Conference Paper

Survey of Hardware-based Security support for IoT/CPS Systems

Alauddin Al-Omary[1], Ali Othman[2], Haider M. AlSabbagh[3], and Hussain Al-Rizzo[4]

[1]Department of Computer Engineering, College of IT, University of Bahrain
[2]Department of Communication Engineering, University of Nineveh, Mosul, Iraq
[3]Department of Electrical Engineering, College of Engineering, University of Basra, Basra, Iraq
[4]Department of Systems Engineering, College of Engineering and Information Technology, University of Arkansas at Little Rock, USA

Abstract

The growth of Internet of Things (IoT) and Cyber-Physical Systems (CPS) has considerably increased the customer accessibility, convenience and boosted the industrial productivity. However, the increased use of IoT/CPS systems raises new security challenges. Due to the nature of IoT/CPS systems that heavily depends on connected low computation power devices equipped with sensors, the security characteristics and needs of these systems differ from the security of traditional software-based security applied in conventional network devices. To secure the IoT/CPS systems, a hardware security support is needed as software-based security is inadequate to protect such systems against cyber-attacks. Recent Field Programmable Gate Array (FPGA) and System on Chips (SoCs) can help in implementing a security system that extends to the IC level. FPGA SoC helps bringing complete range of scalable security and at the same time sustain the low-power system operation. In this article, a survey of hardware-based security support is conducted and introduced. Concentrating on hardware security will help users to have better insight about IoT/CPS security requirements, identify the vulnerabilities of these systems and give good information on how to build secure IoT/CPS systems.

Keywords: Internet of Things (IoT), Cyber security, Cyber-Physical Systems (CPS), Hardware Root of Trust (HRoT), FPGA, SoC.

Corresponding Author:
Alauddin Al-Omary
aalomary@uob.edu.bh

Received: 18 September 2018
Accepted: 10 October 2018
Published: 15 October 2018

Publishing services provided by Knowledge E

Selection and Peer-review under the responsibility of the Sustainability and Resilience Conference Committee.

1. Introduction

The Internet of things (IoT) is the internetworking of physical devices in home, buildings, streets and other places that enable these objects to collect and exchange data [1]. Example of IoT systems include devices embedded in smart home, smart building and smart cities, etc. Cyber-physical systems (CPS) is a new generation of systems that has some similarities with IoT. CPS has integrated computational and physical

capabilities that can interact with humans [2]. A CPS is controlled or supervised by algorithms, which is tightly integrated with the Internet and its users [3]. Examples of CPS include electrical smart grid, autonomous automobile systems, medical monitoring, process control systems in factories, robotics systems, and automatic pilot avionics [3]. By 2020, experts estimate that, the IoT/CPS system will have about 30 billion objects [4] and the global market value of these systems is estimated to reach $7.1 trillion [5]. With this huge deployment many challenges are facing IoT/CPS systems in design, deployment, standardization, architecture modeling and security. The security issue is one of the concern that is facing IoT/CPS systems. Recently there have been a lot of attacks on IoT/CPS systems. Some attacks were done by login to house security camera using default password and spy on the people's houses [6]. There is also attacks on factories. In 2014, it was reported that a cyber-attack on German steel mill control system. The attack prevented the orderly shutdown of the system [7] causing a massive damage to it. Ukrainian power grid was also attacked in December 2015 [8]. The attackers were able to access the power grid and take down the power plant by physically destroying a generator and this cause many losses and cut the cut the power to over 200,000 people. The attackers could do this damage with just 21 lines of code and they were able to not only shut down the breakers, but also wipe hard disks and even flash the firmware on some of the key systems that were needed to bring it back up. This was the first incident of cyber-attack on a power grid and was done after adding the IoT connectivity to the power grid. IoT/CPS systems are different from computers connected to network and internet in that IoT/CPS systems tend to have a long deployment and lifetime usage. They rarely subject to management. This is different with computer systems, where your PC at least are checked, and updates are installed in it when available. For IoT/CPS systems, whether it is a smart grid control system, security camera or an industrial robot arm, they need to be up and running for years at a time, with infrequent chances for downtime. The top main concern for IoT/CPS systems are reliability, safety, efficiency and productivity. For these reasons, depending only on software security is not enough and the only reliable way to maintain security is to depend on hardware security. The hardware security is designed to resist attacks over long periods as most of the attack used tools developed for Windows, UNIX, and Linux and cannot used to break into the security chip. IoT/CPS uses very specialized operating system, a specialized environment that is built into hardware ICs designed with security as a top priority. The use of security chips gives a trusted environment that can be used for what's called a 'hardware root of trust', kind of a strong base

onto which a good secure system can be built. When we have such a strong hardware based IoT/CPS system we don't need to update it regularly and everything on top of it can be stable and secure. In this paper, we are investigating the various hardware-based security measures that are used to secure IoT/CPS to help users to identify the vulnerabilities of IoT/CPS systems and to aid on how to build hardware-based security into these systems. Section 2 will present IoT/CPS security needs, section 3 presents the IoT/CPS structures, section four give an overview about IoT/CPS standardization, and finally section five presents IoT/CPS hardware-based Security.

2. IoT/CPS Security Needs

IoT/CPS needed security features can be summarized as follows:

1. Authentication - For any IoT implementation, device should have authenticated to the network.
2. Privacy – To be sure that the transmission of any data occurs only between approved systems and devices. For instance, a security camera should send information to the smartphone of its owner and vice versa.
3. Confidentiality and Intellectual property protection – some of the sent data to "things" is proprietary that should be protected. For instance, if in a given factory there is a machine that controls the recipe for some food products, your IoT system should only send the recipe to your production machine and not to other machine.
4. Integrity – verify that message is unchanged or tampered.
5. Software updates – if it is needed to install a security update or software patch, a way should be found to check that it was installed in the right device.
6. Reliability – Since the IoT/CPS is deployed and supposed to work for long time and since several CPSs have 24x7 availability requirement, upgrading them or correcting their faults is very challenging. Therefore, it is necessary to include reactive and predictive maintenance in these systems. Reactive maintenance makes classifying and fixing faulty device easy by the help of IoT/CPS monitoring system that sends notification about any problem found. The predictive maintenance require that IoT/CPS system should continuously collect accurate data that enable proactive remedy to the system.

7. Traffic security: IoT traffic has unique security and has different requirements for availability, reliability and bandwidth. Most IoT data is highly latency-sensitive and has high availability requirements. It also need relatively low bandwidth compared with conventional back-office applications and traffic and should not run over a general-purpose network and needs to be handled separately (if possible) using Network segmentation.

8. End-to-end encryption- End-t-end network encryption is needed since most IoT networks are wireless and can be intervened easily.

9. Cloud storage protection: In many cases, IoT data is directed to cloud-based applications or data collection engines. Cloud-based protection and security in this case might be enough and a way of how to protect the data from the sensor through to the cloud aggregation point, and from there on to the applications is needed.

3. IoT/CPS Structure

There are different IoT/CPS platforms introduced by many researchers [20-27] and this makes the operation of finding solution to the IoT/CPS security problems more complicated. Therefore, it is essential to know the foundation and the elements of the IoT. Three factors are attributing the IoT environment [28] and [29]:

1. Internet-oriented

2. Things-oriented

3. Semantic-oriented

We can apply the same model to the IoT/CPS structure. The IoT/CPS architecture, according to K. Zhao [16] is made up of three layers:

1. The perception layer

2. The network layer

3. The application layer

3.1. Application layer

This layer is the topmost layer of the IoT system that the end user can see. Different applications, like smart electrical grid, smart city, smart healthcare system, and intelligent transportation protocols can be found in this layer [30]. Currently, there is no agreed upon standard for making the IoT application layer [16] and this layer can be organized in several ways based on the service it offers. Application layer mostly includes a middleware, a service support platform, a machine-to-machine (M2M) communication protocol and cloud computing [31]. The security issues vary depending on the business, industry, situation and environment [32].

3.2. Network layer

This layer offers transmission and information security. The network layer includes mobile devices, wireless networks (Bluetooth, WIFI, Zigbee, 6LoWPAN), cloud computing, and the Internet [33].

3.3. Perception layer

This layer is responsible for information collection and is grouped into two layers, the perception node layer (sensors, controllers, and so on) and the perception network layer [34]. In the perception node, data are acquired and controlled. In the perception network, layer control instructions for sending and controlling data are carried out.

4. IoT/CPS Standardization and Security

Cooperation, standardization and regulation are significant to certify an end-to-end secure solution in the IoT/CPS Industry. As the volumes of connected devices increases, the threats to privacy, safety, and information security also increase. Because there is no agreed upon framework for the IoT, IEEE and ETSI, have released technology-specific standards for IoT including security guides [16]. This leads to other initiatives for unified architecture and modeling like the Reference Architecture Model Industry 4.0 (RAMI 4.0) [36], the Industrial Internet Reference Architecture (IIRA) [37] and the Internet of Things - Architecture (IoT-A) [38]. In addition to the industry, the scientific community is an important contributor to the standardization of IoT protocols and technology as

well [29], [50], [41], [42]. In 2013, IETF also contributed to IoT security by issuing an Internet-Draft draft-garcia-core-security-06 [35].

5. IoT Security Vs. Wireless Security

In dealing with security and privacy several significant differences exist between the IoT/CPS systems and traditional wireless networks. These differences can be summarized as follows:

The IoT devices exists on low power and lossy networks (LLNs), whereas others have very dynamic topologies that count on the application. LLNs is less secure due to node impersonation. If an attacker can connect to any IoT device in the network, the attacker is considered as trusted node.

The security features and requirements of IoT is also different. In the IoT perception layer, sensor nodes uses microcontrollers, which have limited computational power and low storage capacity therefore, using the public key encryption to secure the IoT devices impossible and instead a Lightweight encryption technology is used for the IoT devices.

In the network layer, security issues, such as man-in-the-middle and counterfeit attacks can be found. These attacks come from sending false information to communicating nodes in the network [16]. Mechanism for identifying authentication and data confidentiality must be used to prevent unauthorized access.

Application layer security requirement is also different which need protection of user privacy across heterogeneous networks.

There is also difference in the communication protocols used by both networks. For example, IoT in the perception layer uses IPv6 over low-power wireless personal area networks (6LoWPAN) which combines IPv6 and Low-power Wireless Personal Area Networks. In conventional networks WIFI is used. In the IoT network layer, the communication protocol used is Datagram Transport Layer (DTLS), whereas wireless network uses TCP protocol. In IoT application layer Constrained Application Protocol (CoAP) is used for communication, whereas the application layer of conventional networks uses HTTP protocol.

In brief, the security of conventional wireless network is designed based on the viewpoint of users and not applicable for IoT/CPS Machine- to-Machine (M2M) communication. The security issue in both networks may be same but handling each network security issue need different approaches and techniques.

6. IoT/CPS Hardware-based Security

Modern applications, like IoT and CPS require reliable security. The United States Department of Homeland Security (DHS) in November 2016 listed six principles to address IOT security challenges (https://www.dhs.gov/sites/default/files/publications/Strategic_Principles_for_Securing_the_Internet_of_Things-2016-1115-FINAL_v2-dg11.pdf):

Security should be Included in the early design phase.

1. Security updates and vulnerability management must be applied
2. Proven security practices should be incorporated.
3. Give priority of security measure that give potential impact on your system
4. Disseminate transparency across IoT
5. Take care of IoT connectivity (carefully and deliberately).

Any user-accessible device including IoT and CPS devices is subjected to intellectual property (IP) robbery and reverse-engineering of the product. IP protection is a significant key manufacturer to safeguard their product. An integrated system solution based on secured hardware is needed which enables software to be stored, run and updated in a protected way. Several efforts have been made earlier to employ purely Software-based solutions for device authentication. Unluckily, software has numerous vulnerabilities; it can be read, analyzed and altered by attacker since is written code. Attacker can re-program the device with modified software and thus the authentication process and system integrity can be broken. Other weakness of software-based solutions can be the improper storage of secret keys. Normally, attackers can identify secret keys from software easily as keys normally act like random numbers when compared to the program code itself. There exit many entropy analyzers programs that can scan the software and find parts with high randomness. The randomness parts typically contain the keys. Entropy analysis scan is very fast, and the keys found can be used to generate counterfeit account to attack the system. However, hardware can be used to protect software. Secured hardware can be used to protect the managing and storage of code using encryption. It can also be used for fault and manipulation detection. Software becomes reliable by blending it with secured hardware. Many measures are used to implement hardware support security. These measures can be used at booting stage, operation stage or maintenance and updating stage.

Following is some of the hardware security techniques used with IoT/CPS systems:

6.1. Defense- in-depth

This approach uses multi security layers to achieve more immunity against cyber-attack. Some of the layers may be implemented in hardware to enhance the software security.

6.2. Trusted execution environment (TEE)

TEE is a secure area of the microcontroller or SoC that assures code and data protection of applications executed with it. It provides a higher level of security than the one provided by some Operating system like (mobile OS) and more functionality than a 'secure element' (SE). It guarantees code and data loaded inside to be protected with respect to confidentiality and integrity. Although, hardware TEEs can protect their code and data even when powered off and offer safeguards at boot time, it is suitable for a relatively low security risk system with low-end devices, like certain wearables but not for sensitive systems with high security risk like IoT based health-care, auto-driving etc. Many Industry associations like GlobalPlatform and Trusted Computing Group have carry out work in recent years for the standardization of TEE.

6.3. Trusted platform module (TPM)

A TPM is chip that is embedded into a computing device to provide hardware-based security. The phrase TPM is sometimes used to refer to the set of specifications applicable to TPM chips. The TMP chip is used to provide hardware authentication on an endpoint device by storing RSA encryption keys specific to that system. It holds an RSA key pair called the Endorsement Key (EK). EK is very secure since it is stored inside the chip and cannot be accessed by software. When the administrator takes ownership of the system, a Storage Root Key (SRK) is generated by the TPM based on the EK and an owner-specified password. TMP also prevent unauthorized firmware modification by generating second key, called an Attestation Identity Key (AIK). AIK protects protect firmware by fragmenting critical sections of the firmware and software before they are executed. When any system attempts to modify the firmware, the fragmented components are sent to a server for verification. If any of the fragmented components

has been modified there will be no match and the system cannot modify the firmware. Thus, TMP can be used for trusted and secure boot. TPM has gained support of IT industry and was initially founded by IBM and then formalized under the Trusted Computing Group (TCG) and is ideal for IoT/CPS applications.

6.4. Trusted network connection (TNC)

TNC is open, standards created by the Trusted Computing Group (TCG). It performs endpoints authentication by inspecting them to ensure that they complied with security policies before connecting them on the protected network. This ensures that all endpoints are prepared to defend against attacks from unauthorized devices. In this way, information sharing is well protected. In TNC environment, we can block unauthorized users, devices and we can permit appropriate levels of access to authorized users/devices. We can also isolate and repair damaging or vulnerable devices and share real-time information about users, devices, threats etc.

6.5. Hardware root of trust (HRoT)

HRoT is used to ensure that remote connected devices used in IoT applications is connected securely and is working in a reliable way. The use of a standalone security processor or co-processor that perform as an HRoT has been well established in the IT industries such as personal computers, servers, chip cards and identity documents. HRoT is ideal for IoT/CPS and other industrial applications and can be used in a Trusted Platform Module (TPM) in conjunction with other security elements to provide IoT/CPS devices with complete security functions such as integrated crypto-processors, encrypted storage, buses and peripheral functions protection as well as integrated error detection and intrusion detection. In this way, network end points can be powerfully protected using this hardware-based approach. Many VLSI venders are building HRoT chips for a typical IoT device with that has constraint of processing power, size, energy consumption. Venders like Synopsys, Intel, AMD, Microsemi have developed security chip or FPGA SoC solutions to provides security support on behalf of client applications running on host CPU(s). HRoT can be used to prevent attack codes such as rootkits and bootkits (https://nostarch.com/rootkits) attacks.

6.6. Chain of trust (CoT)

COT is can be viewed as an extension of RHoT in that the digital certificates are verified using a chain of trusting done by root (anchor) certificate authority (CA). The certificate hierarchy is a structure of certificates that permits individuals to validate the legitimacy of a certificate's issuer. The Unified Extensible Firmware Interface (http://www.uefi.org/) (UEFI) Forum is leading the call to use CoT to secure booting. The trust is preserved via public key cryptography using Platform Key (PK) stored into the firmware. PK represents the Root of Trust. Security is established by requiring that no code will be executed by firmware unless it has been signed by a "trusted" key. For third part certification, a Key Exchange Keys (KEK) can be added to the UEFI key database and the third party is certified if they are signed with the private part of the PK. A centralized Certificate Authority (CA) is used to manage the signing process which is currently operated by Microsoft. As for the case of HRoT, CoT can also be used to prevent attack codes such as rootkits and bootkits attacks.

6.7. Physically unclonable function (PUF)

This is a chip that can be used to generate encryption private key and protect attacks from insider users. The public/private key exchanges are one of the most familiar techniques for securing data communications. In basic terms, this is a method in which two devices knows their public key, but each must get their private key. The most secure type of private key is one that is generated by hardware and not by human. A PUF-based device uses the little differences in each die in the chip to generate a unique key based on the unique properties of each piece of silicon. Using PUF-based device avoids insider user who have entry to the network from hacking it.

6.8. Built-in DPA countermeasures

Any hardware security device used should have countermeasures against Differential Power Analysis (DPA) attacks. DPA is a technique used to detect encryption keys using electromagnetic probe and a simple oscilloscope. To secure the IoT networks, any hardware devices used should have DPA licensed built-in countermeasures to guarantee acceptable design security.

6.9. FPGA-based security system

Field programmable Gate array (FPGA) is configurable hardware devices that can be used support many securities patterns through a combination of design security, hardware security and data security. Design security can be achieved at chip level protection including anti-tamper measures. Hardware security can be performed at board-level and the supply chain. Data security can be used to monitor all communications spanning to/from IoT devices. Many vendors (https://www.prnewswire.com/news-releases/microsemi-enables-fpga-based-root-of-trust-solution-for-embedded-systems-with-introduction-of-secure-boot-reference-design-244858941.html) are manufacturing SoC FPGA that can be used to implement HRoT. These FPGAs products have many hardware built-in capabilities like encrypted bitstreams, multiple key storage elements, licensed DPA countermeasures, secured flash memory, anti-tamper features and incorporate a PUF. Such powerful devices can provide the necessary components for protecting IoT/CPS systems.

7. HRoT Industrial Solution

Many vendors are producing chips or SoC FPGA solution that can be used to implement HRoT systems.

These chips can be used to support the following security features:

1. Secure booting.
2. Secure access control
3. Secure identification and authentication
4. Firmware integrity assurance
5. Protected chip storage
6. Secure debug
7. Runtime security
8. Protected firmware updates when needed

In this section we will introduce some of these chips and there necessary building blocks.

7.1. HRoT building blocks

HRoT can be composed from at least the following four basic building blocks (https://www.synopsys.com/designware-ip/technical-bulletin/understanding-hardware-roots-of-trust-2017q4.html), that implement the following functions:

1. Protective hardware, which provides a trusted execution environment (TEE) for the privilege software to run.
2. At least, implement one or more recognized cryptographic algorithms.
3. Provide a kind of tamper protection for the entire runtime.
4. An easy to use user interface that the host can interact with, through either the host CPU and/or a host controller general-purpose input output (GPIO) ports.

To meet these requirements, HRoT chip need to contain a variety of components. First, designer of the SoC must define what needs to be protected and how to implement the protection. Protected area can be implemented in various ways, including the use of a private bus that connects to the main bus through a gateway.

Next, HRoT need to run secure software/firmware on the secured CPU. The software running on that CPU defines the implementation for most of the security features supported in HRoT. Other resources found in SoC will support the implementation of the security and improve the performance of the secured elements.

The third element of a HRoT is the runtime memory that need to be protected since it contain sensitive data such as keys in plain-text and other important information.

Next essential element for a HRoT is the tamper resistance as outside code needs to be authenticated before running it on the secure CPU. This can be realized using a dedicated ROM that can be accessed only by HRoT

Hardware based cryptography usually uses fewer memory resources and runs faster than software cryptography. Hardware cryptography accelerator is needed to maintain high performance. It is necessary for these accelerators to use slower clock for the CPU to saves power as well as using less runtime memory to save area in the SoC. This feature is needed for cost-sensitive applications such as automotive applications.

HRoT also require a True Random Number Generator (TRNG) to produce a high level of entropy required to reduce predictability by generating secure high ephemeral keys needed by many protocols to secure the connection between end points. This module need secure untampered access.

HRoT need a secure clock which is necessary for applications that require a reliable time measurement.

The last component needed to be included in HRoT is secure storage which is essential for state knowledge applications. An anti-rollback feature for a device is example of state knowledge application and can be secure only if HRoT has secure access to a non-volatile memory.

7.2. Examples of available hardware secure modules

Major hardware vendors across the industry have begun to provide HRoT. In this subsection an overview of famous modules is introduced.

7.2.1. DesignWare tRoot H5 hardware secure module

This module is highly secure hardware root of trust introduced by Synopsys. The components of tRoot H5 is shown in figure (1). The module allows connected IoT/CPS devices to securely identify and authenticate themselves. tRoot's addresses complicated threats including device protection when powered down, at boot time, run time, and, during the communication with other devices or the cloud. tRoot offer SoC designers optimality and efficiency combination regarding of power, size and performance. tRoot gives a high level of security because the TEE is isolated in the hardware. It contains logic blocks such as the Secure Instruction Controller and Secure Data Controller to build many security features. It has also a HSM module that supports multi-stage secure boot, secure update and secure debug. Finally, it uses a PKCS#11 interface a key management support which to help manage both static and temporary keys.

7.2.2. AMD platform security processor

AMD combine a Platform Security Processor(PSP) together with the main CPU's x86 core. The PSP is a standalone 32-bit ARM Cortex-A5 core equipped with its own memory. PSP can use the full set of ARM's TrustZone hardware-enabled security services. It is designed to provide a secure processing path, TEE, TPM, and a cryptographic coprocessor. The PSP's primary role in normal operation is to protect the x86 core and provide HRoT. PSP boots first using its own ROM and SRAM and authenticate the code that

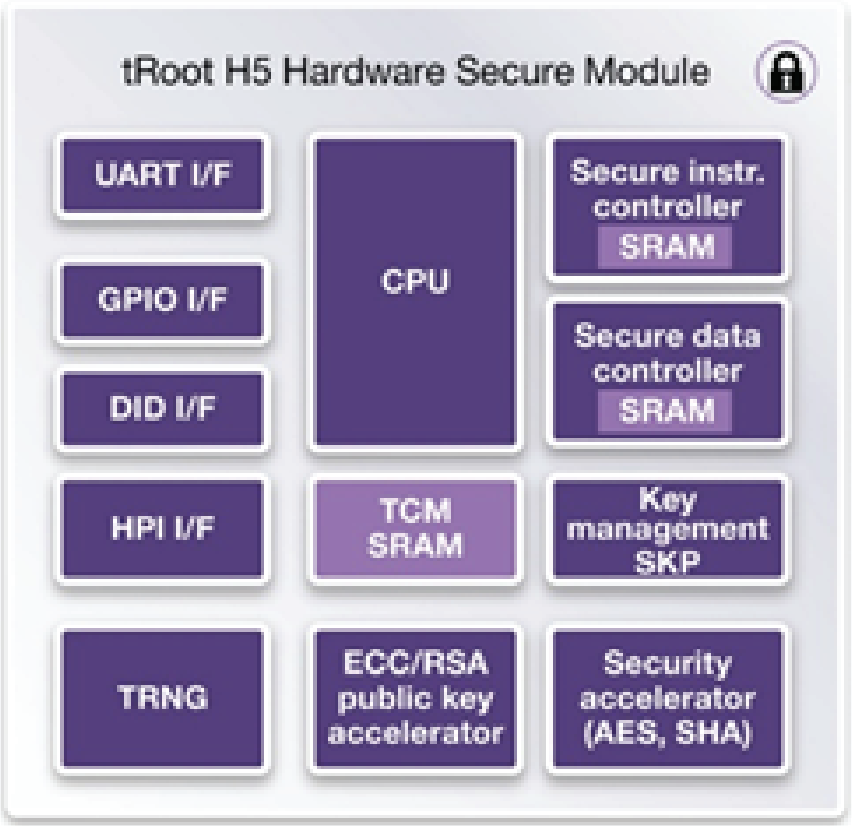

Figure 1: DesignWare tRoot H5 Hardware Secure Module with Root of Trust (https://www.synopsys.com/designware-ip/technical-bulletin/understanding-hardware-roots-of-trust-2017q4.html)

boots the x86 core using the UEFI Secure Boot process. ARM TrustZone (https://www.arm.com/products/security-on-arm/trustzone) is hardware-based security built into ARM SoCs to provide secure end points and roots of trust and it can be integrated into any ARM based system. TrustZone technology is usually used to run trusted boot and a trusted OS to build a TEE. Typical use cases include the safeguarding the authentication mechanisms, cryptography, and key issues. The ARM HRoT implementation is called CryptoCell. CryptoCell includes root of trust/key management using hardware cryptographic engines, secure boot, secure debug and lifecycle management. It is comprised of hardware, firmware and SoC-external tools

7.2.3. Intel boot guard

Intel Boot Guard is a hardware-based technology introduce with Intel fourth generation core processor. It is designed to prevent any replacement or tampering of the low-level UEFI firmware by malware or other unauthorized software. Boot Guard has three operating modes: verified boot mode, measured boot mode, or a combination of both.

Verified boot mode cryptographically authenticates an initial boot block. Measured boot uses a measuring process. The verified boot method is the one used by the Original Equipment Manufacturer (OEM). Boot Guard configurations differ from one OEM to another. In most cases, the OEM is responsible for constructing a public key for the verified boot and creating boot guidelines. The security of the verified boot depends on OEM's key pair. The OEM generates a 2048-bit key that is only used for authenticating the initial boot block, the private part of which need to be kept securely. The public part of the key is programmed into field programmable fuses during the manufacturing process. These fuses cannot be revised and therefore offer a solid beginning for implementing UEFI Secure Boot. The OEM is also responsible for setting rules about what steps to be taken if boot integrity checks fail, like shutting down the system or entering some kind of confidential repair mode.

7.2.4. ARM® Cortex^{TM}-M3 and SmartFusion2 FPGA fabric

Microsemi's SoC FPGAs enable a wide variety of security functions with the lowest power, smallest size and with the highest levels of security as compared with other FPGA family. For the IoT infrastructure, the SmartFusion2 SoC FPGAs provide with MSS (Microcontroller Sub System) ideal platform that balance between functionality and cost. It minimize power, offer small form factors and deploy best security solutions to prevent tampering, counterfeiting and installation of malicious code.

8. Conclusion

The IoT/CPS systems are evolving rapidly, and their security threats are increasing. To meet these threats, hardware-based protection is necessary. In this paper a survey of hardware-based protection methods are presented. The advance in VLSI and the availability of SOC FPGA and other hardware-based security solution from different vendors can be used effectively to design a secure IoT/CPS system with reasonable cost. Concentrating on hardware security will help users to have better insight about IoT/CPS security requirements, identify the vulnerabilities of these systems and gain good information on how to build secure IoT/CPS systems.

References

[1] Chui, Michael; Löffler, Markus; Roberts, Roger. "The Internet of Things". *McKinsey Quarterly*. McKinsey & Company. Retrieved 10 July 2014.

[2] A. Rajhans, S.W. Cheng, B. Schmerl, B.H. Krogh, C. Aghi, and A. Bhave, "An Architectural Approach to the Design and Analysis of Cyber-Physical Systems," Third International Workshop on Multi-Paradigm Modeling, Denver, CO, October 2009.

[3] Khaitan et al., "Design Techniques and Applications of Cyber Physical Systems: A Survey", IEEE Systems Journal, pp.1-14, 2014.

[4] Nordrum, *Amy (18 August 2016).* "Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated". *IEE Spectrum, August, 2016* https://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated

[5] Jump up∧ *Hsu, Chin-Lung; Lin, Judy Chuan-Chuan.* "An empirical examination of consumer adoption of Internet of Things services: Network externalities and concern for information privacy perspectives". *Computers in Human Behavior.* **62**: *516–527.* doi:10.1016/j.chb.2016.04.023.

[6] Smith, **CSO,** *Peeping into 73,000 unsecured security cameras thanks to default passwords, cited 8-3-2018,* https://www.csoonline.com/article/2844283/microsoft-subnet/peeping-into-73-000-unsecured-security-cameras-thanks-to-default-passwords.html

[7] BBC website (22 December 2014), Hack attack causes 'massive damage' at steel works, cited on 6/3/2018, http://www.bbc.com/news/technology-30575104

[8] KIM ZETTER, EVERYTHING WE KNOW ABOUT UKRAINE'S POWER PLANT HACK, 2016, cited 5/3/2018'/

[9] IoT e-book, 2018, https://ihsmarkit.com/forms/thankyou.html?efid=t+m2jEyFYkJQYyoP3YvuHA==&&gasc_id=862037098&&gasc_label=scrXCLnM7moQ6siGmwM

[10] Y. Challal, E. Natalizio, S. Sen, and A. Maria Vegni "Internet of Things security and privacy: Design methods and optimization", Add Hoc Network, vol.32, Science Direct, p.p1-2, 2015.

[11] Ch. Qiang, G. Quan, B. Yu, L. Yang, "Research on Security Issues of the Internet of Things", International Journal of Future Generation Communication and Networking (IJFGCN), vol.6, NO.6, IEEE, pp 1-10, 2013.

[12] A. Riahi, Y. Challal, E. Natalizio, Z. Chtourou, A. Bouabdallah" A systemic approach for IoT security', International Conference on

[13] Distributed Computing in Sensor Systems (DCOSS), Cambridge, MA, IEEE, p.p.351-355, 2013.

[14] F. Olivier, G. Carlos, N. Florent "New Security Architecture for IoT Network", Procedia Computer Science, vol.52, Science Direct, pp.1028-1033, 2015.

[15] Lu, C., 2014. Overview of Security and Privacy Issues in the Internet of Things, 1–11. Mahalle, P., Babar, S., Prasad, N.R., Prasad, R., 2010. Identity management framework towards Internet of Things (IoT): roadmap and key challenges. Recent Trends Netw. Secur. Appl. - Commun. Comput. Inf. Sci. 89, 430–439.

[16] Zhao, K., Ge, L., 2013. A survey on the Internet of Things security. In: Proceedings of the 9th International Conference on Computational Intelligence and Security, CIS 2013, 663–667.

[17] Zhu, C., Leung, V.C.M., Shu, L., Ngai, E.C.H., 2015a. Green Internet of Things for SmartWorld. IEEE Access 3, 2151–2162.

[18] Zhu, S., Setia, S., Jajodia, S., 2015b. LEAP: efficient security mechanisms for large-scale distributed sensor networks categories and subject descriptors. ACM Trans. Sens. Netw. (TOSN) 2 (4), 500–528.

[19] Siddhartha Kumar Khaitan and James D. McCalley," Design Techniques and Applications of Cyberphysical Systems: A Survey, IEEE system Journal, vol.9, issue 2, pp.1-15, 20, April 2018.

[20] M. Ilic *et al.*, "Modeling future cyber-physical energy systems," in *IEEE PES-GM, - Conversion and Delivery of Electrical Energy in the 21st Century*, 2008, pp. 1–9.

[21] M. Ilic *et al.*, "Modeling of future cyber–physical energy systems for distributed sensing and control," *IEEE Trans. on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 40, no. 4, pp. 825–838, 2010.

[22] P. Zhao *et al.*, "A conceptual scheme for cyber-physical systems based energy management in building structures," in *IEEE/IAS Intl. Conf. on Industry Applications (INDUSCON)*. IEEE, 2010, pp. 1–6.

[23] Y. Tan, M. Vuran, and S. Goddard, "Spatio-temporal event model for cyber-physical systems," in *ICDCS Workshops*, 2009, pp. 44–50.

[24] J. Lin *et al.*, "Modeling cyber-physical systems with semantic agents, in *COMPSACW*. IEEE, 2010, pp. 13–18.

[25] Y. Tan, M. Vuran, S. Goddard, Y. Yu, M. Song, and S. Ren, "A concept lattice-based event model for cyber-physical systems," in *ICCPS*, 2010, pp. 50–60.

[26] P. Derler *et al.*, "Modeling cyber–physical systems," *Proc. of the IEEE*, vol. 100, no. 1, pp. 13–28, 2012.

[27] J. Kim and D. Mosse, "Generic framework for design, modeling and simulation of cyber physical systems," *ACM SIGBED Review*, vol. 5, no. 1, pp. 1–2, 2008.

[28] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, 2013.

[29] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer Networks, vol. 54, no. 15, pp. 2787 – 2805, 2010.

[30] Qi Jing, et al, Security of the Internet of Things: Perspectives and challenges, Wireless Networks, Vol. 20, Issue 8, pp. 2481-2501 November 2014, DOI 10.1007/s11276-014-0761-7.

[31] Yaqoob, Ibrar, et al., 2017. Enabling communication technologies for smart cities. IEEE Commun. Mag. 55 (1), 112–120.

[32] Valmohammadi, C., 2016. Examining the perception of Iranian organizations on Internet of Things solutions and applications. Ind. Commer. Train. 48 (2), 104–108.

[33] Pongle, P., Chavan, G., 2015. A survey: attacks on RPL and 6LoWPAN in IoT. International Conference on Pervasive Computing: advance Communication Technology and Application for Society, ICPC 2015, 0(c), 0–5.

[34] Tsai, C.-W., Lai, C.-F., Vasilakos, A.V., 2014. Future internet of things: open issues and challenges. Wirel. Netw. 20 (8), 2201–2217.

[35] O. Garcia-Morchon, S. Kumar, S. Keoh, R. Hummen, and R. Struik, "Security considerations in the ip-based internet of things," Working Draft, IETF Secretariat, Internet-Draft draft-garcia-core-security-06, September 2013, http://www.ietf.org / internet-drafts/draft-garcia-core-security-06.txt. [Online]. Available: http://www.ietf.org/internet-drafts/draft-garcia-core-security-06.txt

[36] M. Weyrich and C. Ebert, "Reference architectures for the internet of things," IEEE Software, vol. 33, no. 1, pp. 112–116, 2016

[37] P. Adolphs, H. Bedenbender, D. Dirzus, M. Ehlich, U. Epple, M. Hankel, R. Heidel, M. Hoffmeister, H. Huhle, B. K¨archer, et al., "Reference architecture model industrie 4.0 (rami4. 0)," ZVEI and VDI, Status Report, 2015.

[38] I. I. Consortium et al., "Industrial internet reference architecture," Industrial Internet Consortium, Tech. Rep., June 2015.

[39] A. B. HEU, P. G. HEU, A. O. CEA, and J. Stefa, "Internet of things architecture," 2013.

[40] R. H. Weber, "Internet of things: Privacy issues revisited," Compute Law & Security Review, vol. 31, no. 5, pp. 618–627, 2015.

[41] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, "Future internet: the internet of things architecture, possible applications and key challenges," in Frontiers of Information Technology (FIT), 2012 10th International Conference on, pp. 257–260, IEEE, 2012.

[42] H. Ning and Z. Wang, "Future internet of things architecture: like mankind neural system or social organization framework?," IEEE Communications Letters, vol. 15, no. 4, pp. 461–463, 2011.