Analysis. In *Non-invasive Attack Testing Workshop NIAT, Nara: Todai-ji Cultural Center (Technical Programs)*.

[27] N. Falliere, L. Murchu, and E. Chien. 2011. W32. Stuxnet dossier. *White paper, Symantec Corp., Security Response* 5 (2011).

[28] P. Fouque, G. Leurent, D. Réal, and Fr. Valette. 2009. Practical electromagnetic template attack on HMAC. In *International Workshop on Cryptographic Hardware and Embedded Systems (CHES)*. Springer, 66–80.

[29] D. Genkin, L. Pachmanov, I. Pipman, and E. Tromer. 2015. Stealing keys from PCs using a radio: Cheap electromagnetic attacks on windowed exponentiation. In *International Workshop on Cryptographic Hardware and Embedded Systems*. Vol. 9293. Springer, 207–228.

[30] D. Genkin, A. Shamir, and E. Tromer. 2014. RSA key extraction via low-bandwidth acoustic cryptanalysis. In *Advances in Cryptology (CRYPTO)*. Springer, 444–461.

[31] N. Goel, T. Naphade, and S. Mahapatra. 2015. Combined trap generation and transient trap occupancy model for time evolution of NBTI during DC multi-cycle and AC stress. In *IEEE Reliability Physics Symposium (IRPS)*. 4A–3.

[32] G. Goller and G. Sigl. 2015. Side Channel Attacks on Smartphones and Embedded Devices Using Standard Radio Equipment. In *International Workshop on Constructive Side-Channel Analysis and Secure Design*. Vol. 9064. Springer, 255–270.

[33] M. Guri, A. Kachlon, O. Hasson, G. Kedma, Y. Mirsky, and Y. Elovici. 2015. GSMem: data exfiltration from air-gapped computers over GSM frequencies. In *USENIX Security Symposium*. 849–864.

[34] M. Guri, G. Kedma, A. Kachlon, and Y. Elovici. 2014. AirHopper: Bridging the air-gap between isolated networks and mobile phones using radio frequencies. In *IEEE International Conference on Malicious and Unwanted Software*. 58–67.

[35] M. Guri, M. Monitz, Y. Mirski, and Y. Elovici. 2015. BitWhisper: Covert Signaling Channel between Air-Gapped Computers using Thermal Manipulations. In *IEEE Conference on Computer Security Foundations*. 276–289.

[36] H.Amrouch and J. Henkel. 2016. KIT short-term aging models, tools and degradation-aware cell libraries. http://ces.itec.kit.edu/dependable-hardware.php. (2016).

[37] M. Hanspach and M. Goetz. 2014. On covert acoustical mesh networks in air. *arXiv preprint arXiv:1406.1213* (2014).

[38] M. Hutter and J. Schmidt. 2013. The temperature side channel and heating fault attacks. In *International Conference on Smart Card Research and Advanced Applications*. Springer, 219–235.

[39] N. Karimi, A. K. Kanuparthi, X. Wang, O. Sinanoglu, and R. Karri. 2015. MAGIC: Malicious Aging in Circuits/Cores. *ACM Transaction on Architecture Code Optimization* 12, 1 (Apr. 2015), 5:1–5:25.

[40] T. Kasper, D. Oswald, and C. Paar. 2011. Side-channel analysis of cryptographic RFIDs with analog demodulation. In *International Workshop on RFID. Security and Privacy*. Vol. 7055. Springer, 61–77.

[41] J. Keane, X Wang, D. Persaud, and C. H. Kim. 2010. An All-In-One Silicon Odometer for Separately Monitoring HCI, BTI, and TDDB. *IEEE Journal of Solid States Circuits* 45, 4 (Apr 2010), 817 – 829. DOI: http://dx.doi.org/10.1109/JSSC.2010.2040125

[42] A. Keliris, H. Salehghaffari, B. Cairl, P. Krishnamurthy, M. Maniatakos, and F. Khorrami. 2016. Machine learning-based defense against process-aware attacks on industrial control systems. In *IEEE International Test Conference (ITC)*. 1–10.

[43] S. Khan, Nor Z. Haron, S. Hamdioui, and F. Catthoor. 2011. NBTI monitoring and design for reliability in nanoscale circuits. In *IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems*. 68–76.

[44] F. Khorrami, R. Karri, and P. Krishnamurthy. 2017. Instrumenting Code for Embedded Controlled Remote Autonomous Monitoring. (Jan 2017).

[45] F. Khorrami, P. Krishnamurthy, and R. Karri. 2016. Cybersecurity for control system: A process aware perspective. *IEEE Design & Test* 33, 5 (Oct 2016), 75–83.

[46] F. Khorrami, P. Krishnamurthy, and H. Melkote. 2003. *Modeling and Adaptive Nonlinear Control of Electric Motors*. Springer Verlag.

[47] E. Kovacs. 2014. Cyberattack on German Steel Plant Caused Significant Damage. [Online]: http://www.securityweek.com/cyberattack-german-steel-plant-causes-significant-damage-report. (Dec 2014).

[48] D. Kravets. 2009. Feds: Hacker Disabled Offshore Oil Platforms' Leak-Detection System. [Online]: http://www.wired.com/2009/03/feds-hacker-dis/. (Mar 2009).

[49] P. Krishnamurthy, F. Khorrami, R. Karri, D. Paul-Pena, and H. Salehghaffari. 2017. Process-aware side-channel information leakage from physical instrumentation/devices in cyber-physical systems. (2017). submitted for journal publication.

[50] D. Kushner. 2013. The Real Story of Stuxnet. [Online]: http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet. (Feb. 2013).

[51] H. Li, A. T. Markettos, and S. Moore. 2005. Security evaluation against electromagnetic analysis at design time. In *International Workshop on Cryptographic Hardware and Embedded Systems (CHES)*. Springer, 280–292.

[52] V. Lomné, E. Prouff, M. Rivain, T. Roche, and A. Thillard. 2014. How to estimate the success rate of higher-order side-channel attacks. In *International Workshop on Cryptographic Hardware and Embedded Systems (CHES)*. Springer, 35–54.

[53] S. McLaughlin, C. Konstantinou, X. Wang, L. Davi, A. Sadeghi, M. Maniatakos, and R. Karri. 2016. The Cybersecurity Landscape in Industrial Control Systems. *Proc. IEEE* 104, 5 (May 2016), 1039–1057.

[54] O. Meynard, D. Réal, S. Guilley, F. Flament, J-L. Danger, and F. Valette. 2010. Characterization of the electromagnetic side channel in frequency domain. In *International Conference on Information Security and Cryptology*. Springer, 471–486.

[55] C. Mims. 2013. If it ain't broke, of course Apple is engaging in planned obsolescence. http://qz.com/141297/of-courseapple-is-engaging-in-planned-obsolescence. (2013).

[56] Oracle. 2006. OpenSPARC T1. http://www.oracle.com/technetwork/systems/opensparc/opensparc-t1-page- 1444609.html. (2006).

[57] N. Patel, P. Krishnamurthy, H. Amrouch, J. Henkel, M. Shamouilian, R. Karri, and F. Khorrami. 2017. Towards a New Thermal Monitoring Based Framework for Embedded CPS Device Security. (2017). submitted for journal publication.

[58] D. Paul-Pena, P. Krishnamurthy, R. Karri, and F. Khorrami. 2017. Process-aware side channel monitoring for embedded control system security. In *IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*.

[59] C. Rampell. 2013. Cracking the Apple trap. http://www.nytimes.com/2013/11/03/magazine/why-apple-wants-to-bust-your-iphone.html. (2013).

[60] C. Rechberger and E. Oswald. 2004. Practical template attacks. In *International Workshop on Information Security Applications*. Vol. 3325. Springer, 440–456.

[61] J. H. Reed and C. R. A. Gonzalez. 2012. Enhancing Smart Grid cyber security using power fingerprinting: Integrity assessment and intrusion detection. In *International Workshop on Future of Instrumentation*. 1–3.

[62] J. Robertson and M. Riley. 2014. Mysterious '08 Turkey pipeline blast opened new cyberwar. [Online]: http://www.bloomberg.com/news/articles/2014-12-10/mysterious-08-turkey-pipeline-blast-opened-new-cyberwar. (Dec 2014).

[63] D. Rodopoulos, S.B. Mahato, V.V. de Almeida Camargo, B. Kaczer, F. Catthoor, S. Cosemans, G. Groeseneken, A. Papanikolaou, and D. Soudris. 2011. Time and workload dependent device variability in circuit simulations. In *IEEE International Conference on IC Design & Technology (ICICDT)*. DOI: http://dx.doi.org/DOI:10.1109/ICICDT.2011.5783193

[64] P. Rohatgi. 2009. Improved techniques for side-channel analysis. In *Cryptographic Engineering*. Springer, 381–406.

[65] M. Rosoff. 2012. Microsoft: Apple makes old iPhones 'unusably slow' on purpose. (2012). http://www.businessinsider.com/microsoft-apple-makes-old-iphones-unusably-slow-on-purpose-2012-3

[66] D. K. Schroder and J. A. Babcock. 2003. Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing. *Journal of Applied Physics* 94, 1 (July 2003), 1–18. DOI: http://dx.doi.org/10.1109/MDT.2003.1232254

[67] H. Skipworth. 2012. The myth of the Sony kill switch. http://www.telegraph.co.uk/technology/news/7054587/Themyth-of-the-Sony-kill-switch.html. (2012).

[68] F. Standaert. 2010. Introduction to side-channel attacks. In *Secure Integrated Circuits and Systems*. Springer, 27–42.

[69] R. J. Turk. 2005. Cyber Incidents Involving Control Systems. [Online]: https://inldigitallibrary.inl.gov/sti/3480144.pdf. (Oct. 2005).

[70] V. van Santen, H. Amrouch, N. Parihar, S. Mahapatra, and J. Henkel. 2016. Aging-aware Voltage Scaling. In *Design, Automation and Test in Europe*. EDA Consortium, 576–581.

[71] V. M. van Santen, J. Martin-Martinez, H. Amrouch, M. Nafria, and J. Henkel. 2017. Reliability in Super- and Near-Threshold Computing: A Unified Model of RTN, BTI and PV. *IEEE Transactions on Circuits and Systems-I (TCAS-1)* (2017). DOI: http://dx.doi.org/10.1109/TCSI.2017.2717790

[72] M. Vuagnoux and S. Pasini. 2009. Compromising Electromagnetic Emanations of Wired and Wireless Keyboards. In *USENIX Security Symposium*. 1–16.

[73] W. Wang, S. Yang, S. Bhardwaj, S. Vrudhula, F. Liu, and Y. Cao. 2010. The Impact of NBTI Effect on Combinational Circuit: Modeling, Simulation, and Analysis. *IEEE Transactions on Very Large Scale Integration Systems* 18, 2 (2010), 173–183. DOI: http://dx.doi.org/10.1109/TVLSI.2008.2008810

[74] X. Wang and R. Karri. 2013. Numchecker: Detecting kernel control-flow modifying rootkits by using hardware performance counters. In *IEEE/ACM Design Automation Conference*. 79:1fi?!79:7.

[75] X. Wang and R. Karri. 2016. Reusing Hardware Performance Counters to Detect and Identify Kernel Control-Flow Modifying Rootkits. *IEEE Transactions on Computer-Aided Design* 35, 3 (March 2016), 485–498.

[76] X. Wang, C. Konstantinou, M. Maniatakos, and R. Karri. 2015. ConFirm: Detecting firmware modifications in embedded systems using Hardware Performance Counters. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. 544–551.

[77] G Wolrich, E. McLellan, L. Harada, J. Montanaro, and R. Yodlowski. 1984. A high performance floating point co-processor. *IEEE Journal of Solid-State Circuits* 19, 5 (May 1984), 690–696.

[78] T. Worstall. 2013. Certainly there is planned obsolescence in Apple's iKit it is just not planned by Apple. http://www.forbes.com/sites/timworstall/2013/10/31/certainly-theres-planned-obsolescence-in-apples-ikit-its-just-not-planned-by-apple. (2013).