Introduction to Cyber-Physical System Security: A Cross-Layer Perspective

Jacob Wurm, Yier Jin, *Member, IEEE*, Yang Liu, Shiyan Hu, *Senior Member, IEEE*, Kenneth Heffner, Fahim Rahman, and Mark Tehranipoor, *Senior Member, IEEE*

Abstract—Cyber-physical systems (CPS) comprise the backbone of national critical infrastructures such as power grids, transportation systems, home automation systems, etc. Because cyber-physical systems are widely used in these applications, the security considerations of these systems should be of very high importance. Compromise of these systems in critical infrastructure will cause catastrophic consequences. In this paper, we will investigate the security vulnerabilities of currently deployed/implemented cyber-physical systems. Our analysis will be from a cross-layer perspective, ranging from full cyber-physical systems to the underlying hardware platforms. In addition, security solutions are introduced to aid the implementation of security countermeasures into cyber-physical systems by manufacturers. Through these solutions, we hope to alter the mindset of considering security as an afterthought in CPS development procedures.

Index Terms—Cyber-physical system, hardware security, vulnerability

1 Introduction

Research relating to cyber-physical systems (CPS) has recently drawn the attention of those in academia, industry, and the government because of the wide impact CPS have on society, the economy, and the environment [1]. Though still lacking a formal definition, cyber-physical systems are largely referred to as the next generation of systems that integrate communication, computation, and control in order to achieve stability, high performance, robustness, and efficiency as it relates to physical systems [2]. While ongoing research focuses on achieving these goals, security within CPS is largely ignored [1]. Cyber-physical systems are in the process of being widely integrated into various critical infrastructures, however given the lack of countermeasures, security breaches could have catastrophic consequences. For example, if communication channels within a power grid are compromised, the whole power grid may become unstable, possibly causing a large-scale cascaded blackout. In fact, the emergence of smart grids may further complicate the problem if security is not considered during the smart grid construction process [3].

In addition to security concerns, CPS privacy is another serious issue. Cyber-physical systems are often distributed across wide geographic areas and typically collect huge amounts of information used for data analysis and decision making. Data collection helps the system make decisions through sophisticated machine learning algorithms. Breaches in the data collection process could lead to wide-scale data leakage, much of which is private or sensitive information related to national security. Breaches can occur in different stages of the system's operation, including data collection, data transmission, data operation, and data storage. Most current CPS design methodologies do not consider data protection, which puts collected data in jeopardy.

In this paper, we analyze cyber-physical systems from a cross-layer perspective with security in different layers being considered. More importantly, we will have a detailed discussion about the security considerations made in current CPS structures. Through this discussion we will be able to depict a full map of security needs for each layer. Different from previous work that treats CPS as one entity and tries to develop security methods for the entire system, we identify the different security challenges present in each layer and summarize countermeasures. Specifically, three different layers will be introduced in this paper ranging from the home automation systems to underlying/low-level hardware security:

- *Home automation systems.* Home automation systems are important components of future smart grid implementations and play a critical role in our daily lives. We will introduce possible attack vectors on home automation systems along with countermeasures to protect the system against various attacks.
- *Smart device security in CPS.* Smart devices comprise the backbone of CPS construction, however, security in these devices is often seen as an afterthought. Because of this mindset, devices are manufactured

- *J. Wurm and Y. Jin are with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816. E-mail: jacob.wurm@knights.ucf.edu, yier.jin@eecs.ucf.edu.*
- *Y. Liu and S. Hu are with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI 49931. E-mail: {yliu18, shiyan}@mtu.edu.*
- *K. Heffner is with Honeywell International, Inc., Clearwater, FL 33764-7290. E-mail: kenneth.h.heffner@honeywell.com.*
- *F. Rahman and M. Tehranipoor are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611. E-mail: fahim034@ufl.edu, tehranipoor@ece.ufl.edu.*

Manuscript received 6 Dec. 2015; revised 5 Mar. 2016; accepted 21 Apr. 2016. Date of publication 17 May 2016; date of current version 14 Sept. 2017. Recommended for acceptance by S. Bhunia. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TMSCS.2016.2569446

without proper security considerations. In this paper, security vulnerabilities in both commercial and industrial smart devices will be discussed. Design solutions are also proposed in order to enhance the resilience of smart devices.

- *Hardware security in CPS.* Hardware security is another important component of CPS security. First, the compromise of underlying hardware components through Trojans and backdoors can invalidate circuit- or system-level security protection methods. Second, security-enhanced hardware can play an active role in CPS protection, offering effective and efficient solutions.

A plethora of security and privacy solutions exist for the aforementioned layers of cyber-physical systems. Solutions discussed in this paper deal with network-level security, device security enhancement, physical unclonable functions (PUFs), machine learning approaches, and firmware diversity. The rest of this paper is organized as follows: Section 2 introduces the risks of cyber incidents. Section 3 focuses on home automation systems, Section 4 discusses smart device security and its potential attack vectors, and Section 5 presents the hardware security primitives for CPS security. Conclusions are drawn in Section 6.

2 Cyber Physical Security Overview

Cyber security risks are prevalent in today's information age, and new cyber incidents appear regularly in the news. In fact, many people may have been directly affected by cyber incidents [4]. Most notably, as much as one-third of the population of the United States was impacted due to the recent cyber attack on the retail store Target [5]. In this situation, hackers attacked the system with credentials stolen from a Target vendor [6]. The type of attack that impacted Target and their consumers is but one example of the numerous methods by which cyber attacks may be carried out. While the mega-breaches, like Target, grab the national headlines, smaller breaches are still costly, averaging $5.4 million in 2012, and the average cost of data theft in the United States in 2012 was $188 per customer account [7]. There has been a significant increase in attacks on cyber physical systems (CPS) as evidenced through public information. The average American company fielded a total of 16,856 attacks in 2013 [8]. Industry data breaches and cyber attacks increased in 2014 by 23.9 percent compared with 2013 to 761 reported breaches exposing 83,176,279 records [9].

These cyber attacks are costly to consumers as well as to the nation. More importantly, our nations critical infrastructure is dependent upon information technology and communication systems, as well as the supply chains that support them.

3 Smart Home Security

The *smart home* has become an indispensable component of the smart grid infrastructure, specifically on the residential side. Due to the massive deployment of advanced metering infrastructure (AMI), smart home systems employ controlling and scheduling techniques to facilitate the management of household activities in an effort to save energy. Because there are a large number of residential customers, a slight energy savings in each home can result in a significant reduction in energy consumption for the entire power grid. It has been demonstrated that a 5 percent energy savings on the residential side across the U.S. can lead to a reduction in energy consumption and carbon emissions similar to removing 52 million cars [10]. Despite the benefits of adopting smart home systems, they also pose security concerns.

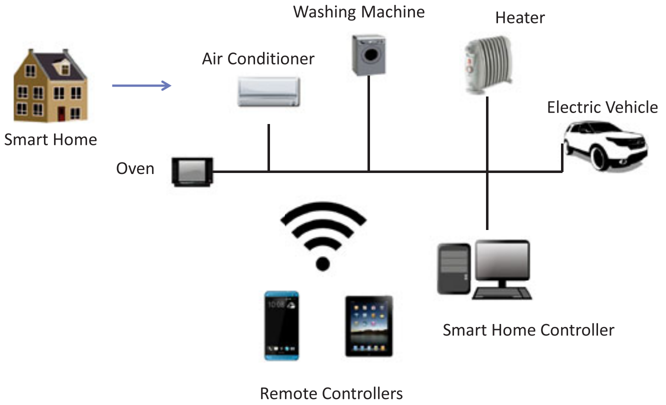


Fig. 1. A typical smart home consists of various smart home appliances which are connected to a smart home controller. The customer can also control the smart home remotely using mobile devices such as smart phones and tablets.

3.1 Smart Home Infrastructure

A smart home infrastructure allows for automatic control of household activities as well as control over the amount of electricity used. They employ various communication and control techniques to enable automatic and remote management of household appliances. In a smart home system, household appliances are usually connected to a centralized controller which schedules energy usage based on information such as sensor data and price data from utility companies. Fig. 1 shows an example of a smart home infrastructure. Remote control of the system is enabled by mobile applications along with wireless communication channels such as WiFi and Zigbee. There are industries dedicated to the development of such mobile applications and their corresponding software frameworks such as Google and Apple. In particular, Apple has developed the HomeKit framework for iOS, which provides a convenient interface for the remote control of devices in a smart home, and stores each user's configuration online using iCloud [11]. Google's Nest ecosystem also focuses on hardware-based smart home controllers [12]. The Google Nest Thermostat uses big data techniques to analyze the historical data of weather, energy usage and temperature to optimize the control of heating and air conditioning. This helps reduce the electricity bill for heating and air conditioning by 20 percent on average.

Currently in the U.S. energy market, utilities usually design their pricing based on historical energy consumption data. Pricing data, known as *guideline pricing* is given to customers one day in advance in an effort to influence their energy usage. If the price of energy varies hourly, this is known as *dynamic pricing*. Fig. 2 shows the dynamic energy pricing provided by Ameren Illinois Corporation [13]. Based on the guideline energy prices provided by utilities, various techniques have been developed to shift energy

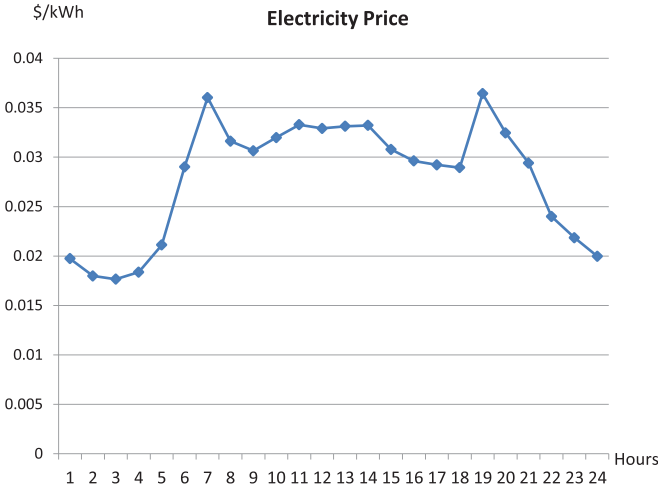


Fig. 2. Dynamic electricity price provided by Ameren Illinois Corporation.

consumption away from peak-pricing hours in an effort to lower electricity bills. These techniques are utilized by a smart home controller when controlling household appliances, which is known as *smart home scheduling*. The utilities can also benefit from smart home scheduling since it helps balance the energy load on the power grid given a sophisticated guideline energy pricing model. This can effectively mitigate the burden of energy generation on the utilities, instead spreading the energy load throughout the day.

3.1.1 Smart Home Scheduling Techniques

In existing literature, various smart home scheduling techniques have been developed. Smart home scheduling for a specific customer mainly depends on the configuration of their home appliances. In [14], a linear programming technique is proposed to solve the smart home scheduling problem based on the linear pricing model. Their technique has been improved further to address the uncertainty of renewable energy and the workload of household appliances. While most existing works assume the energy usage of household appliances is continuous, [15] developed a dynamic programming-based technique to handle household appliances with discrete power levels, which is more feasible for smart home scheduling in practice.

In communities consisting of multiple customers, the total energy bill is computed based on the total energy consumption of the community, which is comprised of each customer's individual usage during the past time window. This means that the energy bill depends on the total usage of all customers in the community. Thus, when smart home scheduling is deployed in a community, a game theoretic framework is commonly developed to address the mutual impact of the customers. The game theoretic multiple customer smart home scheduling technique is an iterative procedure where in each iteration, each customer schedules his/her own energy consumption according to other customers' usage in the previous iteration. This significantly increases the communications overhead of the smart home system. The work [15] proposes a hierarchical framework to effectively reduce the communications overhead, which is further deployed in city level smart home scheduling. Existing research has demonstrated that the smart home scheduling technique can reduce the electricity bill of customers by 34.3 percent and the global peak to average ratio (PAR) of the energy load by 35.9 percent [10].

3.2 Smart Home Cyberattacks

Smart home cybersecurity has started to attract significant research interests. Hardware backdoors can be leveraged by hackers to launch cyberattacks (see Fig. 3). Smart device vulnerabilities have already been reported in the public media [16], [17]. For example, the Google Nest thermostat has recently been proven to be insecure [16]. The Google Nest thermostat can be exploited to allow attackers to remotely control the device (see Section 4.2 for more details).

In fact, cyberattacks on smart devices are commonly reported. According to the report of CNN, a long list of smart devices such as security cameras, baby monitors, smart TVs, smart door locks, power outlets, and even smart toilets contain security vulnerabilities which may be exploited [18].

Similar to other smart devices, smart meters can also be compromised so that hackers can remotely control the device. Modern smart meters are usually based on microcontrollers and utilize advanced embedded operation systems. For example, the smart meter manufactured by Texas Instruments is based on the automatic meter reading (AMR)/AMI platform [19], which supports two-way communications to enable the periodical remote firmware updates. This backdoor may be utilized by the malicious hackers to launch cyberattacks and execute malicious code (a detailed example of smart meter security analysis can be found in Section 4.3).

3.2.1 Pricing Cyberattack

In the context of the smart home, the guideline electricity price is crucial since customers reference it to conduct smart home scheduling. Thus, if the guideline electricity price is manipulated, schedulers will be misled, which can impact the energy load in the power grid. On the other hand, electricity bills depend on the energy consumption in a past time window. Thus, if the energy load is impacted, the electricity bills of the customers will be influenced as well. Basically, a malicious attacker can manipulate the guideline electricity price to mislead the customers, which is known as a pricing cyberattack [20]. The pricing cyberattack would be launched for the following reasons:

- The attacker can manipulate the guideline electricity price to create a peak energy load. Customers tend to use more energy when the electricity price is lower. Thus, if the attacker manipulates the electricity price and sets it to zero in a certain time slot, a massive energy load will be scheduled during that period, leading to an overexertion of generation capabilities of the grid. As demonstrated in [20], the pricing cyberattack targeting the creation of a peak energy load can increase the peak-to-average ratio (PAR) of the energy load by 35.7 percent, which significantly unbalances the energy load in the power grid. This can impact the stability of the power grid and even lead to a larger area blackout through cascading effects.
- The attacker can manipulate the guideline electricity price to reduce his/her own electricity bill at the cost of increasing those of others. Note that the utilities

use the guideline electricity price to facilitate smart home scheduling and bill the customers based on the real-time electricity price, which is computed from real-time energy usages in a past time window. Thus, if the attacker intends to schedule the energy consumption during expensive time slots, he/she can further increase the guideline price in these time slots. This increase will mislead the smart home schedulers, causing them to avoid using energy during that time. Thus, the real-time energy load during these time slots is low, which lowers the real-time electricity price. Subsequently, the hacker schedules the energy consumption at these time slots, thus saving a significant portion of their electricity bill. As demonstrated by the work [20], the pricing cyberattack for bill reduction can reduce the attacker's electricity bill by 34.3 percent and increase the electricity bill of other customers by 7.9 percent on average.

3.2.2 Energy Theft

In energy theft, a malicious hacker can manipulate the measurement of energy consumption of the smart meter and decrease it. This will significantly reduce the attacker's electricity bill since energy consumption is billed based on the reported measurement. If each customer is billed individually, the utility will suffer a significant loss of profit. Within a community, customers are usually billed based on the community's total energy consumption and they share the electricity bill based on their individual energy usage. Thus, electricity bill reduction will result in the bill increase of other customers. In addition, the real energy load could be much higher than the reported measurement. Thus, the utility needs to inspect the smart meters in the local area, which induces a large labor cost. If the mismatch is significant, the utility companies may have to shut down the energy supply [21].

Smart home also suffers from privacy threats in addition to cybersecurity issues. A hacker can gain access of the smart home controller through AMI and reveal the energy usage of each home appliance. Such information can be utilized for data onboarding to make profit. In fact, the hackers can analyze the energy usage of each home appliance only through measuring the voltage and current, even without really attacking the smart homes. This is called non-intrusive load monitoring [22]. According to the most recent research, machine learning techniques and probabilistic models are commonly used to analyze the contributions of each home appliance to the total energy load based on their energy consumption signatures [23]. Potential solutions have been studied to address the privacy leakage of smart homes. In particular, a rechargeable battery can be used as a relay to store the electricity energy before supply the energy usage of the smart home such that the energy consumption signature of each home appliance is not exposed [24].

3.3 Multi-Level Smart Home Security Protection

3.3.1 Device Level Protection

The straightforward approach for building highly secure hardware infrastructures for defense against cyberattacks is to design hardware platforms secured with resilient architectures. Due to the uniqueness of hardware in terms of the low update frequency compared to its firmware/software counterparts, hardware security must be ensured from the very beginning of the design and manufacturing stages [25]. In recent research, a cross-boundary security platform was developed through co-designing a secure Linux kernel running on a security-enhanced SPARC V8 compatible processor [26], [27]. This platform ensures trusted execution of privileged kernel extensions and device drivers, which may be used for highly-secure smart devices development which supports customizable, user-friendly security policies and monitoring capabilities at the OS-level.

3.3.2 System Level Protection

In addition to device level protection, system level defense techniques have been proposed in existing works. In [28], the single event and long term detection techniques are developed based on support vector regression (SVR) and partially observable Markov decision process (POMDP), respectively. Note that the guideline price curves usually tend to be similar in the short term. Thus, the cyberattack can be detected through analyzing the historical electricity price and comparing it with the received guideline price. The single event detection technique in [28] employs SVR to predict the guideline electricity price from historical data. The electricity bill and PAR are computed using the predicted and received guideline electricity prices, respectively. A cyberattack is reported if the electricity bill and PAR computed from the received guideline electricity price are significantly higher than those computed from the predicted price.

Further, a long term detection technique is developed in [28] using the POMDP. The POMDP technique has properties such as the belief state, expected reward and policy transfer graph to estimate the impact of the possible future cyberattacks. Based on the Markov model of the cyberattacks, the POMDP technique computes the optimal action (e.g., check the smart meters or ignore the cyberattack report) that maximizes the expected reward.

Renewable energy generation is important to the smart home infrastructure. In addition, the net metering protocol and distributed storage facilitate the storage of the excessively generated renewable energy and sell it back to the power grid. However, this impacts the energy demand from customers, which further influences electricity pricing. Based on the detection framework in [28], the work [29] investigates the impact of net metering on electricity pricing to make the pricing prediction more accurate. This significantly improves the detection accuracy of the POMDP based detection framework.

Inserting feeder remote terminal units (FRTUs) is a common solution for energy theft cyberattacks. The FRTU takes measurements of energy flow in the distribution subnetwork and compares it with measurements from smart meters. If the mismatch is significant, on-site inspection will be performed in the corresponding sub-networks. The FRTU insertion techniques aim to minimize the investment for installing the FRTUs while maintaining the detection accuracy. The work [21] has developed a cross entropy based optimization method to compute the locations of the FRTUs that optimize the detection rate while limiting

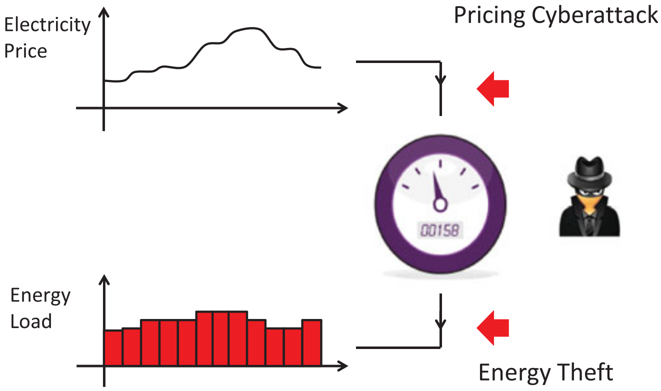


Fig. 3. Pricing cyberattacks and energy theft.

the cost in FRTU deployment. A dynamic programming algorithm is proposed in [21], which further improves the computational efficiency of the detection. Given historical anomaly rates of smart meters, the dynamic programming algorithm deploys FRTUs in the distribution network to minimize the labor overhead due to checking smart meters. Such an algorithm can handle the FRTU insertion in a large scale distribution network efficiently.

Since a cyber-physical system heavily involves interactions and communications among different components, the system reliability largely depends on the communication security. In recent years, physical unclonable function (PUF), which enables the challenge-response authentication for secure communications, gains significant popularity. Such a technology can be naturally deployed in a cyber-physical system to enhance the security at both the device level and the system level. On the other hand, emerging technologies such as carbon nanotube based circuit designs can be leveraged in developing highly secure PUFs due to the strong variations induced from the fabrication process. This motivates the recent works [30], [31] to develop such techniques for cyber-physical system applications.

4 Smart Device Security in Cyber-Physical Systems

As discussed earlier, there are significant vulnerabilities present in modern cyber-physical systems at the system level. While many of the attacks are derived from improper/insecure communication protocols and system configurations, the widespread usage of smart devices with security vulnerabilities is also a major cause of the deterioration of high-level protection schemes. However, the security analysis and protection of smart devices has long been ignored in CPS security research. In order to better understand the security vulnerabilities present within modern smart devices and the disastrous consequences to entire systems if the underlying devices are compromised, different types of security vulnerabilities and design loopholes in modern smart devices will be introduced in this section. Real world devices will be used as examples when we elaborate different categories of security vulnerabilities [32], [33].

4.1 Security Threat Taxonomy

Security threats that affect smart devices can be further categorized into six types based on how attacks are performed on the device. A full taxonomy of these security threats is shown in Fig. 4, and the components of the taxonomy are listed below.

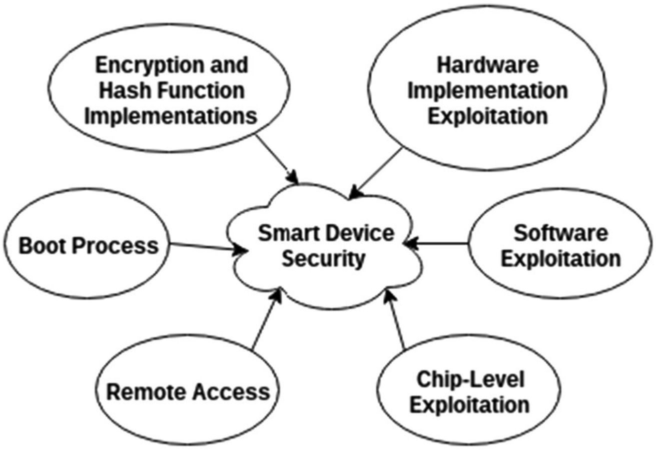


Fig. 4. Security vulnerabilities in smart devices.

- *Boot Process Vulnerabilities*. The boot sequence is one of the main targets of attack, as many of the high-level protection mechanisms are unable to be executed during the boot process. Since these mechanisms are not present, it leaves the system open for attack, which makes this a critical area to protect. For example, the attack on the iPhone's bootloader leads to a chain-of-trust exploit [34]. Mitigation methods to this type of vulnerability were discussed in [35], [36].
- *Hardware Exploitation*. Hardware level exploitation is a critical point for security as most security protection implementations are located at the software or firmware levels. These attacks target the hardware implementations themselves, which involve looking for debugging ports left open by manufacturers, reflashing external memory, timing attacks, etc. For example, the exploits on Xbox 360 allows systems to downgrade to a vulnerable kernel version through a timing attack [37]. In order to prevent this type of attacks, various countermeasures have been developed, e.g., the protection methods to prevent timing attacks [38].
- *Chip-Level Exploitation*. Chip-level exploitation of integrated circuits, including semi-invasive and invasive intrusions are a serious threat to smart devices, as trusted boot sequences rely on trusted on-chip assets. For a long time, encryption/decryption keys, and other sensitive information was stored on-chip as it was considered a secure means of storage. Newly developed invasive methods can reveal valuable assets stored in the chip, and may compromise any protocols utilizing the secret information. For example, by "bumping" the internal memory on an Actel ProASIC3 FPGA, researchers were able to extract the stored AES key [39].
- *Encryption and Hash Function Implementations*. Encryption and hash functions are used in smart devices to secure passwords and other sensitive information, in addition to playing a key role in device communication and authentication. These

functions are mathematically proven to be secure and robust, however side-channel attacks and information based cryptanalysis methods are threatening their integrity. In addition, improper implementations of these functions and the utilization of cryptographically weak encryption algorithms threaten the security of these devices. For example, the Sony PlatStation 3 firmware was downgraded due to a series of vulnerabilities in weak cryptographic applications [40], [41]. Interestingly, while the problems have been repeated in modern smart devices, the mitigation methods have already been proposed decades ago [42].

- *Backdoors in Remote Access Channels*. Smart devices are often equipped with channels that allow for remote communication and debugging after manufacturing. These channels are also used for over-the-air (OTA) firmware upgrades. Though these channels are extremely useful, their implementations are not always secure. During development, manufacturers may leave in APIs which allow arbitrary command execution, or developers may not properly secure the communications channel. Through this attack vector, attacks may be able to remotely obtain the status of the device, or even control the device. A modern example of a backdoor in a remote channel is the Summer Baby Zoom WiFi camera, which has hard-coded credentials for administrator access [43]. Other remote exploits were applied in multiple smart house devices [44]. Efforts to mitigate these vulnerabilities include requiring users to change default credentials before usage, sanitizing string input to avoid remote command execution, etc.
- *Software Exploitation*. Software-level vulnerabilities in smart devices are similar to those in traditional embedded systems and general computing systems. Because smart device software stacks are often derived from the general computing domain, any software vulnerabilities found in the general computing area will also affect these devices. Therefore, software patches are required to update smart devices against known software-level attacks. Recent examples include a stack-based buffer overflow attack in glibc [45]. Methods to mitigate software exploitation attacks often follow those developed in general computing areas [46], [47]. However, as discussed in [48] that these solutions may not fit in smart devices due the resource constraints.

Throughout the rest of this section, we will introduce in detail some of the device-level security vulnerabilities. In addition, real-world examples of these vulnerabilities will be elaborated on in an effort to emphasize the impact these vulnerabilities have on real devices.

4.2 Boot Process Hijacking

Boot process hijacking invalidates software level protection schemes before they are properly installed and loaded. In this case, attackers try to break the normal boot process through the vulnerabilities within the chain-of-trust and install customized userland images or kernel modules. Malicious payloads can be inserted into the kernel modules and/or userland filesystems. One example of this type of attack is the compromise of the Google Nest Thermostat [16], [33].

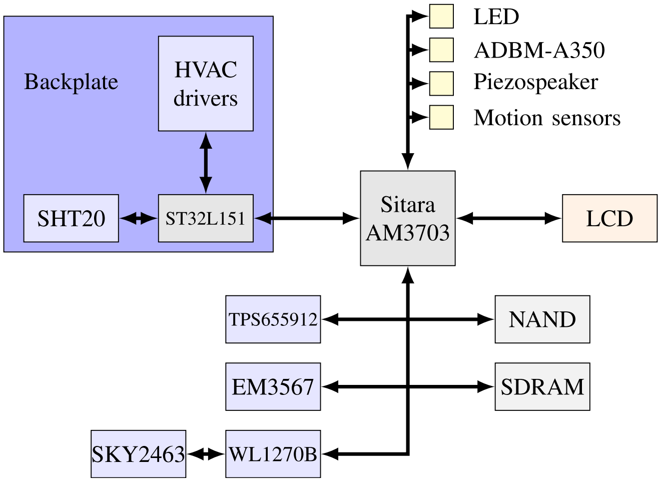


Fig. 5. Device map of the Nest Thermostat [33].

The Nest Thermostat is a smart device designed to control a standard heating, ventilation and air conditioning (HVAC) unit based on heuristics and learned behavior. Coupled with a WiFi module, the unit is able to connect to the user's home or office network and interface with the Nest Cloud, thereby allowing for remote control of the unit. The thermostat is divided into two main components, a backplate which interfaces with the HVAC unit and a front plate which presents the main user interface. The largest part count is found in the front plate of the thermostat, which is driven by a Texas Instruments Sitara AM3703 system-on-chip (SoC) [49], interfacing directly with a Micron ECC NAND flash memory module, a Samsung SDRAM memory module and a LCD screen. Fig. 5 shows the device's internal components and the overall device configuration.

Upon normal powering on process, the Sitara AM3703 starts to execute the code in its internal ROM. This code initializes the most basic peripherals, including the General Purpose Memory Controller (GPMC). It then looks for the first stage bootloader, `x-loader`, and places it into SRAM. Once this operation finishes, the ROM code jumps into `x-loader`, which proceeds to initialize other peripherals and SDRAM. Afterwards, it copies the second stage bootloader, `u-boot`, into SDRAM and proceeds to execute it. At this point, `u-boot` initializes the remaining subsystems and executes the `uImage` in NAND flash with the configured environment. The system finishes booting from NAND flash as initialization scripts are executed and services are run, culminating with the loading of the Nest Thermostat proprietary software stack. Fig. 6 shows the normal boot sequence of the device.

The device boot configuration is set by six external pins, `sys_boot[5:0]`. After power-on reset, the value of these pins is latched into the `CONTROL.CONTROL_STATUS` register. Table 1 describes the boot selection process for a selected set of configurations.

After performing basic initialization tasks, the on-chip ROM may jump into a connected execute in place (XIP) memory, if the `sys_boot` pins are configured as such. This boot mode is executed as a blind jump to the external

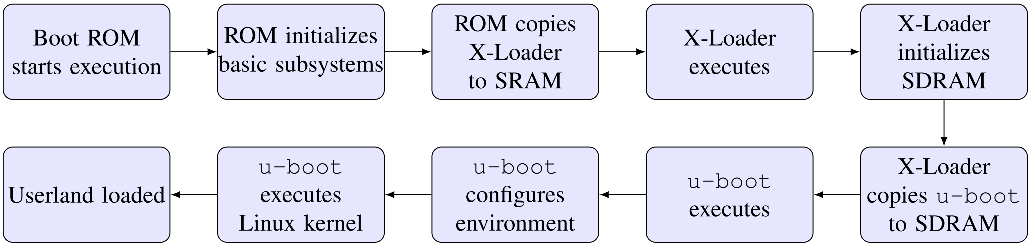

Boot ROM starts execution
ROM initializes basic subsystems
ROM copies X-Loader to SRAM
X-Loader executes
X-Loader initializes SDRAM
X-Loader copies u-boot to SDRAM
u-boot executes
u-boot configures environment
u-boot executes Linux kernel
Userland loaded

Fig. 6. Standard nest thermostat boot process.

addressable memory as soon as it is available. Otherwise, the ROM constructs a boot device list to be searched for boot images and stores it in the first location of available scratchpad memory. The construction of this list depends on whether or not the device is booting from a power-on reset state. If the device is booting from a power-on reset, the boot configuration is read directly from the `sys_boot` pins and latched into the `CONTROL.CONTROL_STATUS` register. Otherwise, the ROM will look in the scratchpad area of SRAM for a valid boot configuration. If it finds one, it will utilize it, otherwise it will build one from *permanent devices* as configured in the `sys_boot` pins. Through this vulnerability, attackers can send a modified `x-loader` into the device, coupled with a custom `u-boot` crafted with an argument list to be passed to the on-board kernel. Arbitrary payloads can then be inserted into the device through the custom `u-boot` image [33].

4.3 Hardware Exploitation

Hardware level exploitation is another type of attack targeting hardware platforms of smart devices leveraging vulnerabilities within debugging ports, side-channel information, and hardware-based authentication schemes. The main goal of these attacks is to retrieve sensitive information stored in hardware modules or to bypass hardware protection mechanisms. Hardware exploitation is also used to invalidate device authentication schemes for illegitimate cloud service access. One example of this type of attack is the ID manipulation on the Itron Centron smart meter as shown in Fig. 7 [32].

The primary functionality of this Itron Centron CL200 smart meter is to measure a customer's energy usage and report the collected information through an RF channel to a nearby meter reader or to a local substation. This information is then used to charge the customer for their energy usage, and may also be used to get statistics on community energy usage.

One attack scenario on smart meter is to modify the smart meter ID in order for a meter reader to read the wrong ID for the device. Through the on-board unprotected debugging port, it becomes possible to identify the location of the device ID. In fact, researchers found that the meter stores its ID on an external EEPROM, which does not contain any read or write protection. By looking at the ID of the meter and cross-referencing it with the data from the EEPROM dump, the ID was located and modified [32]. Given the known smart meter ID location and the access to the EEPROM, attackers can easily re-flash the EEPROM. As a result, the meter was able to represent itself as any other smart meter.

TABLE 1
Selected Boot Configurations

sys_boot[5:0]	First	Second	Third	Fourth	Fifth
001101	XIP	USB	UART3	MMC1	
001110	XIPwait	DOC	USB	UART3	MMC1
001111	NAND	USB	UART3	MMC1	
101101	USB	UART3	MMC1	XIP	
101110	USB	UART3	MMC1	XIPwait	DOC
101111	USB	UART3	MMC1	NAND	

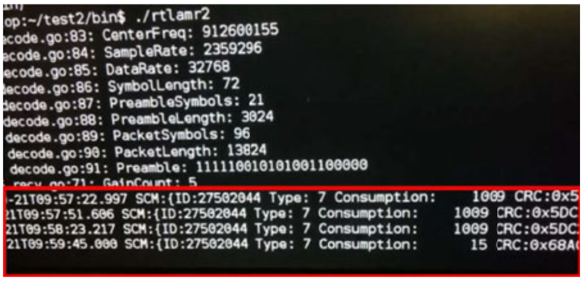
Fig. 8 details the results of the ID change. The first three entries shown in the red box are from one meter under testing. Another meter is then connected which before modification has its own unique ID. After modification, the second meter broadcasts with the same ID number as the first meter, as shown in the fourth entry in Fig. 8.

4.4 Weak Encryption and Hash Functions

Many smart device implementations rely on low-power components, and therefore are incapable of performing computationally intensive tasks. As for encryption, many devices either utilize low-complexity encryption algorithms or lack encryption entirely. One example of a weak encryption implementation causing device-level security vulnerabilities is the Haier SmartCare home automation system [32].

The Haier SmartCare is a smart device designed to control and read information from various sensors placed throughout a user's home which include a smoke detector, a water leakage sensor, a sensor to check whether doors are open or closed, and a remote power switch. These sensors are connected through the ZigBee protocol. The primary function of this device is to allow the user to better monitor their homes when they are away and to get alerts based on sensor information. This smart device takes advantage of remote debugging, where developers are able to modify

Fig. 7. Itron Centron CL200 smart meter (Credit: Itron).



```
op:~/test2/bin$ ./rtlamr2
ecode.go:83: CenterFreq: 912600155
ecode.go:84: SampleRate: 2359296
ecode.go:85: DataRate: 32768
decode.go:86: SymbolLength: 72
decode.go:87: PreambleSymbols: 21
decode.go:88: PreambleLength: 3024
decode.go:89: PacketSymbols: 96
decode.go:90: PacketLength: 13824
decode.go:91: Preamble: 111110010101001100000
recv.go:71: GainCount: 5
-21T09:57:22.997 SCM:{ID:27502044 Type: 7 Consumption:    1009 CRC:0x5
21T09:57:51.606 SCM:{ID:27502044 Type: 7 Consumption:    1009 CRC:0x5DC
21T09:58:23.217 SCM:{ID:27502044 Type: 7 Consumption:    1009 CRC:0x5DC
21T09:59:45.008 SCM:{ID:27502044 Type: 7 Consumption:      15 CRC:0x68A
```

Fig. 8. Demonstration of the security vulnerability on the meter.

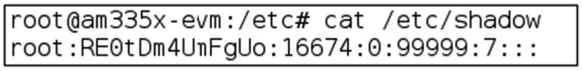
parameters and analyze the system operation remotely. Their implementation utilizes Telnet, through which developers can log into the system. The remote access channel is theoretically secure since the root account through the Telnet channel is password-protected. However, the password hash reveals that the device is using DES encryption on the password while also not using a salt (see Fig. 9).

This means that the password is truncated to a maximum of eight characters for password hashing. Given the small space of all possible passwords, a brute force attack becomes possible. The total keyspace for a DES password using printable ASCII characters is $\sum_{i=0}^{8} 96^i$. This is a medium sized keyspace, and can cracked within hours using personal computers utilizing graphics cards with parallel processing capabilities. Through obtaining the root password, remote access to the device was able to be achieved. Through a remote vulnerability such as that in the SmartCare, attackers will be able to run their own code on the device. This can lead to the leakage of private user data and network data.

4.5 Smart Device Protection

Smart devices often provide a full operating system in which binaries are loaded into a userland. This simplifies the interface to the hardware and provides high level Application Programming Interfaces (APIs). The Nest Thermostat, for example, employs an embedded Linux stack which is used to launch the proprietary Nest application which relays commands to the backplate of the unit and controls the communications channels. As demonstrated in previous work, binaries can be injected into the filesystem of the unit and executed in devices that utilize this model. As such, extra protection must be added to devices that load binaries into a userland. A possible approach is to only load and execute cryptographically signed binaries. This requires the kernel to have a custom loader that verifies these binaries as they are prepared for execution. If the signature verification fails, then the binary is not run and the device is set into a fail-safe mode, notifying the user of possible tampering.

From the hardware perspective, debug interfaces also require proper protection. While debug interfaces are often left as residues from development prototypes or as test points used during manufacturing. These debug interfaces can also serve as the means to service IoT or wearable devices on the field, in order to ease repairs. However, these interfaces must be protected against attackers. Microprocessors should be enhanced with functionality restricting access to its debug ports. As such, manufacturers are able to still expose these interfaces for testing purposes and disable them before they are deployed.



```
root@am335x-evm:/etc# cat /etc/shadow
root:RE0tDm4UnFgUo:16674:0:99999:7:::
```

Fig. 9. SmartCare hashed root password.

5 Hardware Security for Cyber-Physical Systems

In this section, we discuss the vulnerabilities and security challenges that arise from the system hardware, such as integrated circuits (ICs), sensors and actuators, printed circuit boards (PCBs), etc. that comprise the core level physical architecture of cyber-physical systems and discuss some possible security primitives and countermeasures that can be employed to enhance CPS security. Traditionally, CPS are built mostly using existing designs and architectures, with hardware that are not necessarily developed, or intended, for CPS applications in the very first place. It should also be noted that CPS have inherent design challenges in terms of control, resource management, reliability, integrity and more importantly – security, and hence it requires special attention to identify the vulnerabilities and attacks and address proper solutions [50], [51].

5.1 Hardware-Based Vulnerabilities and Attacks

In a large scale system (for example, a power grid, an automated industrial factory, or even at an operating room in a hospital [52]) where multi-functional hardware components are connected to each other via a networking scheme, and more importantly, when legacy parts comprise a significant portion, it poses a serious problem for verifying the overall correctness and safety of designs at the system level. Historically, trends in CPS security are mostly dominated by cyber-security with heavyweight software and cryptographic protocols layering the higher levels of system abstraction [51], [53]. Complexity of such approaches arises at multiple temporal and spatial scales since CPS-oriented cyber security needs to address real time communication among embedded systems and sensors, the data communication layer, controlling and processing units that might not originally be designed to comply with such security protocols. Further, the vulnerabilities are more pronounced since these task-level security layers hardly consider hardware level security. A few of such hardware based vulnerabilities and attacks are as follows:

5.1.1 Theft of Cryptographic Keys

The security of a cyber-physical system largely depends on the security of the communication layer, which uses different public/private encryption systems using various cryptographic keys with additional hardware security modules (HSM) or trusted platform module (TPM) to maintain privacy and integrity. The most common approach is to store these keys into non-volatile memories from which it can be stolen if proper security measures are not taken. Once the attacker steals these valuable keys, he/she eventually can launch different cyber-attacks on the system leading to catastrophic results. Such an attack may pose similarities with key (identity) theft from smart cards, however in a different level of abstraction that comprises both hardware and software assisted attacks with possible cross-layer information flow [54], [55], [56].

5.1.2 Theft of Device Identity

As discussed before, the attackers can steal the device's ID to breach the system's integrity. It allows them to incorporate fraudulent devices into cyber-physical system and launch attacks, such as relay and replay attacks [57]. For example, one can steal the ID of a remote sensor and breach the system security by feeding fabricated (malicious) data, impersonating with the stolen ID, for which the entire system may shutdown. This poses similar, however malicious, impacts as observed in 2007 nuclear power plant shutdown incident [58].

5.1.3 Physical Tampering of System Elements

A physically tampered device can expose backdoors to an attacker breaching security and integrity, as well as impacting CPS in terms of performance and cost if undetected. For example, a physically tampered energy meter may record a lower energy consumption than the actual consumption causing financial loss for the provider, as discussed previously in Section 4.

5.1.4 Counterfeit Elements with No/Low Security

Cyber-physical systems that rely on legacy parts often require more maintenance as well as frequent replacements. It opens up the possibility of counterfeit components sneaking into the system due to lack of strong supply chain management. These counterfeit elements themselves breach CPS security and pose different vulnerabilities since they may have a very low lifetime with degraded performance, have different defects, might be out of specification, contain backdoors for remote attacks, and many other vulnerabilities [59]. As an example, a counterfeit IC with low lifetime and/or with out-of-spec performance deployed for a critical application (such as radioactivity sensor with shutdown interrupt used in a nuclear power plant) itself impose high risk to the overall system.

We note that the above vulnerabilities and attack examples are ad-hoc in nature to ones described in the previous sections. Since CPS comprise of different level of physical abstractions with over and under-lying secured communication layers (cyber in nature), nature of such hardware-oriented vulnerabilities and attacks may remain similar in different layers while they may vary in coverage and different degree of threat-levels. To ensure the hardware security of CPS, it is essential that all possible attacks and vulnerabilities are taken into consideration.

5.2 Hardware Security Primitives and Countermeasures

It is apparent that secured hardware plays an essential part in maintaining the integrity of CPS to provide security from within. 'Upgrading' all hardware to a more 'secured' version is not viable, since it does not offer the same flexibility that software/firmware update patches do, involves higher labor and hardware costs, and in many cases the system consists of a significant amount of legacy parts which have been integrated, as well as evolved, into CPS in an ad-hoc way. In such cases various hardware security primitives come into play to ensure the security of devices, as well as systems. Hardware security primitives, such as Physical Unclonable Functions (PUFs) and True Random Number Generators (TRNGs), as well as design and architecture based countermeasures for hardware tampering and counterfeiting, possess a unique potential to offer solutions to various security issues that are vital, if not unique, to CPS, and might not easily be achieved via software-based higher level abstractions only. Below we discuss some existing hardware security primitives and countermeasures with possible applications in regards to CPS security:

5.2.1 Physical Unclonable Function (PUFs)

PUFs are identically designed architectures that produce non-deterministic keys/signatures using inherent physical variations resulting from the manufacturing process in elements such as transistors, interconnects, etc. Since PUFs can generate responses on the fly, they offer a volatile, less-expensive, and tamper-resistant alternative to conventional approaches that rely on storing keys in non-volatile memory [60], [61], [62].

Since CPS highly depends on the interactions of different multidimensional elements, the communication layer requires security via cryptographic protocols and authentication schemes using secret keys and unique device identities that possess high vulnerabilities from attacks as mentioned in Sections 5.1.1 and 5.1.2. PUFs can help to combat attacks based on these vulnerabilities, as it can generate the necessary keys and authentication IDs, without requiring any on-device key storage mechanism eliminating crypto-key and device identity theft. PUF based authentication protocols may range from simple challenge-response based mechanisms that can be used in a one-time authentication token, or using embedded sequences of challenge-response numbers to enable authentication [61]. Since the PUF responses can be generated with individual PUFs embedded in different chips, the authentication scheme needs to choose between different PUFs for key generation, or may use a composite system-level PUF designed for the authentication protocol.

The authors in [63] proposed a system-level PUF to have an integrated cyber defense framework for CPS. It is based on the system that describes the composite behavior of multiple PUFs to establish system level properties for security. The architecture of the system-level PUF consists of a system of embedded components, each equipped with PUF circuits, and consists of a group of readers acting as cluster heads with the communication model limited to a challenge-response system between the reader and the components. This system-level PUF can be used to make a general authentication scheme that allows the verification of the integrity of the system by ensuring integrity of each of the components. To verify the integrity of the components, the trusted party collects the response of the system-level PUF (a collection of responses from elementwise PUFs, as proposed by [63]) and verifies if overall integrity holds. Otherwise, the system will need to move to component-level authentication to determine which components caused the authentication to fail. A simple example of such a scheme might be used in checking the integrity of a printed circuit board with several elements [64]. The PUFs embedded in the elements can generate individual authentication IDs that provide an overall board ID or key used for system-level authentication. Altering the elements eventually alters the board-level ID/key compromising the security. This scheme can be taken into higher level of abstraction as well to offer a CPS-compatible security scheme.

However, since the success rate of the overall system authentication depends on the individual PUF responses, an unintentional error (due to reliability degradation from environmental variation and aging) introduced in one PUF may cause the authentication scheme to fail. Hence it requires compatible error correction techniques, which lead into relatively larger area and power overhead in digital electronic chips, and may not be readily applicable for analog devices and electro-mechanical components such as sensors and actuators.

5.2.2 True Random Number Generators (TRNGs)

TRNG is used in a wide variety of security applications most notably, generation of nonces, one time pads, LFSR seeds, and cryptographic keys [65], [66]. A TRNG generally consists of an entropy source, an entropy extraction/sampling unit and in most cases, a cryptographic conditioning unit. The entropy source is the focal point of a TRNG. As opposed to pseudo-random number generators, a TRNG relies on electrical and/or thermal processes that are inherently random to serve as its entropy source. The sources may include RTN found in scaled transistors, power supply noise, radioactive decay, latch metastability, jitter in ring oscillators and so on. The analog entropy source is then sampled using the entropy extraction/sampling unit. This could be in the form of a latch sampling a ring oscillator signal or a voltage comparator producing a digital output from comparison of a RTN-prone signal to a reference voltage [67].

TRNGs may be used in unique security applications in CPS. Since CPS have multiple elements connected in real-time, TRNGs may be used for generating random keys for one time pads in possible crypto-protocols, or creating session keys that restricts unauthorized accesses (and cyber attacks) to the cyber-physical system. As the key may be shared among numerous elements with high speed applications, the TRNG is also required to have a high throughput, high randomness, with minimal application of cryptographic hash functions.

5.2.3 Design for Anti-Tamper

Not only to software based cyber-attacks, CPS may also be vulnerable to different hardware-based attacks that may be remote or physical in nature. Design-for-anti-tamper hence plays a crucial role in preventing secrets (cryptographic keys or other valuable data) from being stolen and preventing denial of service attacks targeting CPS. Adversaries can carry out such attacks that may be invasive, semi-invasive or non-invasive in nature. Prevention of such attacks requires a proper understanding of the threat model as well as developing adequate protection mechanisms for the system [68].

Remote attacks targeting hardware may cause data leakage, or even system malfunction by fault injections (e.g., power supply and clock glitching in the system, etc.) or side channel attacks (e.g., cache timing attacks, etc.). Since CPS consists of a large number of devices of different nature, employing real-time remote attack evident/resistant schemes, in both the system and device levels, is a challenge. However, since PUF and TRNG performances vary considerably with the operating conditions (power supply, temperature, etc.), monitoring the performance (e.g., error in PUF responses, change in throughput and randomness in TRNG outcomes, etc.) may give the trusted authority some indication of out of spec operations and possible security breaches [55].

Semi-invasive and invasive attacks on large scale CPS may take different forms in comparison to attacks on traditional integrated circuits such as microprobing and reverse engineering. However, it is of high importance that sophisticated tamper-sensing mechanisms are employed to avoid any kind of physical tampering. Researchers have proposed silicon-level solutions to counteract passive and active attacks, however that does not eliminate the threat on other crucial elements like physical sensors and actuators [69]. Active sensor nets can also be employed at the device level [70], and with proper extensions, at the system level to detect any unauthorized intrusion as mentioned in Section 5.1.3. However, a universal architecture for CPS design with anti-tampering in mind needs thorough scrutiny since the CPS have a wide variety of micro and macro designs focusing on different applications [53].

5.2.4 Design for Anti-Counterfeit

Counterfeit ICs are an increasingly common problem in today’s CPS. Most of the large-scale and industrial CPS largely depend on legacy parts that need occasional replacements. They also pose security and compatibility issues with upgraded systems. Hence the user often relies on the off-the-shelf components that are available in the open market. These parts often do not have a guaranteed supply-chain history and pose a high risk of being counterfeit. Counterfeit chips that are recycled, remarked, cloned or defective pose a significant threat, as they can compromise critical CPS infrastructures (transportation, military, health, etc.). Detection mechanisms for counterfeit ICs usually involve the identification of the defects produced by counterfeiting. This sometimes requires time-consuming and sophisticated physical inspection processes. In the case of recycled ICs, embedded sensors can detect prior usage of ICs by measuring device aging [59], [71]. However, old parts that are already in the system may not have such embedded mechanisms. This still poses vulnerabilities and requires complex detection scheme [72]. Researchers have tried to ensure security, integrity, and data confidentiality for some legacy systems as well, such as legacy SCADA system [73], and IEEE P1711 standard for legacy serial links [74], however these are not enough to eliminate all the threats for large scale CPS. Since counterfeit ICs pose different levels of threats, a reliable fault-tolerant scheme needs to be adopted along with proper counterfeit detection and avoidance schemes to minimize the risk factors mentioned in Section 5.1.4 to its best.

It should be noted that not all the threats can be eliminated via hardware security primitives alone, since the threats are distributed in both physical and cyber domain of CPS. Further, threats and vulnerabilities that exist in different level if abstraction pose different challenges for securing CPS and require different approaches for eliminating them.

Application of above discussed defense techniques are hence versatile although much generalized. A more specific threat and attack model and with possible defense scenario requires a more elaborated and in-depth analysis of CPS abstractions with application specific security protocols (in cyber or software domain) and hardware interaction (in physical domain) among different layers. For a more elaborated discussion on the CPS design challenges and vulnerabilities from the hardware security perspective, we refer the readers to [51], [75].

6 Conclusion

In this paper, we introduced the security concerns in modern cyber-physical systems from a cross-layer perspective. Security vulnerabilities and possible consequence as well as countermeasures were introduced on the system-, device- and hardware-levels. Through this introductory paper, we try to provide researchers who are interested in this area a full map of the current challenges and state-of-the-art solutions. As we mentioned in this paper, the existing solutions are far from enough to secure the future CPS which are being widely used in national critical infrastructures. Therefore, the paper also discussed the research directions in this area as a guideline for future research.

Acknowledgments

J. Wurm and Y. Jin are supported in part by the Florida Cybersecurity Center (FC2) 2015 Collaborative Seed Grant Program and the Southeastern Center for Electrical Engineering Education (SCEEE 15-001). Y. Liu and S. Hu are supported in part by US National Science Foundation CAREER Award CCF-1349984.

References

[1] K.-D. Kim and P. Kumar, "Cyber-physical systems: A perspective at the centennial," *Proc. IEEE*, vol. 100, no. Special Centennial Issue, pp. 1287–1308, May 2012.

[2] R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: The next computing revolution," in *Proc. 47th Design Autom. Conf.*, 2010, pp. 731–736.

[3] C. Konstantinou, M. Maniatakos, F. Saqib, S. Hu, J. Plusquellic, and Y. Jin, "Cyber-physical systems: A security perspective," in *Proc. 20th IEEE Eur. Test Symp.*, 2015, pp. 1–8.

[4] D. DiMase, Z. A. Collier, K. Heffner, and I. Linkov, "Systems engineering framework for cyber physica l security and resilience," *Environ. Syst. Decisions*, vol. 35, no. 2, pp. 291–300, 2015.

[5] G. Wallace, "Target and neiman marcus hacks: The latest," CNN Money, (2014). [Online]. http://money.cnn.com/2014/01/13/news/target-neiman-marcushack/

[6] J. Finkle, "Target says criminals attacked with credentials stolen from vendor," *Reuters*, 2014. [Online]. Available: http://www.reuters.com/article/us-target-cyberattack-idUSBREA0S25Z20140129

[7] Ponemon Institute, "2013 cost of data breach study: Global analysis," *Ponemon Inst., Res. Rep.*, 2013. [Online]. Available: http://www.ponemon.org/local/upload/file/2013%20Report%20GLOBAL%20CODB%20FINAL%205-2.pdf

[8] L. Grossman, "World war zero: How hackers fight to steal your secrets," *Time Mag.*, Jun. 2014.

[9] Identity Theft Resource Center, "Identity theft resource center breach report hits record high in 2014," 2014. [Online]. Available: http://www.idtheftcenter.org/ITRC-Surveys-Studies/2014 databreaches.html

[10] Y. Liu, S. Hu, H. Huang, R. Ranjan, A. Zomaya, and L. Wang, "Game theoretic market driven smart home scheduling considering energy balancing," *IEEE Syst. J.*, vol. 11, no. 2, pp. 910–921, Jun. 2017.

[11] Apple, "HomeKit," [Online]. Available: https://developer.apple.com/homekit/

[12] NEST, "NEST Smart Home," [Online]. Available: https://nest.com/

[13] Ameren Illinois Cop., "Real Time Price," [Online]. Available: https://www2.ameren.com/RetailEnergy/RealTimePrices

[14] X. Chen, T. Wei, and S. Hu, "Uncertainty-aware household appliance scheduling considering dynamic electricity pricing in smart home," *IEEE Trans. Smart Grid*, vol. 4, no. 2, pp. 932–941, Jun. 2013.

[15] L. Liu, Y. Liu, A. Zomaya, L. Wang, and S. Hu, "Economical and balanced energy usage in the smart home infrastructure: A tutorial and new results," *IEEE Trans. Emerging Topics Comput.*, vol. 3, no. 4, pp. 556–570, Dec. 2015.

[16] G. Hernandez, O. Arias, D. Buentello, and Y. Jin, "Smart Nest Thermostat: A smart spy in your home," *Black Hat USA*, 2014.

[17] C. Heres, A. Etemadieh, M. Baker, and H. Nielsen, "Hack all the things: 20 devices in 45 minutes," in *DEFCON*, 2014.

[18] (2013). [Online]. Available: http://www.cnn.com/2013/08/02/tech/innovation/hackable-homes/

[19] Texas Instruments, "Smart e-meter: AMR/AMI." [Online]. (2016). Available: http://www.ti.com/solution/docs/appsolution.tsp?appId=407

[20] Y. Liu, S. Hu, and T.-Y. Ho, "Vulnerability assessment and defense technology for smart home cybersecurity considering pricing cyberattacks," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2014, pp. 183–190.

[21] C. Liao, C.-W. Ten, and S. Hu, "Strategic frtu deployment considering cybersecurity in secondary distribution network," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1264–1274, Sep. 2013.

[22] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891 Dec. 1992.

[23] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," in *Proc. AAAI Conf. Artif. Intell.*, 2012, pp. 356–362.

[24] W. Yang, N. Li, Y. Qi, W. Qardaji, S. McLaughlin, and P. McDaniel, "Minimizing private data disclosures in the smart grid," in *Proc. ACM Conf. Comput. Commun. Security*, 2012, pp. 415–427.

[25] C. Konstantinou, M. Maniatakos, F. Saqib, S. Hu, J. Plusquellic, and Y. Jin, "Cyber-physical systems: A security perspective," in *Proc. IEEE Eur. Test Symp.*, 2015. pp. 1–8.

[26] Y. Jin and D. Oliveira, "Extended abstract: Trustworthy SoC architecture with on-demand security policies and HW-SW cooperation," in *Proc. 5th Workshop SoCs, Heterogeneous Architectures Workloads*, 2014.

[27] D. Oliveira, J. Navarro, N. Wetzel, and M. Bucci, "Ianus: Secure and holistic coexistence with kernel extensions - A immune system-inspired approach," in *Proc. 29th Annu. ACM Symp. Appl. Comput.*, 2014, pp. 1672–1679.

[28] Y. Liu, S. Hu, and T.-Y. Ho, "Leveraging strategic defense algorithms for smart home pricing cyberattacks," *IEEE Trans. Dependable Secure Comput.*, vol. 13, no. 2, pp. 220–235, Mar./Apr. 2016.

[29] Y. Liu, S. Hu, J. Wu, Y. Shi, Y. Jin, Y. Hu, and X. Li, "Impact assessment of net metering on smart home cyberattack detection," in *Proc. 52nd Annu. IEEE/ACM Design Autom. Conf.*, 2015, Art. no. 97.

[30] S. T. C. Konigsmark, L. K. Hwang, D. Chen, and M. D. F. Wong, "CNPUF: A carbon nanotube-based physically unclonable function for secure low-energy hardware design," in *Proc. 19th Asia South Pacific Design Autom. Conf.*, 2014, pp. 73–78.

[31] Y. Liu, L. Liu, Y. Zhou, and S. Hu, "Leveraging carbon nanotube technologies in developing physically unclonable function for cyber-physical system authentication," in *Proc. IEEE INFOCOM Cyber-Physical Syst. Authentication Workshop*, 2016.

[32] J. Wurm, O. Arias, K. Hoang, A.-R. Sadeght, and Y. Jin, "Security analysis on consumer and industrial iot devices," in *Proc. 21st Asia South Pacific Design Autom. Conf.*, 2016, pp. 519–524.

[33] O. Arias, J. Wurm, K. Hoang, and Y. Jin, "Privacy and security in internet of things and wearable devices," *IEEE Trans. Multi-Scale Comput. Syst.*, vol. 1, no. 2, pp. 99–109, Apr.-Jun. 2015.

[34] "Apple iphone bootloader attack," 2008. [Online]. Available: http://rdist.root.org/2008/03/17/apple-iphone-bootloader-attack/

[35] B. Parno, J. M. McCune, and A. Perrig, "Bootstrapping trust in commodity computers," in *Proc. IEEE Symp. Security Privacy*, 2010, pp. 414–429.

[36] A. Cui, J. Kataria, and S. J. Stofo, "From prey to hunter: Transforming legacy embedded devices into exploitation sensor grids," in *Proc. 27th Annu. Comput. Security Appl. Conf.*, 2011, pp. 393–402.

[37] "Xbox 360 timing attack," 2007. [Online]. Available: http://beta.ivc.no/wiki/index.php/Xbox_360_Timing_Attack

[38] D. Brumley and D. Boneh, "Remote timing attacks are practical," *Comput. Netw.: Int. J. Comput. Telecommun. Netw. - Web Security*, vol. 48, no. 5, pp. 701–716, 2005.

[39] S. Skorobogatov, "Fault attacks on secure chips: From glitch to flash," in *Proc. Design Security Cryptographic Algorithms Devices*, 2011.

[40] Bushing, Marcan, Segher, and Sven, "Console hacking 2010: Ps3 epic fail," in *Proc. 27th Chaos Commun. Congr.*, 2010.

[41] R. Lemos, "Sony left passwords, code-signing keys virtually unprotected," *eWeek*, 2014. [Online]. Available: http://www.eweek.com/security/sony-left-passwords-code-signing-keys-virtuall y-unprotected.html

[42] B. Schneier, "Cryptographic design vulnerabilities," *Computer*, vol. 31, no. 9, pp. 29–33, Sep. 1998.

[43] B. Fowler, "Some top baby monitors lack basic security features, report finds," 2015. [Online]. Available: http://www.nbcnewyork.com/news/local/Baby-Monitor-Security-Research-324169831.html

[44] M. Smith, "Security holes in the 3 most popular smart home hubs and honeywell tuxedo touch," 2015. [Online]. Available: http://www.networkworld.com/article/2952718/microsoft-subnet/security-holes- in-the-3-most-popular-smart-home-hubs-and-honeywell-tuxedo-touch.html

[45] "Critical security flaw: Glibc stack-based buffer overflow in getaddrinfo() (cve-2015-7547)," 2015. [Online]. Available: https://access.redhat.com/articles/2161461

[46] C. Cowan et al., "Stackguard: Automatic adaptive detection and prevention of buffer-overflow attacks." in *Proc. 7th Conf. Usenix Security Symp.*, 1998, pp. 63–78.

[47] C. Cowan, S. Beattie, J. Johansen, and P. Wagle, "Pointguard: protecting pointers from buffer overflow vulnerabilities," in *Proc. 12th Conf. Usenix Security Symp.*, 2003, pp. 91–104.

[48] O. Arias, J. Wurm, K. Hoang, and Y. Jin, "Privacy and security in internet of things and wearable devices," *IEEE Trans. Multi-Scale Comput. Syst.*, vol. 1, no. 2, pp. 99–109, Apr.–Jun. 2015.

[49] Texas Instruments, "AM3715, AM3703 Sitara ARM Microprocessor," (2011). [Online]. Available: http://www.ti.com/lit/ds/symlink/am3715.pdf

[50] P. Antsaklis, "Goals and challenges in cyber-physical systems research editorial of the editor in chief," *IEEE Trans. Autom. Control*, vol. 59, no. 12, pp. 3117–3119, Dec. 2014.

[51] S. Khaitan and J. McCalley, "Design techniques and applications of cyberphysical systems: A survey," *IEEE Syst. J.*, vol. 9, no. 2, pp. 350–365, Jun. 2015.

[52] J. Shi, J. Wan, H. Yan, and H. Suo, "A survey of cyber-physical systems," in *Proc. IEEE Int. Conf. Wireless Commun. Signal Process.*, 2011, pp. 1–6.

[53] A. Cardenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. Sastry, "Challenges for securing cyber physical systems," in *Proc. Workshop Future Directions Cyber-Phys. Syst. Security*, 2009.

[54] H. Bar-El, "Known attacks against smartcards," Discretix Technologies Ltd., White Paper. 2005.

[55] S. P. Skorobogatov, "Semi-invasive attacks: A new approach to hardware security analysis," Ph.D. dissertation, Citeseer, NEC Res. Inst. (now NEC Labs), Princeton, NJ, USA, 2005.

[56] F. Demaertelaere, "Hardware security modules," 2010, [Online]. Available: https://handouts.secappdev.org/handouts/2010/Filip

[57] G. P. Hancke, "A practical relay attack on ISO 14443 proximity cards," Univ. Cambridge Comput. Laboratory, Cambridge, U.K., Tech. Rep., vol. 59, pp. 382–385, 2005.

[58] B. Krebs, "Cyber incident blamed for nuclear power plant shutdown," *Washington Post*, 2008. [Online]. Available: http://www.washingtonpost.com/wp-dyn/content/article/2008/06/05/AR2008060501958.html

[59] M. M. Tehranipoor, U. Guin, and D. Forte, *Counterfeit Integrated Circuits: Detection Avoidance*. Berlin, Germany: Springer, 2015.

[60] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Silicon physical random functions," in *Proc. 9th ACM Conf. Comput. Commun. Security*, 2002, pp. 148–160.

[61] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. 44th Annu. Design Autom. Conf.*, 2007, pp. 9–14.

[62] M. Rahman, F. Rahman, D. Forte, and M. Tehranipoor, "An aging-resistant ro-puf for reliable key generation," *IEEE Trans. Emerging Topics Comput.*, vol. 3, no. 3, pp. 335–348, Jul.-Sep. 2016.

[63] O. Al Ibrahim and S. Nair, "Cyber-physical security using system-level pufs," in *Proc. 7th Int. Wireless Commun. Mobile Comput. Conf.*, 2011, pp. 1672–1676.

[64] L. Wei, C. Song, Y. Liu, J. Zhang, F. Yuan, and Q. Xu, "Boardpuf: Physical unclonable functions for printed circuit board authentication," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2015, pp. 152–158.

[65] B. Sunar, W. J. Martin, and D. R. Stinson, "A provably secure true random number generator with built-in tolerance to active attacks," *IEEE Trans. Comput.*, vol. 56, no. 1, pp. 109–119, Jan. 2007.

[66] M. Stipčević and Ç. K. Koç, "True random number generators," in *Open Problems in Mathematics and Computational Science*. Berlin, Germany: Springer, 2014, pp. 275–315.

[67] M. T. Rahman, K. Xiao, D. Forte, X. Zhang, J. Shi, and M. Tehranipoor, "TI-TRNG: Technology independent true random number generator," in *Proc. 51st Annu. Design Autom. Conf.*, 2014, pp. 1–6.

[68] T. E-security, "Tamper-resistant security: Today's challenge," [Online]. Available: https://www.thales-esecurity.com/solutions/by-technology-focus/tamper-resistant-security

[69] S. Guilley, L. Sauvage, J.-L. Danger, N. Selmane, and R. Pacalet, "Silicon-level solutions to counteract passive and active attacks," in *Proc. IEEE 5th Workshop Fault Diagnosis Tolerance Cryptography*, 2008, pp. 3–17.

[70] D. Shahrjerdi, J. Rajendran, S. Garg, F. Koushanfar, and R. Karri, "Shielding and securing integrated circuits with sensors," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2014, pp. 170–174.

[71] X. Zhang, N. Tuzzio, and M. Tehranipoor, "Identification of recovered ICs using fingerprints from a light-weight on-chip sensor," in *Proc. 49th Annu. Design Autom. Conf.*, 2012, pp. 703–708.

[72] H. Dogan, D. Forte, and M. M. Tehranipoor, "Aging analysis for recycled FPGA detection," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Nanotechnology Syst.*, 2014, pp. 171–176.

[73] P. P. Tsang and S. W. Smith, "YASIR: A low-latency, high-integrity security retrofit for legacy SCADA systems," in *Proc. IFIP TC 11 23rd Int. Inform. Security Conf.*, 2008, pp. 445–459.

[74] S. Hurd, R. Smith, and G. Leischner, "Tutorial: Security in electric utility control systems," in *Proc. IEEE 61st Annu. Conf. Protective Relay Eng.*, 2008, pp. 304–309.

[75] CSRA, "Designed-in cyber security for cyber-physical systems," in *Proc. CSRA -NIST Alliance Workshop*, 2013, pp. 1–60. [Online]. Available: http://www.cybersecurityresearch.org/documents/CSRA_Workshop_Report.pdf

Jacob Wurm is currently a senior undergraduate student studying computer engineering at the University of Central Florida. He is currently a research assistant in the Security in Silicon Laboratory lead by Dr. Yier Jin. His research interests include, embedded device security, secure communication protocols, and network traffic analysis.

Yier Jin received the BS and MS degrees in electrical engineering from Zhejiang University, China, in 2005 and 2007, respectively, and the PhD degree in electrical engineering from Yale University in 2012. He is currently an assistant professor in the EECS Department at the University of Central Florida. His research focuses on the areas of trusted embedded systems, trusted hardware intellectual property (IP) cores and hardware-software co-protection on computer systems. He proposed various approaches in the area of hardware security, including the hardware Trojan detection methodology relying on local side-channel information, the post-deployment hardware trust assessment framework, and the proof-carrying hardware IP protection scheme. He is also interested in the security analysis on Internet of Things (IoT) and wearable devices with particular emphasis on information integrity and privacy protection in the IoT era. He is the Best Paper Award Recipient of DAC'15 and ASP-DAC'16.

Yang Liu received the BS degree in telecommunications engineering, Huazhong University of Science and Technology, Wuhan, China, in 2011. He is currently working toward the PhD degree in electrical engineering, Michigan Technological University, Houghton, MI. His research focuses on smart home system, cyber-physical systems, and big data analytics. He was a visiting student at Carnegie Mellon University, Pittsburgh, PA, in Fall 2015.

Shiyan Hu received the PhD degree in computer engineering from Texas A&M University in 2008. He is an associate professor at Michigan Technological University where he is director of the Center for Cyber-Physical Systems and associate director of the Institute of Computer and Cybersystems. He was a visiting professor at IBM Research (Austin) in 2010, and a visiting associate professor at Stanford University from 2015 to 2016. His research interests include cyber-physical systems, cybersecurity, computer-aided design of VLSI circuits, and embedded systems, where he has published more than 100 refereed papers. He is an ACM distinguished speaker, an IEEE Computer Society distinguished visitor, an invited participant for U.S. National Academy of Engineering Frontiers of Engineering Symposium, a recipient of a US National Science Foundation (NSF) CAREER Award, a recipient of the ACM SIGDA Richard Newton DAC Scholarship (as the faculty advisor), and a recipient of the JSPS Faculty Invitation Fellowship. He is the chair for the IEEE Technical Committee on Cyber-Physical Systems. He serves as an associate editor for the *IEEE Transactions on Computer-Aided Design*, *IEEE Transactions on Industrial Informatics*, and *IEEE Transactions on Circuits and Systems*. He is also a guest editor for seven IEEE/ACM Transactions such as the *IEEE Transactions on Computers* and *IEEE Transactions on Computer-Aided Design*. He has served as conference chairs, track chairs, and TPC members for more than 70 times. He is a senior member of the IEEE.

Kenneth Heffner received the PhD degree in chemistry from the University of South Florida, Tampa, FL. He is currently an engineering fellow for Honeywell Aerospace in Clearwater, FL, supporting Honeywell's Aerospace business units. He is the technology leader for Honeywell's new Systems Security Engineering business unit. His research includes sensors for inertial navigation systems, autonomous thin film instrumental analysis, high-density vertically-integrated microsystems, high-performance computing, and embedded secure microelectronics systems. He holds 16 US patents. He is also a certified Design for Six Sigma Black Belt for hardware design.

Fahim Rahman received the BSc degree in electrical and electronic engineering from the Bangladesh University of Engineering and Technology, Bangladesh, in 2009 and the MS degree in electrical and computer engineering from the University of Connecticut in 2015. He is currently working toward the PhD degree in electrical and computer engineering at the University of Florida. He is an active researcher at the Florida Institute of Cyber-Security (FICS), with contributions and interests in the field of hardware security and trust. His specialties include design of low-cost hardware security primitives for cyber-physical systems and internet of things, and evaluate security aspects of emerging nano-electronic devices with potential cryptographic and trusted supply chain applications.

Mark Tehranipoor received the PhD degree from the University of Texas at Dallas in 2004. He is currently the Intel Charles E. Young Preeminence endowed professor in cybersecurity at the University of Florida (UF). His current research projects include: hardware security and trust, supply chain security, VLSI design, test, and reliability. He has published more than 300 journal articles and refereed conference papers and has given more than 150 invited talks and keynote addresses. He has published 6 books and 11 book chapters. He received several best paper awards as well as the 2008 IEEE Computer Society (CS) Meritorious Service Award, the 2012 IEEE CS Outstanding Contribution, the 2009 US NSF CAREER Award, and the 2014 MURI award. He serves on the program committee of more than a dozen leading conferences and workshops. He served as program chair of the 2007 IEEE Defect-Based Testing (DBT) workshop, program chair of the 2008 IEEE Defect and Data Driven Testing (D3T) workshop, co-program chair of the 2008 International Symposium on Defect and Fault Tolerance in VLSI Systems (DFTS), general chair for D3T-2009 and DFTS-2009, and vice-general chair for NATW-2011. He co-founded the IEEE International Symposium on Hardware-Oriented Security and Trust (HOST) and served as HOST-2008 and HOST-2009 general chair. He is currently serving as an associate editor for *JETTA*, *JOLPE*, *IEEE TVLSI*, and *ACM TODAES*. Prior to joining UF, he served as the founding director for CHASE and CSI centers at the University of Connecticut. He is currently serving as co-director for the Florida Institute for Cybersecurity Research (FICS). He is a senior member of the IEEE, a Golden Core Member of IEEE, and member of the ACM and ACM SIGDA.