ISSN: 2455 - 1597

ijergs
Open Access

International Journal of Engineering Research and Generic Science (IJERGS)
Available Online at www.ijergs.in

Volume - 5, Issue - 3, May - June - 2019, Page No. 28 - 39

Study of Security Challenges in IoT System

[1]Dr. Alok Mishra, Research Supervisor, Sunrise University, Alwar, Rajasthan

[2]Ajay Saini, Research Scholar, Sunrise University, Alwar, Rajasthan

Abstract

The Internet of Things (IoT) holds many benefits for our lives by removing menial tasks and enhancing the accessibility of everyday objects. With your personal data and device access, you trust the manufacturers and you may not be aware of the danger of transmitting your information over the internet to jeopardize your privacy. The internet-of-things may not be as secure as you think because the devices used are constrained by a number of variables that attackers may use to gain access to your data/device and anything they connect to, and as the internet-of-things is all about connecting devices together, there may be a weak point in everything it takes to gain full access. In this paper, we have a look at the new IoT security technologies and the most effective methods to secure IoT devices.

Keywords: Hardware, IoT, Security, Efficiency

Introduction

The Internet of Things (IoT) has developed immense interest over the past few years. The concept of the IoT was first proposed by Kevin Ashton in 1999. Due to rapid advances in mobile communication, Wireless Sensor Networks (WSN), Radio Frequency Identification (RFID), and cloud computing, connectivity between IoT devices has become more convenient than ever. IoT systems are capable of communicating with one another. The IoT world contains a large range of devices, including cell phones, personal computers, PDAs, laptops, tablets, and other embedded portable devices. IoT systems are based on cost-effective sensors and wireless connectivity schemes to communicate with each other and transfer valuable information to a centralized system. Information from IoT devices is further processed within the centralized system and transmitted to the intended destinations. Our daily lives, with the exponential growth of communication and internet technology, are more focused on the imagined space of the virtual world. People can work, shop, talk (keep pets and plants) in the virtual environment created by the network, while humans live in the real world. However, it is very hard to substitute all human tasks for fully automated life. There is a limit to fictional space that restricts the future development on the internet of better services. The IoT, on the same platform, has successfully targeted imaginary space and the real world. The main objectives of IoT are to configure a smart environment and self-conscious autonomous devices such as smart living, smart appliances, smart health, and smart cities, among others.

Nowadays, with more and more devices connected through the internet, the IoT device adoption rate is very high. According to the assessment, there are approximately 30 billion connected items with approximately 200 billion connections, generating revenues of around EUR 700 billion by 2020. Now in China, there are nine billion devices that are expected to exceed 24 billion by 2020. In the future, the IoT will totally alter our preferences and business models. This will allow people and devices to connect to any computer at anytime, anywhere, under ideal conditions, using any network and any service. The main aim of the IoT in the future is to create a superior world for humans.

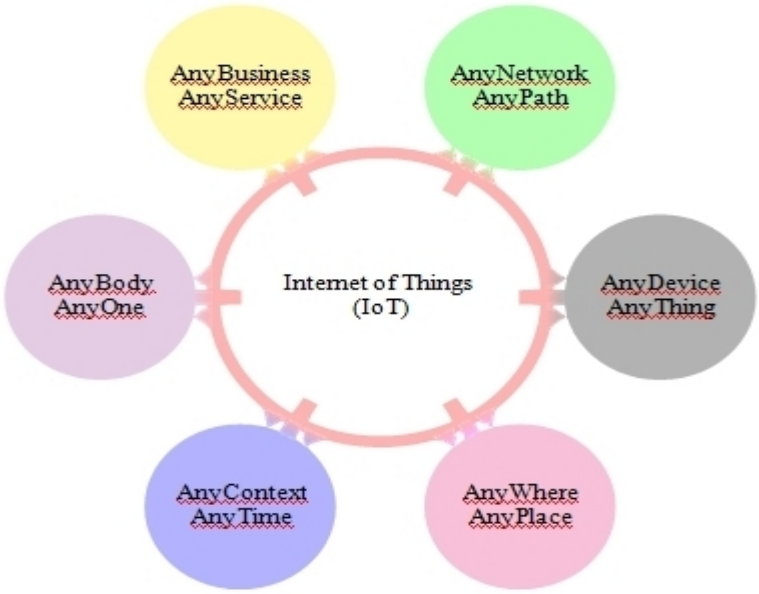

Fig. 1: Definition of IoT

IOT Architecture

Layers: The IoT architecture can be split into three fundamental layers, but they can adapt depending on the use case, as additional layers may be needed for some industry solutions.

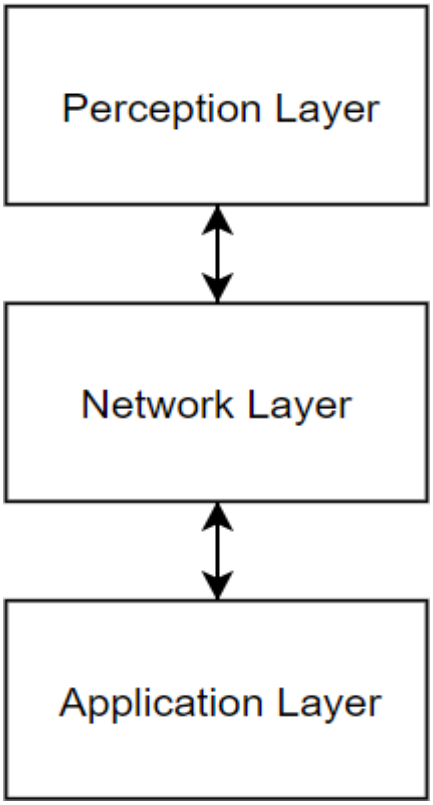

Fig. 2: IoT Layer Diagram

Perception Layer: Physical devices that communicate with other devices and the physical world, such as the sensors and

actuators, are composed of the perception layer to both transmit and receive data to other devices using wireless technology. The aim of this layer is to collect from its sensors and actuators all the data. And may be sent to the network layer.

Network Layer : The network layer handles data that is transmitted between smart devices as well as between network devices and servers, and can also be used in a readable format to transfer and process data from the perception layer to the receiving device. Some say that this is where the Internet of Things exists as it links the cyber and physical world and allows them to communicate with each other. This layer uses a variety of technologies (routers, switches, cloud computing) that process and direct the data to the specific application layer where the data can be read. Many communication technologies are used to relay data that may depend on the implementation of the perception layer, but this requires more than just WIFI, as other wireless technologies may have benefits such as short-range data capture by Bluetooth or aggregation of RFID data.

Application Layer: The application layer, which can also be called the business layer, offers basic services to users and receives the data from the sensors/actuators from the perception layer after being translated into a readable format by the network layer. The application layer can then use this data to provide services or perform operations based on the data obtained. This layer would analyze and store the information collected in order to make predictions or see trends that may be invaluable for a company to see its products/devices in current and future states.

Enabling Technologies

Hardware Platforms: In abundance, Table, IoT hardware platforms and development kits are available. 1 highlights some of the most popular and recent launches and you can see that when they are designed to accommodate many solutions, the requirements vary a lot.

Table. 1: IOT Hardware Platforms

Hardware	Processor	Clock speed	System Memory	Flash memory	Voltage
Arduino Yun	Atheros AR9331	16Mhz	2.5KB	32KB	5V
Raspberry Pi 3 B+	Broadcom BCM2837	1.2GH Z	1GB	SD card (-32gb)	5V
ESP8266	Tensilica L106 32-bit microcontrol Ler	80MH Z	64KB	94KB+16MB external	3.3V
Beaglebon e Black	AM335x ARM , Cortex-A8	1GHZ	512MB	4GB	5V
Intel Edison	22 nm Intel SoC	500MHZ	1GB	4GB	3.3 –4.5V
Netduino 3 wifi	Cortex-M4	168M HZ	164 + kb	1408KB+ 2GB SD	3.3 – 5V
UP Squared	Celeron N3350	2.4GH Z	2GB	32GB	5V

The lower-specific devices would be suitable for Processing power which would make them more effective at a lower cost than using a system with high specifications.

Communication: IOT requires a way to send data between devices to both collect the required information and receive instructions based on the sent data. IOT needs to work. A variety of commuting systems could be used depending on the deployment of the system and there are both short and long-range requirements.

Table. 2: IoT Commutation standards

Name	Frequency	Range	Examples
RFID	13.56 MHz	10cm - 200m	Road tolls, Building Access, Inventory
En Ocean	315 MHz,868 MHz,902 MHz	30 -300m	Wireless switches, sensors and controls
NFC	13.56 MHz	< 0.2 m	Smart Wallets/Cards, Action Tags,Access Control
Bluetooth	2.4GHz	1-100m	Hands-free headsets, key dongles, fitness Trackers
WIFI	2.4 GHz,3.6 GHz	100m +	Routers, Tablets, etc
Weightless	470–790MHz	Up to 10km	Smart meters, traffic sensors, industrial monitoring
GSM	850 -900MHz	n/a	Cell phones, M2M, smart meter, asset tracking

Table. 2. Shows some of the most used concepts of communication, as well as some instances deployed. All come with advantages and disadvantages, such as RFID, which is ideal for very close proximity commutation, but lacks some encryption, so it is prone to data hijacking, but the attacker would have to be very close to doing so, making long-range attacks difficult to execute.

Cloud solutions: For the internet-of-things, cloud solutions are very relevant as it enables ubiquitous access to a common pool of resources, all the devices in the perception layer can send the information to be processed and accessed by the application layer using the network layer.

There are many different types of cloud computing that are ideal for various solutions. Displays and characteristics of some of the most used suppliers.

IaaS: Infrastructure as a Service is a model where an organization/business leases specific services offered for the solution, and a pay-as-you-go basis is prevalent. This means that you can only pay for what you use, even though you don't use any services, unlike other companies that rent all their services for a fixed amount.

PaaS: The Platform as a service is designed to streamline the deployment process by moving system management to the provider and offering pre-configured business/organization components such as databases / application servers /

programming languages.

SaaS: The Platform as a service is designed to streamline the deployment process by moving system management to the provider and offering pre-configured business/organization components such as databases / application servers / programming languages.

IOT Device Constraints

Power Consumption: Devices for a role are created and will be designed based on that intent. And the more a device has to do, such as adding data to power usage from storage/collection. To introduce additional security to a device, it would take more power than the original design, implementing methods such as encryption would increase the power needed to complete the same operation.

Processing: The processing on a device is another drawback to the implementation of better safety as the process will have to perform its designed task as well as the security on top of which additional gates/transistors and additional modules can be used to do so.

Design: The design of a computer is also a constraint, as a factor that adds additional modules can also be the unit size. The size and complexity of device designs can be affected by transistors. The cost and efficacy of these implementations could make the solution unviable for deployment. Through running simulations, the design can be checked and refined before construction, which will reduce the cost and make the system more efficient.

IOT Device Efficiency

The benefit of making a deice more robust is that to do the same job, the machine does not need as much power or energy, which will reduce costs as a result.

Code compression: Text compression can improve the performance and power consumption of computers when used with encryption and integrity checking to secure processor memory transactions and can reduce the memory footprint as well as provide more information per memory access.

IoT Device Security

With this increase in IoT devices that streamline a lot of processes, we can see many advantages for both clients and businesses, but they can also come with some disadvantages. One of these disadvantages is security & privacy, and having our personal data (banking information/location/activity) exchanged between devices risks losing a lot of our privacy. The IoT opens the doors to many malicious hacks who try to abuse IoT system vulnerabilities in order to access our personal data for their own benefit to be abused.

Authentication

Noise Insertion: Noise cancelation helps protect the raw data when it is inside the processing system to prevent an attacker from using side-channel attacks to retrieve the data. The way this method works, by inserting noise using a key for confidential data, although this method is not as reliable as encryption, it has the advantage of being very lightweight in comparison. Through selecting key places where the data noise is canceled to make it readable, you can keep the data safe on the device and eliminate unnecessary overhead.

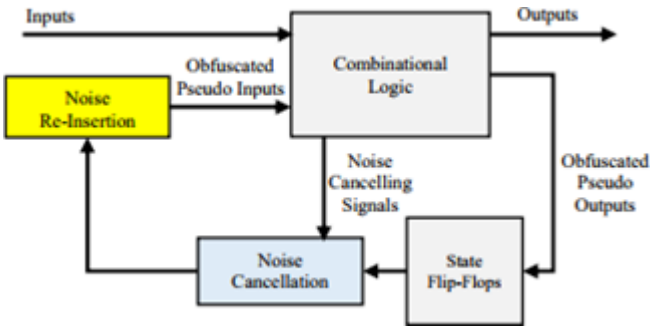

Fig. 3: Noise cancelling scheme

Logic Locking: Logic locking is a relatively recent technique involving adding additional gates to the locking configuration of the "Key Gates" that would change the output and affectively lock the correct functionality of the gates.

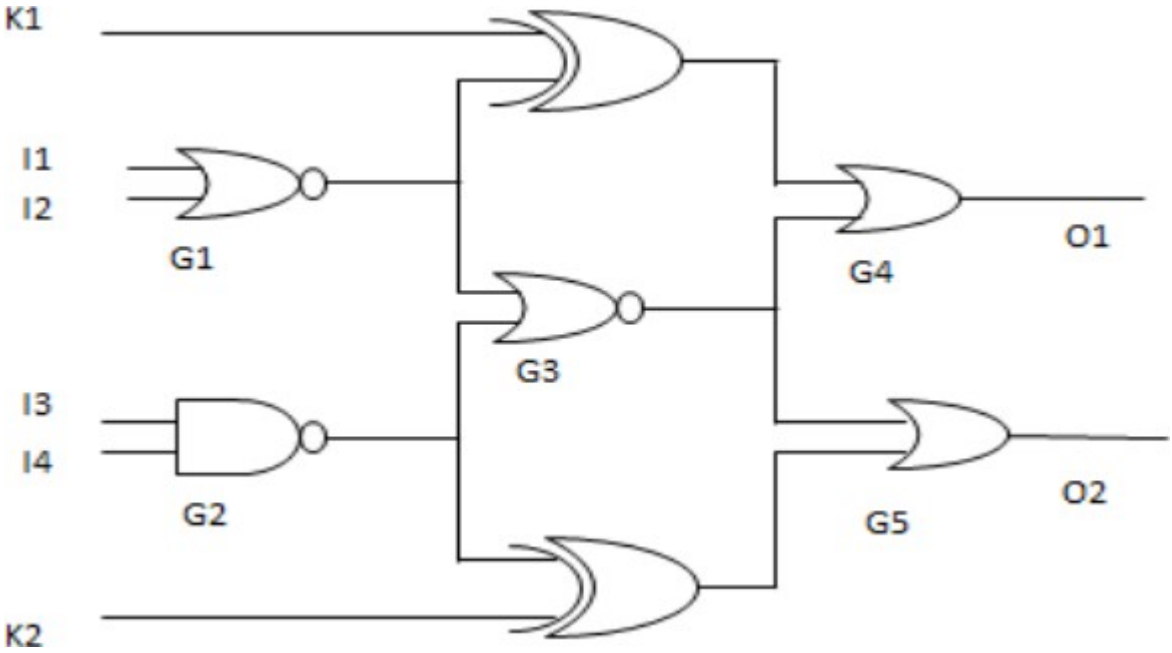

Fig. 4: Logic Locking example

This technique increases the level of security over other less effective strategies, such as IC Camouflaging, which is the dummy contact insertion process, so that an attacker is able to extract an incorrect net list. Just in Fig. 3. For the process to continue, some XOR gates are used as main gates that need to be 0, otherwise k1 and k2 would mask the original output. When comparing logic locking to an older method (OC Cell) without compromising security, there was a drastic drop in the delay.

Detection & prevention

Security Auditing Module: The purpose of the proposed security auditing module, in addition to a security framework, is to control both internal and external operations in order to evaluate the reliability of devices that prevent system damage, alert the network of any fatal device problems, and recognize security threats.

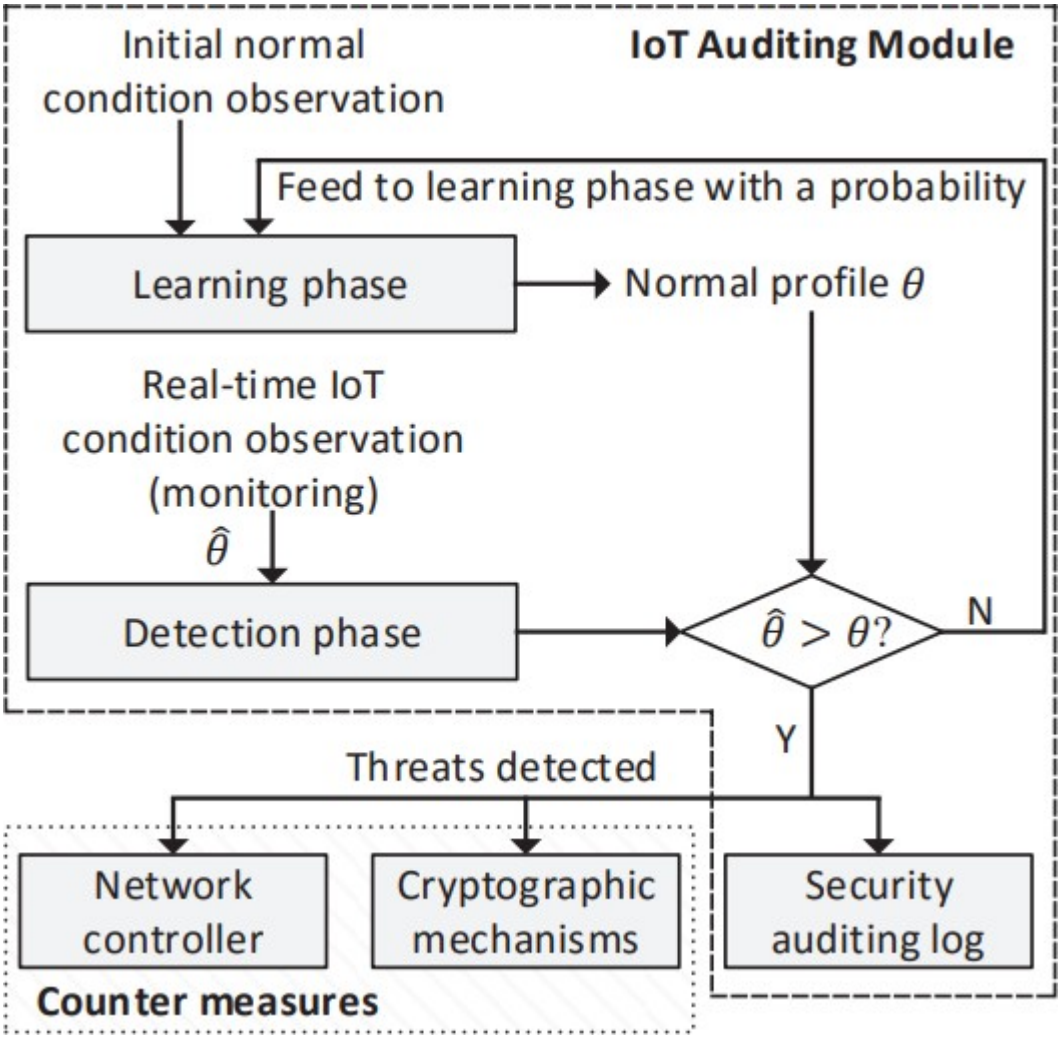

Fig. 5: Security auditing module

Other time can be used to build profiles that represent system behavior that can be analyzed to improve efficiency and quickly detect threats that could avoid data loss / compromise.

Attack Detection Unit: When a system is under threat, the purpose of the attack protection unit is to detect and alert the machine of this attack so that device damage or a compromised device can be prevented.

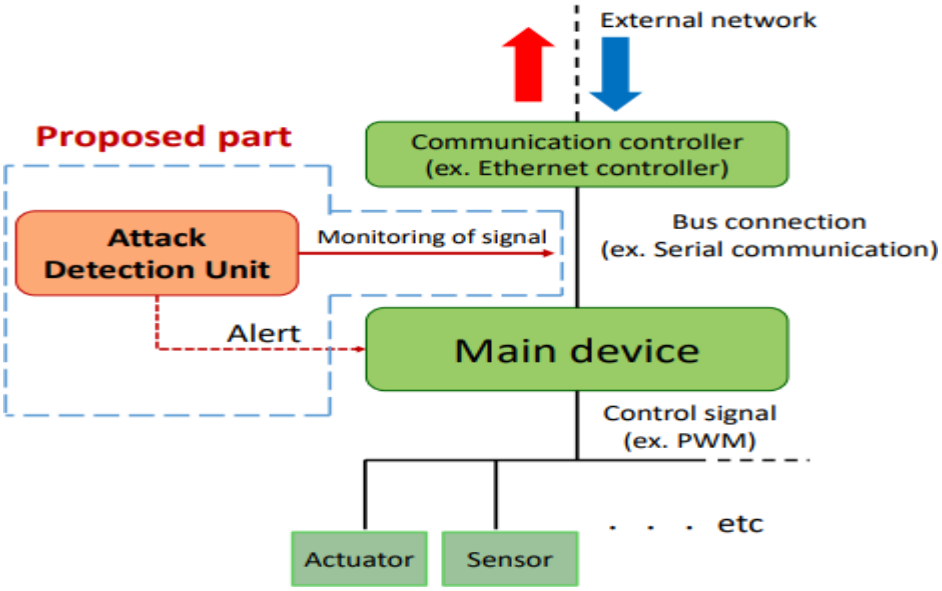

Fig. 6: Attack Detection unit Implementation

This is done by monitoring electrical signals from the communication controller that detect irregularities in the physical characteristics of bus communication between the CM and the main device. This is not part of the main machine and will not interfere with the devices' processing power and can be implemented on several devices with a touch controller.

Random Canaries Repository: RSR aims to defend against Stack smash attacks that leverage buffer overflow vulnerabilities to hijack application power. RCR is an extension of the Stack Smash protector by building a repository of canary random values that are used when an attack is detected by the program. The RCR solution improves the attack complexity.

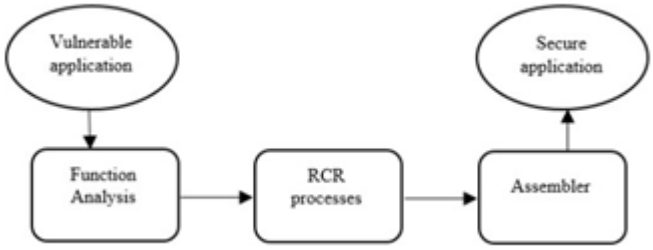

Fig. 7: RCR Approach

No special hardware is needed to implement this method and can prevent SSA against canary stacks with negligible overhead.

Fig. 7 shows the RCR method steps, and in the RCR Processes section, the safe code is prepared and the region of random canaries stacked in the memory is used. Using the time, date and application ID, random values are created and stored in the RCR, which can then be copied three times and stored in the heap's memory. It will prohibit attackers from accessing or changing the read-only RCR with only "const" variables, and it does not need to be referenced in the stack since the RCR is a global variable.

Isolation

Secure sensing: A sensor design has been proposed based on hardware isolation, which will protect the sensors on a device from a compromised application. This is achieved by using the hardware isolation feature of the ARM processor and by installing a sensor IP in the isolated region that is shielded from compromised applications.

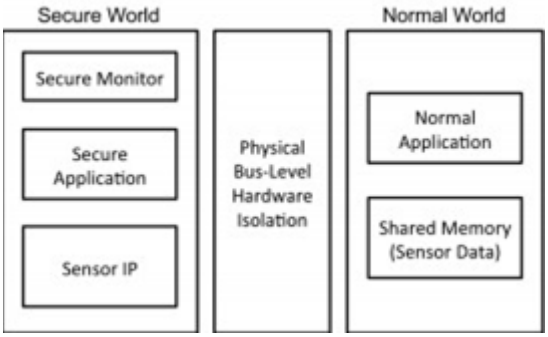

Fig. 8: Secure sensing overview

When required, we are able to exchange sensor data by installing shared memory in the normal world that can be accessed by the safe world. The application will then request a protected monitor in a secure environment, which will then switch the application mode from the normal world to the secure world to verify that the request has been accepted. The senor data may be written to the shared memory that can be read by the normal world if the request has been checked and verified for any potential risks.

SecPage is a lightweight hardware and software architecture that aims to secure computer memory by providing an independent memory environment that safeguards sensitive code and data. This security approach provides a safe, isolated and trusted environment for compromised system software to prevent unauthorized users from purchasing data and accessing code, but also provides easy access to pages that do not need to be secure, minimizing the architecture's overhead implementation.

IoT Security and Challenges

Our ways of living have been modified by the IoT period. While the IoT provides considerable benefits, it is vulnerable to numerous security threats in our daily lives. The majority of security threats are associated with information leaks and lack of operation. The physical security risk in the IoT is directly affected by security threats. The IoT consists of various devices and platforms with different credentials, where the security requirement is needed for every system, depending on its features. The privacy of a consumer is also the most important aspect of sharing a lot of personal data between various device types. Therefore, to protect sensitive information, a secure system is necessary. In fact, for IoT services, there are several kinds of devices that use various networks to communicate. This suggests that with the user privacy and network layer, there are a number of security concerns. It's also possible to disclose user privacy from different paths. Some protection threats in the IoT include the following:

- **E2E Data life cycle protection:** To ensure the security of data in the IoT environment, end-to-end data protection is supported on a full network. Data is collected from different devices connected to each other and instantaneously shared with other devices. In the complete data life cycle, it also includes a data protection, data confidentiality and information privacy management

scheme.

- **Secure thing planning:** The interconnection and interaction between the IoT devices are different depending on the situation. Therefore, systems must be capable of retaining a degree of security. For example, if local devices and sensors are used on a home-based network to communicate securely with each other, their communication with external devices should be based on the same security policy as well.
- **Visible/usable security and privacy:** Several of the security and privacy concerns are due to mis-configuring users. Executing certain privacy policies and complex security mechanisms is very difficult and impractical for users. It is important to pick security and privacy policies that should automatically apply.

IoT Challenges

The biggest IoT barrier is the security dilemma. Information on the IoT application may be manufacturing, corporate, consumer or personal. This knowledge about the application should be protected from fraud and tampering and must remain confidential. The IoT apps, for instance, will store a patient's health or shopping store results. The IoT increases device-to-device compatibility, but scalability, usability and response time problems still remain. Security is a concern where data is securely transmitted over the internet. Government legislation such as the Health Insurance Portability and Accountability (HIPA) Act can implement the Safety Measure Act when transporting data across international borders. Among the different security issues, the most important IoT-related challenges are discussed.

- **Data Privacy**: Some smart TV manufacturers collect data about their customers in order to analyze their viewing patterns, so that data collected by smart TVs will question the privacy of data during transmission.
- **Data Security**: Data security is still a big concern, too. Hiding from monitoring devices on the internet while seamlessly transmitting data is important.
- **Insurance Concerns**: In order to take insurance decisions, insurance companies gather health and driving status data by stalling IoT devices on vehicles.
- **Lack of Common Standard**: Since there are many norms for IoT goods and IoT development industries. Therefore, a major obstacle is the distinction between permitted and non-permitted internet-connected devices.
- **Technical Concerns**: Due to the increased usage of IoT devices, the traffic generated by these devices is also increasing. There is also a need to increase network capacity, so it is also a challenge to store the enormous amount of data for review and further final storage.
- **Security Attacks and System Vulnerabilities**: In the IoT security situation, until now, there has been a lot of work completed. The related work can be split into machine safety, device security, and network security.

Conclusion

The main focus of this paper, in particular, is to highlight major IoT security concerns, focusing on security threats and their countermeasures. Because of a lack of security mechanisms in IoT devices, many IoT devices become soft targets, and even this is not in the awareness of being exploited by the victim. In this text, security requirements such as confidentiality, integrity, and authentication, etc. are discussed. Twelve distinct types of attacks are listed in this study, along with their nature/behavior, as low-level attacks, medium-level attacks, high-level attacks, and extremely high-level

attacks, as well as suggested ways to deal with these attacks.

Given the importance of security in IoT applications, the implementation of security mechanisms on IoT devices and communication networks is really relevant. Furthermore, it is also recommended that devices not use default passwords to protect against any intruders or security threats and to read the devices' security specifications before using them for the first time. By eliminating features that are not used, the chances of security risks can be reduced. Moreover, it is vital to research different security protocols used in IoT devices and networks.

References

1. Y. Alkabani and F. Koushanfar. 2007. Active Hardware Metering for Intellectual Property Protection and Security. In USENIX Security. 291–306.

2. ARM. 2013. Cortex-M0 Processor. (2013). h1ps://www.arm.com/products/processors/cortexm/cortex-m0.php

3. J.P. Baukus, L.W. Chow, R.P. Cocchi, P.O., and B.J. Wang. 2012. Building Block for a Secure CMOS Logic Cell Library. (2012). US Patent no. 8111089.

4. J.P. Baukus, L.W. Chow, R.P. Cocchi, and B.J. Wang. 2012. Method and Apparatus for Camoufiaging a Standard Cell based Integrated Circuit with Micro Circuits and Post Processing. (2012). US Patent no. 20120139582.

5. A. Baumgarten, A. Tyagi, and J. Zambreno. 2010. Preventing IC Piracy Using Reconfigurable Logic Barriers. IEEE Des. Test. Comput. 27, 1 (2010), 66–75.

6. R. Brayton and A. Mishchenko. 2010. ABC: An Academic Industrial-strength Verification Tool. In International Conference on Computer Aided Verification. Springer, 24–40

7. F. Brglez, D. Bryan, and K. Kozminski. 1989. Combinational Profiles of Sequential Benchmark Circuits. In IEEE International Symposium on Circuits and Systems. 1929–1934.

8. Q. Chen, A. M. Azab, G. Ganesh, and P. Ning. 2017. PrivWatcher: Non-bypassable Monitoring and Protection of Process Credentials from Memory Corruption A1acks. In ACM Asia Conference on Computer and Communications Security. 167–178.

9. Chipworks. 2012. Intel‘s 22-nm Tri-gate Transistors Exposed.

10. S. Chu and C Burrus. 1984. Multirate filter designs using comb filters. IEEE Transactions on Circuits and Systems 31, 11 (1984), 913–924.

11. S. Davidson. 1999. Notes on ITC’99 Benchmarks..

12. J. Diguet, S. Evain, R. Vaslin, G. Gogniat, and E. Juin. 2007. NOC-centric security of reconfigurable SoC. In IEEE First International Symposium on Networks-on-Chip. 223–232

13. C. Helfmeier, D. Nedospasov, C. Tarnovsky, J.S. Krissler, C. Boit, and J.P. Seifert. 2013. Breaking and Entering through the Silicon. In ACM SIGSAC Conference on Computer and Communications Security. 733–744.

14. F. Imeson, A. Emtenan, S. Garg, and M. V. Tripunitara. 2013. Securing Computer Hardware Using 3D Integrated Circuit (IC) Technology and Split Manufacturing for Obfuscation. In USENIX Conference on Security. 495–510

15. Maxim Integrated. 2010. DeepCover Security Manager for Low-Voltage Operation with 1KB Secure Memory and Programmable Tamper Hier- archy.

16. R.W. Jarvis and M.G. McIntyre. 2007. Split Manufacturing Method for Advanced Semiconductor Circuits. (2007). US Patent 7,195,931.

17. A.B. Kahng, J. Lach, W. H Mangione-Smith, S. Mantik, I.L. Markov, M. Potkonjak, P. Tucker, H. Wang, and G. Wolfe. 1998. Watermarking Techniques for Intellectual Property Protection. In IEEE/ACM Design Automation Conference. 776–781. A.B. Kahng, S. Mantik, I.L. Markov, M. Potkonjak, P. Tucker, Huijuan Wang, and G. Wolfe. 1998. Robust IP watermarking methodologies for physical design. Design Automation Conference (1998), 782–787.

18. M. Kammerste1er, M. Muellner, D. Burian, D. Platzer, and W. Kastner. 2014. Breaking Integrated Circuit Device Security ftrough Test Mode Silicon Reverse Engineering. In ACM SIGSAC Conference on Computer and Communications Security. 549–557.

19. R. Karri, J. Rajendran, K. Rosenfeld, and M. Tehranipoor. 2010. Trustworthy Hardware: Identifying and Classifying Hardware Trojans. Computer 43, 10 (2010), 39–46.

20. Kirovski and M. Potkonjak. 2003. Local watermarks: methodology and appli- cation to behavioral synthesis. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 22, 9 (2003), 1277–1283.