International Journal of Scientific Research in Computer Science and Engineering
Vol.6, Issue.5, pp.41-44, October (2018)

Survey Paper

E-ISSN: 2320-7639

A Survey on Developing Secure IoT Products

Priyang Bhatt[1*], Bhasker Thaker[2], Neel Shah [3]

[1]Gujarat Technological University, V. V Nagar, Anand , India
[2] Dept. of ECE, G H Patel College of Engineering and Technology, V. V Nagar, Anand, India
[3]TCR Advanced Engineering Pvt. Ltd., Vadodara, India

[*]*Corresponding Author: priyangbhatt@gcet.ac.in*

Available online at: www.isroset.org

Received: 24/Sept/ 2018, Accepted: 11/Oct/ 2018, Online: 31/Oct/2018

Abstract **—** The Internet of Things (IoT) is a vision of a globally interconnected network of various sensors, actuators and embedded systems connected to each other via wired/wireless connections, interacting with the environment and each other to create value. This vision would revolutionize current models and processes in business and daily life. Needless to say being such an impactful concept, security in IoT becomes even more critical than before. In this we paper we survey and highlight some crucial and primary concepts related to IoT Security which have to be incorporated in the IoT system design and development as well as during deployment of the IoT applications.

Keywords—IoT, Privacy, Security, Product Development, Secure Design

I. INTRODUCTION

Knowing the significant benefits of the Internet of Things (IoT) i.e., having a connected single ubiquitous network of devices. IoT covers a huge range of applications, from wearable technologies, industrial automation, etc. The challenge to this concept of Internet-connected devices is security. In IoT, there are various security issues and major privacy concerns for the end users. Its advanced capabilities and value generation for businesses diminishes when the concept is viewed from a security viewpoint, proper steps should be taken in the initial phase itself before going for further development of the concept, for an effective and widely accepted adoption.[1]

Security becomes an important concern for IoT due to the unprecedented volume of data that is flowing through these systems. The level of interconnectivity in these IoT systems which can potentially be exploited to harm the devices and the end users. These factors force developers and product designers to rethink the concept of security, which becomes even more critical than before. Millions of devices that are under control of external entities, the integrity of perennially flowing data, and heterogeneous environments that co-exist in the same IoT environment must be taken into consideration while securing IoT devices.[2]

This paper explores this factors initiating from various security risk and threats in section II, section III explores security aspects of embedded systems their requirements, challenges, and current state. In Section IV device support for IoT security is discussed with the current works various leading semiconductors companies, and drawing the conclusion in section V.

II. RISK AND THREATS

In [1] the risk of physical attacks on IoT systems, attacks on systems where data or control of the system is in the cloud, threats on interoperable systems, remote attacks on IP and gateways devices, possibilities of MITM or Masquerading attacks while using smartphones and computers with IoT systems are discussed, also the possibility of using old and decommissioned devices as Trojan horses, and phishing attacks are also highlighted.

In [2] risks and threats of wireless sensor networks(WSN) and RFID based IoT systems are highlighted and discussed. The following figures provide the summary of it.

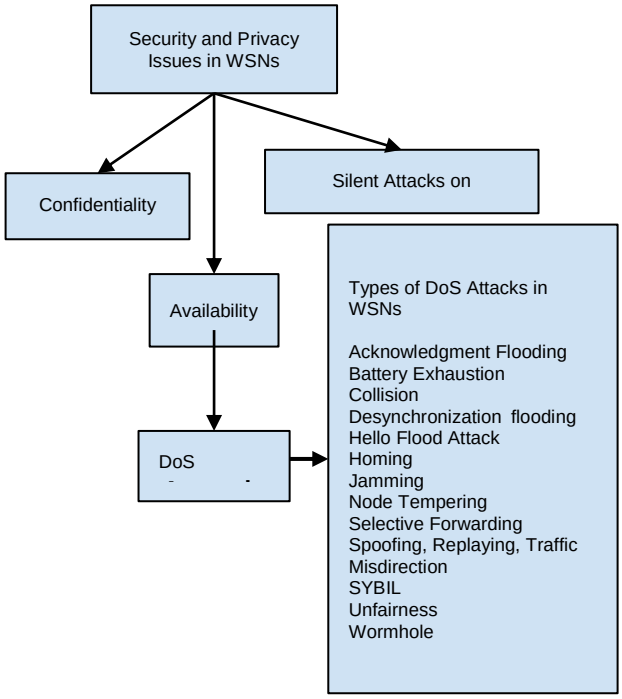

Fig. 1. *Security and Privacy Issues in WSNs*

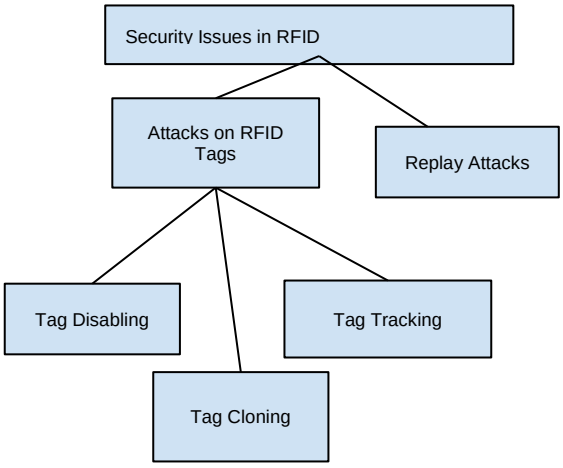

Fig. 2. *Security Issues in RFID*

Thus, [2] shows possibilities and the major risk of DoS(Denial-of-Service) or DDos(Distributed Denial of Service) attacks. [3] provides reasons for complications arising in IoT security such as,

- **Each connected device is a potential point of attack:** as every connected device can serve as an entry point to the network of devices which can lead the attacker to compromise the whole network or system of interconnected devices.
- **New kinds of devices bring new kinds of vulnerabilities:** Internet of things(IoT) conceptualizes a network of ubiquitous devices, as each kind of device, has its own vulnerabilities and risks which they bring to the whole system when connected thus applying a common policy or protocol will not seem like a workable idea.
- **Data originates on devices in unsecured locations:** Devices will generate data from all types of locations from normal to extreme. Thus securing such devices and maintaining data integrity will be a challenge.

III. SECURITY OF EMBEDDED DEVICES

A. *Background*

proven security techniques. But in the case of embedded devices firewalls are virtually absent and most of them rely on simple password authentication and security protocols, which in present times where the number and sophistication of attacks are increasing leaves these devices and systems vulnerable.[4]

B. *Requirements[5]*

Fig. 3. *Typical security requirements of embedded systems*

Figure 3 lists the typical security requirements seen across a wide range of embedded systems, in general, they can be classified as below,

- **User identification:** authorization of users before allowing them to use the system.
- **Secure network access:** letting a device use network connection or service access only if it is authorized.
- **Secure communications:** authentication of communication between peers(devices), confidentiality and integrity of flowing data, Non-repudiation of a transaction, and protecting the identity of the communicators.
- **Secure storage:** mandates confidentiality and integrity of sensitive information stored in the system.
- **Content security:** enforces the usage restrictions of the digital content stored or accessed by the system.
- **Availability:** ensures that the system can perform its intended function and service legitimate users at

all times, without being disrupted by denial-of-service attacks.

C. Challenges

As mentioned in [5], that a general or a common security model will not apply to all of the embedded systems as their design and deployment differ from each other considerably, with this there are other challenges which are touched upon,

- **Processing Gap:** most of the embedded systems architectures are not capable of handling the computational demands of security processing as of the constantly streaming data.
- **Battery Gap:** encryption and decryption are resource intensive tasks i.e., systems will have a significant energy consumption overheads of supporting security on battery-constrained embedded systems are very high. Slow growth rates in battery capacities are easily outpaced by the increasing energy requirements of security processing, leading to a battery gap.
- **Flexibility:** An embedded system is often required to execute multiple and diverse security protocols and standards in order to support, **multiple security objectives, interoperability in different environments, and security processing in different layers of the network protocol stack.** Furthermore, with security protocols have to keep continuously evolving to become immune with increasing sophistication of attacks, therefore, the security architecture has to be flexible enough to adapt easily to changing requirements.
- **Tamper Resistance:** Attacks due to malicious software such as viruses and trojan horses are the most common threats to any embedded system that is capable of executing downloaded applications. These attacks can exploit vulnerabilities in the operating system or application software, procure access to system internals, and disrupt its normal functioning. These attacks compromise Integrity, Privacy, and Availability thus it becomes necessary to incorporate countermeasures against these attacks in both Hardware and Software.
- **Assurance Gap:** Reliable systems must be able to handle the wide range of situations. Secure systems must continue to operate reliably despite attacks from intelligent adversaries who seek out undesirable failure modes. As the complexity of the system increases, there are inevitably more possible failures that need to be addressed. With an increase in complexity in an embedded system, it becomes more and more difficult for the designers to have not overlooked a serious weakness.
- **Cost:** It is one of the fundamental factors that influence the security architecture of an embedded system. The implications of a security-related design choice on the overall system cost must be considered. The Federal Information Processing Standard (FIPS 140-2) [6] there are four levels of secure systems; **Level 1:** requires minimum physical protection, **Level 2:** requires the addition of tamper-evident mechanisms such as a seal or enclosure, **Level 3:** specifies stronger detection and response mechanisms, **Level 4:** mandates environmental failure protection and testing, as well as highly rigorous design processes. Thus, providing increasing levels of security using increasingly advanced measures will lead to higher system costs, design effort, and design time. Thus an optimal system design to be developed which will not leave any aspect untouched.

IV. Device Support for IoT Security

Many new IoT devices developed in recent times like NXP iMX7 Solo applications processor that is used on the NXP WaRP7 IoT and Wearable Development Platform which is an implementation of the Arm® Cortex®-A7 core, operating at speeds of up to 1 GHz, as well as the Arm® Cortex®-M4 core. It supports multiple memory types including 16/32-bit DDR3L/LPDDR2/LPDDR3-1066, Quad SPI memory, NAND, eMMC, and NOR and several high-speed connectivity connections including Gigabit Ethernet with AVB and USB. Both parallel and serial Display provide built-in support that makes it easier to build in security.[7, 8]

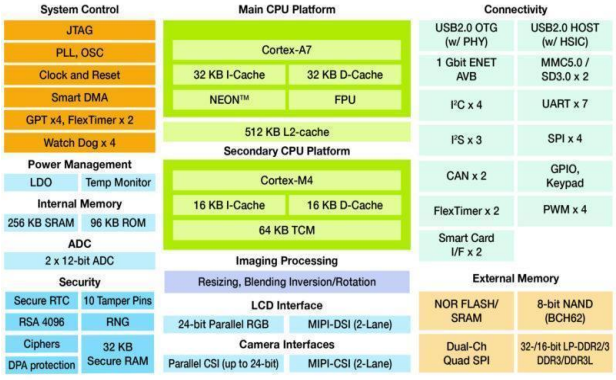

Fig. 4. *iMX 7 Solo Applications Processor Block Diagram[7]*

As shown in the figure above, the processor includes hardware-accelerated encryption support with the CAAM (cryptographic acceleration and assurance module). This module contains cryptographic and hash engines that support a wide range of cryptographic standards. The random number generator (RNG) which is one of the

requirements for encryption algorithms which is used to generate the keys. The CAAM offers National Institute of Standards and Technology (NIST) certified pseudo and true random number generators. By providing such support in the hardware itself, the IoT systems can be developed with enhanced security than before.[6, 7]

Still one cannot be assured that security aspect will be taken care of just by software and hardware encryption, because using simple power analysis (SPA) or differential power analysis (DPA) the attacker can predict keys by measuring device power consumption to determine secret keys, which can be much easier than it sounds, as the cryptographic algorithms can involve rotating registers that contain the keys. In the case of NXP WaRP7 IoT and Wearable Development Platform countermeasures of these potential attacks can be found on the board.[7]

Secure non-volatile storage (SNVS) [7] is a hardware storage system that can be used in the IoT systems as it determines whether the device is in a secure state, which in turn determines whether its resources can be accessed. When in a secure state, special cryptographic keys can be used to decrypt long-term secrets. A security violation can be triggered by JTAG events, power glitches, Master Key ECC check failure, software-reported issues, and hardware reported tampering using the tamper pins. Is situation as such is identified, then system activates security-related hardware or software. In the case of sensitive applications, the tamper pins could trigger the hardware that will automatically and immediately erase the Master Key, denying access to and erasing the contents of the secure memory. In some cases, real-time clocks are a potential vulnerability. A simple example is electricity metering, where the clock could always be set to a time where the lowest rate is in force. Protecting the real-time counter using the same SNVS offers additional protection.

To ensure code integrity approaches to isolate applications and ensure that the firmware of the system has not been compromised. ARM's TrustZone technology [8] is an example of such an approach, identifying secure and non-secure worlds, blocking non-secure software from accessing secure resources directly. Effectively each physical processor has two virtual cores: one considered secure and the other non-secure. This technology relies on the secure low-level firmware. The inclusion of a high-assurance boot (HAB) feature, which uses digital signatures to recognize authentic software and prevent unauthorized software from gaining control of the boot sequence, ensures that this firmware cannot be compromised.[9, 10]

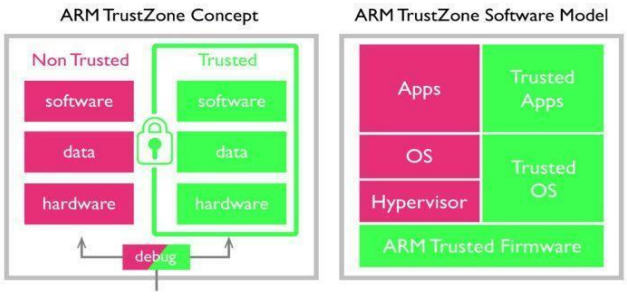

Fig. 5. *ARM TrustZone*

V. CONCLUSION

All the papers and articles which we have surveyed point to a single conclusion that, security is an important concern for IoT and its full-scale deployment using current insecure IoT products in daily life can have huge privacy and ethical issues. Also while designing IoT system one must adopt a holistic approach towards it also, an active research is being carried out on this front and various different technologies and techniques are been developed which can aid in secure design of an IoT system, but still we believe that the security part should be worked upon in its embryonic stage to ensure that the magnification of dangers with scale and complexity is mitigated and its effect can be significantly reduced.

REFERENCES

[1] Developing Secure Products in the Age of IoT - Internet Of Things, https://electronicsofthings.com/research-articles/developing-secure-products-age-iot/.

[2] Tuhin Borgohain, Uday Kumar, Sugata Sanyal, "Survey of Security and Privacy Issues of Internet of Things," https://arxiv.org/ftp/arxiv/papers/1501/1501.02211.pdf.

[3] A Whitepaper on The Internet of Things: Reduce Security Risks with Automated Policies, Cisco, https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/security-risks.pdf.

[4] Security Requirements for Embedded Devices – What is Really Needed?, http://www.iconlabs.com/prod/security-requirements-embedded-devices-%E2%80%93-what-really-needed.

[5] Srivaths Ravi, Anand Raghunathan, Paul Kocher, Sunil Hattangady, "Security in Embedded Systems:Design Challenges," http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.586&rep=rep1&type=pdf.

[6] FIPS PUB 140-2.Security Requirements for Cryptographic Modules.Available at http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf.

[7] i.MX 7Solo Processors - Heterogeneous Processing with Arm® Cortex®-A7 and Cortex-M4 cores, https://www.nxp.com/products/processors-and-microcontrollers/applications-processors/i.mx-applications-processors/i.mx-7-processors/i.mx-7solo-processors-heterogeneous-processing-with-arm-cortex-a7-and-cortex-m4-cores:i.MX7S.

[8] i.MX 7Solo Family of Applications Processors Datasheet, https://www.nxp.com/docs/en/data-sheet/IMX7SCEC.pdf.

[9] TrustZone – Arm, https://www.arm.com/products/security-on-arm/trustzone.

[10] TrustZone – Arm Developer, https://developer.arm.com/technologies/trustzone.

