Carleton University
Systems and Computer Engineering Department
SYSC 5809
Winter 2023

Security in IoT

Project Report

Internet of Things

Instructor: Prof. Mohamed Ibn Kahla

04/09/2023

Report by:

Iman Alavi Fazel	101261338
Mahmoud Selim	101281147
Aziz Al-Najjar	101244840
Shakir Sayed	101250615

Abstract

The Internet of Things (IoT) industry has expanded significantly in recent years by adopting new technologies and integrating with existing ones. With this expansion, the vulnerability to security breaches has escalated, leading to greater financial implications and posing a direct risk to individuals' well-being. In this project, we summarized the security aspects of these systems and discussed the solutions proposed in the literature. We first explored what is called “security landscape”, which refers to the unique characteristics of IoT and the security implications they have. Subsequently, different security attacks and their mitigations have been presented. These threats can target various aspects of an IoT infrastructure, from the sensor devices and actuators to the cloud and backend services. To provide this information in an organized manner, different threats were analyzed relative to the 4-layer model of IoT. After examining different attack surfaces and their prevention mechanisms, the use of edge computing and blockchain-based technologies as emerging technologies were discussed. As case studies, the last section of this report contains information about some of the previous well-known security incidents that occurred in the field. These details include the security flaws that caused these security attacks, the damages they imposed, and the lessons learned from them. This report can be useful for any practitioners and developers in the field and can provide a generalized overview of the challenges and possible solutions.

Table of Contents

List of Figures

List of Abbreviations

IoT	Internet of Things
IoC	Internet of Computing
WSN	Wireless Sensor Network
DoS	Denial-of-Service
DDoS	Distributed Denial-of-Service
MITM	Man-in-the-Middle
VANET	Vehicular Ad-hoc Network
QoS	Quality of Service
ARP	Address Resolution Protocol
XSS	Cross-Site-Scripting
ASLR	Address Space Layout Randomization
MPE	Memory Protection Extensions
UART	Universal Asynchronous Receiver/Transmitter
TCP	Transmission Control Protocol
UDP	User Datagram Protocol

1. Introduction

The emergence of the Internet of Things (IoT) has captured the interest of numerous service providers, businesses, and other sectors, including Healthcare, Autonomous Vehicles, Smart Grids, Digital Agriculture, among others. By allowing devices to perceive, communicate, and make intelligent decisions, IoT has become widely adopted as a key component of the industrial revolution [1]. However, the integration of IoT in diverse areas has also made security breaches in IoT more costly and directly impacting people's lives. Therefore, it is crucial for professionals in the field to understand the security threats in IoT, their mitigation, and best practices. Compared to traditional systems, securing IoT infrastructure is challenging due to the various technologies and devices involved, which leads to more points of failure, making IoT infrastructure more susceptible to security breaches [2].

This project aims to explore different aspects of IoT security, including general definitions and concepts, recent advancements, and major incidents that occurred over the years. Through this study, we aim to understand the IoT security landscape and identify various attacks on IoT layers, including the sensing, network, middleware, and application layers, and analyze their impacts on IoT security. Afterward, we will introduce IoT security solutions using edge computing and blockchain. It is crucial to comprehend these security threats and their mitigation techniques to develop secure IoT systems. Practitioners in the field can use this study's findings to develop and deploy robust security solutions that can protect IoT infrastructure against these attacks.

2. Security Landscape

IoT devices have characteristics that make them distinct from the conventional computational devices which are now also referred to as IoC (Internet of Computing). The main characteristics are [3]:

1. Limited memory and computational power: IoT market favors inexpensive devices [4]. Subsequently, a large class of devices being used in the industry have limited computational capabilities.

2. Non-Standard interfaces: IoT devices may use non-standard interfaces, adding new methods for exploitation in addition to the well-researched methods.
3. Cyberphysical interaction: IoT devices interact with the physical world, and consequently require new threat models.
4. The expectation of Long-Lived devices: Most IoT devices are designed to operate for a long period of time without being routinely updated and maintained.
5. Unclear authority: Many IoT devices are used by multiple users with varying levels of authority, which sometimes makes it difficult to manage access and enforce security policies.

In what follows, we will address the security implications of each layer individually, and briefly discuss some of the proposed solutions present in the literature.

2.1. Limited Memory and Computation Power

Low-cost IoT devices often have limited computational resources, making it difficult to run strong cryptographic algorithms [5]. This can result in weaker encryption methods that facilitate attacks such as *man-in-the-middle* attacks or *spoofing*. Manufacturers may also use open-source software or generic solutions to reduce costs, which may not be designed with high-security standards in mind. For example, a low-cost IoT camera may use an open-source operating system and lack firmware updates, making it vulnerable to cyberattacks such as data breaches. Moreover, attackers can exploit vulnerabilities in low-cost IoT devices to gain unauthorized access to the device as well as the network they are connected to. To address these security challenges, manufacturers should consider implementing stronger cryptographic algorithms and security features, such as secure boot and firmware updates, to better protect these devices and their users.

2.2. Non-Standard Interfaces

IoT devices often require alternative methods for setup, configuration, and usage, such as using a cloud management service, smartphone app, or voice commands. These novel interactions result in new attack surfaces that can be exploited by attackers. Some examples of these attacks are:

- IoT devices can take unintended sounds from the attackers as commands such as "Hidden voice commands". Hidden voice commands refer to inaudible messages that cannot be detected by the human ear, but IoT devices can understand. [7]
- Sensors such as temperature and noise can be exploited by attackers to generate false data or extract sensitive information from them (discussed in the Sensing layer section)

Manufacturers and providers should consider implementing robust security measures to secure non-standard interfaces, such as authentication mechanisms and encryption protocols.

2.3. Cyberphysical Interaction

IoT devices link the physical environment to digital space and compromising these systems means that we are impacting the physical world as well. For instance, actuator devices such as smart locks or thermostats directly interact with their physical environment. Consequently, these devices may also be vulnerable to physical attacks such as tampering, theft, or destruction, in addition to cyber-attacks, making it essential to adopt new security threat models [6]. To mitigate these risks, IoT devices must be secured against both cyber and physical attacks, including implementing physical security measures such as tamper-resistant hardware and secure enclosures.

2.4. The Expectation of Long-Lived Devices

Consumers expect their IoT devices to maintain their functionality and security for an extended period of time. The risk of potentially unpatched vulnerabilities is a significant threat to devices that do not receive regular software updates, especially because of the nature of "*set-and-forget*" of IoT devices. To solve this issue, first, out-of-date devices should be identified using a standard methodology and be fixed on-site by the providers [7]. Second, *push-based updates* should be used when applicable, where the update is initiated by the publisher or central server.

2.5. Unclear Authority

In IoC, roles are assigned to individual users based on the context of their interaction. In contrast, devices in IoT belong to an environment and are generally not linked to specific users, such as voice assistants and sensors, thus being a “many-user” device. The consequence of this new attack surface is that it is a challenge for IoT systems to distinguish between legitimate and unauthorized users. Moreover, users may inadvertently compromise the security of the device through their actions, such as sharing passwords or clicking on malicious links. In general, the many-user environment, solving the authorization problem remains a challenge, and there is no agreed-upon solution in the literature.

3. Security Threats

In the previous section we discussed the characteristics of IoT devices and the general implications they have on security. In this section, we will explore specific methods which can be used to target IoT infrastructure and some of the proposed solutions. The attacks that target IoT devices and networks are often classified according to the impact they have on the multi-layer architectures that are proposed for IoT. In this project we consider the four-layered model which is depicted in Figure 1.

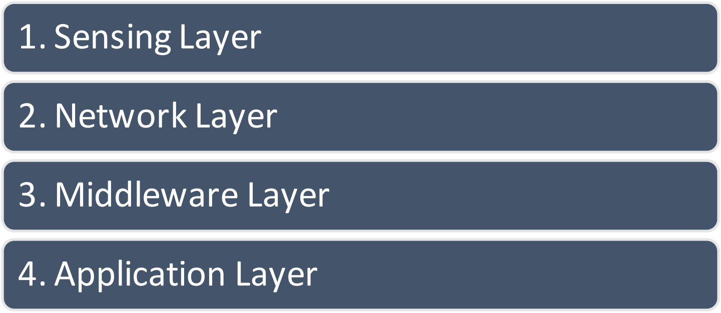

Figure 1. 4-layered architecture of IoT

The role and description of each layer is provided below:

1. **Sensing Layer**: Comprised of sensor and actuators.
2. **Network Layer**: Technologies, protocols and devices that are used to send information from the sensing layer to the computational unit for further processing.
3. **Middleware Layer**: Database servers, processing units and other devices that help resource allocation, computing, and data storage.
4. **Application Layer**: The layer that the end-users interact with, such as a Smart Home

application.

3.1. Sensing Layer Threats

Common attacks that target the sensing layer are as follow [8] [9] [10]:

1. **Node capturing attack:** An intruder physically captures the sensors and extracts the encoded information, such as cryptographic keys and other sensitive information. The attacker can use this information to set up a malicious node as an authorized node to connect to the IoT network or system.
2. **Malicious code injection:** Injection of malicious code directly to the sensors is a common threat targeting the sensing layer. This attack is performed, for instance, by the "*over the air update*" feature present in some IoT nodes. An attacker can force a node to behave unexpectedly or use it to affect other nodes in the network.
3. **False data injection:** An attacker alters sensor readings in a way that undetected errors are introduced into the system. The consequences are:
   1. Deviation of the application from expected behavior by reading erroneous data.
   2. DDoS (Distributed Denial-of-Service) of services that gather the sensors' data.
4. **Sleep deprivation attacks:** Disrupt nodes' scheduled sleep patterns and keep them constantly awake until they run out of energy.
5. **Side-channel attacks:** In contrast to direct attacks that target the system or its code, side-channel attacks use indirect methods to exploit the targets. An example of this attack is the *timing attack*, in which an attacker can obtain encryption keys by analyzing the timing information required for the execution of encryption algorithms.
6. **Eavesdropping and interference:** Most devices in the IoT communicate over wireless networks and are vulnerable to interception of both data transmission and authentication. Attackers can also send noise data for interference.
7. **Replay attacks: Intercepting** sent authentication or access control and retransmitting it with the goal of causing an unauthorized effect or gaining unauthorized access.

To mitigate the attacks on the sensing layer, different hardware-based and software-based techniques have been proposed and are presently deployed in IoT configurations. *Code authentication schemes* is a practical method in which the algorithms can detect any change in memory contents, hence detecting malicious code or unexpected activities [11]. *False data filtering algorithms* is another method which can use probabilistic methods to detect anomalies in sensor values [12]. *Secure time stamps* can prevent replay attack by verifying the time in which the traffic gets generated [13]. Lastly, when applicable, *energy harvesting* is a technique in which we provide external sources of energy to devices for their operation. Consequently, the sensors and actuators will not be drained out of energy by sleep deprivation attacks [14].

3.2. Network Layer Threats

From [8] [9] [10], we can summarize the common attacks that target the network layer are as follows:

1. **Routing attacks:** Routing attack is a general technique where the adversary redirects the communication channels during data transmission. There are different methods to execute this type of attack, and the most common ones are:
    a. **Sinkhole attack:** The adversary creates an artificial fast path that lures nodes' traffic to use them, presenting them as the optimal communication channel. Examples of this attack in Wireless Sensor Network (WSN) are MintRoute and TinyAODV protocols [15]. In the MiniRoute protocol, each node holds a table comprising of neighboring nodes with their link qualities. The attacker can advertise an optimal but false link quality for itself. Hence, changing the base station for a victim node in WSN.

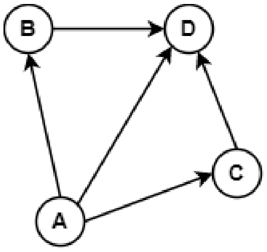

Figure 2. Nodes in a Wireless Sensor Network.

For instance, the link quality of neighboring nodes of A before the attack could be:

Node	Link Quality
D	170
B	**50**
C	130

And after the attack by node B:

Node	Link Quality
D	170
B	**255**
C	130

Therefore, the base station for A will change to B instead of the valid one, D.

b. **Wormhole attack:** This type is mostly performed in VANETs network and refers to a situation where two or more malicious nodes are present, and data packets are being transmitted from one malicious node to another instead of the standard route. An example of this attack is when the two malicious nodes are connected via a wired medium such as ethernet in a WSN, as depicted in Figure 3.

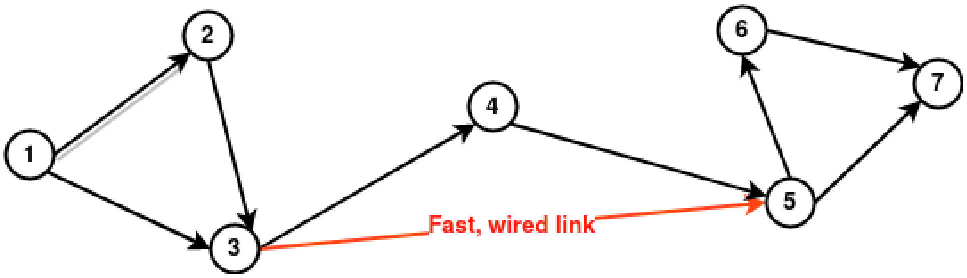

Figure 3. Wormhole attack example

2. **DoS/DDoS attack:** Denial-of-Service (DoS) is done by flooding and overwhelming target servers with unwanted requests/traffic so that the server is unable to respond to legitimate requests. It disrupts the server's ability to communicate with a genuine node resulting in disrupted service quality. If multiple sources are used to launch the attack, then the attack is called Distributed Denial-of-Service (DDoS) attack. SYN flood and UDP Flood are the most widely used attacks.
3. **Sybil attack:** A malicious actor creates multiple fake identities, also known as "sybil nodes" or "sybil devices," and uses them to impersonate legitimate devices on an IoT network. This way, the attacker can gain control over the network, disrupt its operations, or steal sensitive data. Consequently, normal users will either receive limited access to resources, or in the case of blockchain-based technologies carry out *51% attack*, in which the adversary manipulates the records or disrupts their operations.
4. **Unauthorized access:** Due to the absence of proper authentication mechanisms, many RFID tags are vulnerable to unauthorized access, which enables an adversary to obtain, modify, or delete the information stored in them. There is still active and on-going research on securing these types of devices.

To defend against the sinkhole attack, techniques such as *secure multiple routing protocols* were proposed and applied. Wormhole attack can be prevented by modifying the routing protocols to enhance the security in the route selection process, while other techniques involve deploying additional hardware such as GPS, antenna, etc. Securing networks against DoS/DDoS requires a combination of approaches. The main ones are:

1. Rate Limiting: Limiting the rate at which devices can send requests through the

network.

2. Network Segmentation: Grouping IoT devices into different segments and isolating them from other ones.
3. Device Authentication: Only authenticated devices may join the network.
4. Traffic Filtering: Unwanted network access should be filtered.
5. Cloud-based Protection: Advanced cloud infrastructure can detect abnormal traffics e.g., with the usage of AI/ML, and block them before reaching IoT devices.

3.3. Middleware Layer threats

Common attacks that target the middleware layer are as follows [8] [9] [10]:

1. **Man-in-the-Middle (MITM):** The adversary eavesdrops on the communication between two targets. This attack is commonly carried out via:
   a. ARP spoof: Attackers exploit Address Resolution Protocol to map their MAC address to the IP address of a target host, becoming the man-in-the-middle of the network traffic.
   b. DNS/mDNS spoof: Attackers create fake DNS (Domain Name System) or mDNS (Multicast Domain Name System) records to redirect network traffic to a malicious website or server, exposing sensitive information to the attacker as the man-in-the-middle.
   c. Rogue access point: Unauthorized access points, such as rogue hotspots, can be used to intercept network traffic, allowing attackers to steal sensitive information or infect devices with malware.

   In the context of IoT, the MQTT protocol uses a publish-subscribe model for communication between clients and subscribers. The MQTT broker acts as a proxy, decoupling the publishing and subscribing clients. If the attacker gains control of the broker, they can act as a Man-in-the-middle, gaining complete control of all communication.

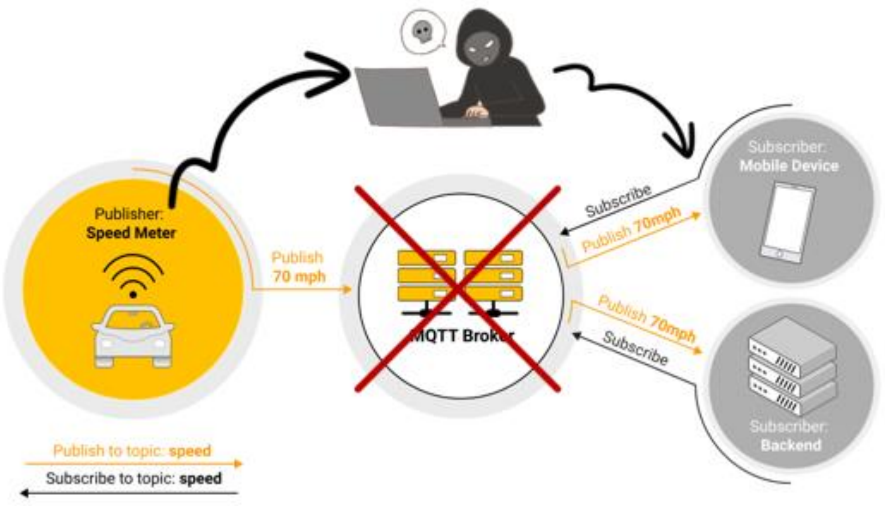

Figure 4. Man-in-the-middle on the MQTT broker.

2. **SQL Injection:** SQL Injection is a type of security vulnerability that allows an attacker to inject malicious SQL code into an application's backend and compromise the underlying database. In IoT, the middleware layer is vulnerable due to its function as the communication channel between devices and the database. For instance, in an insecure backend system that takes username/password for authorization, an attacker can provide:

   Username: Name ="" or ""=""
   Password: Name ="" or ""=""

   Which causes the following command to execute if the backend system does not perform input validation:

   *SELECT * FROM Users WHERE Name ="" or ""="" AND Pass ="" or ""=""*

   This statement causes all usernames and password to be return to the attacker.

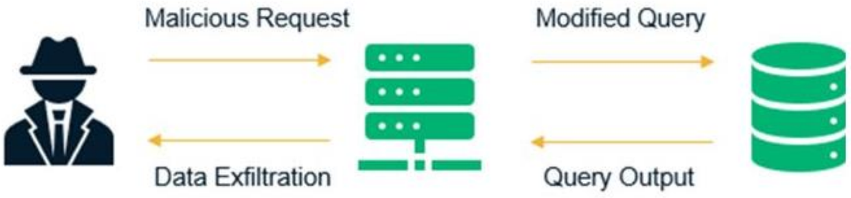

Figure 5. SQL Injection.

3. **Signature Wrapping Attack:** SOAP is a widely used protocol in IoT for exchanging data which defines an XML syntax and their processing rules. In the signature wrapping attack, the attacker breaks the signature algorithm and can modify the captured messages in the SOAP protocol without breaking the included signature. The attacker alters a message's digital signature, making it appear as though it was sent by a trusted source.
4. **Cloud Malware Injection:** A major threat which targets the Infrastructure as a Service (IaaS) clouds is its exposure to malware. This attack is carried out by setting up a valid service which users interact with. Consequently, the adversary can gain access to sensitive information of the IoT users, as well as to disrupt their services.

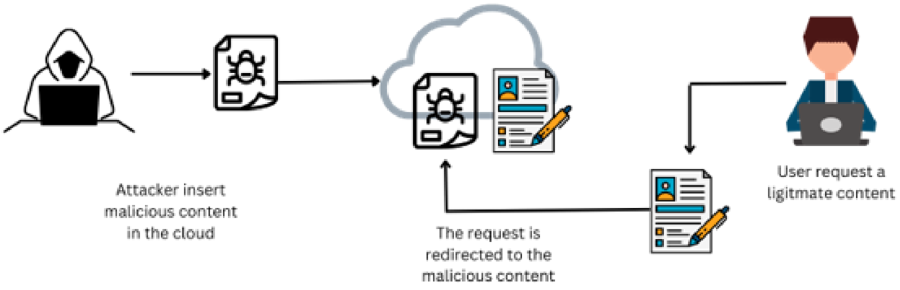
Attacker insert malicious content in the cloud
The request is redirected to the malicious content
User request a ligitmate content

Figure 6. Cloud Malware Injection

5. **Flooding Attack in Cloud:** Cloud services, similar to services in the network layer, can be a target of DoS attacks. In this scenario, the services employed on the cloud receive a flood of requests from the adversaries which affects the QoS (quality of service) and the load on the servers. Flooding a targeted machine or application, or service with lots of requests until regular traffic becomes challenging to process results in denial-of-service to other requesters.

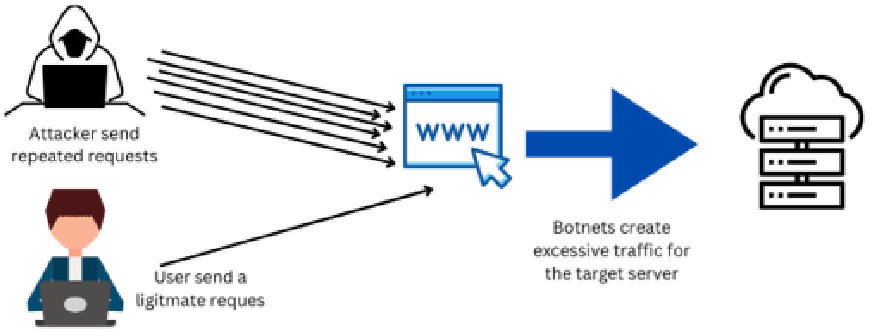
Attacker send repeated requests
User send a ligitmate reques
www
Botnets create excessive traffic for the target server

Figure 7. Flooding attack in cloud

The use of sophisticated firewall can mitigate the discussed attacks. Additionally, for SQL Injection, parameterized queries, input validation and following secure software artifacts can be employed. Encryption communication can further enhance the security of this layer, especially against MITM attack. Lastly, signature wrapping attack can be prevented with Proper XPath expression filtering.

3.4. Application Layer threats

Common attacks that target the application layer are as follow [8] [9] [10]:

1. **Sniffing Attack:** Like Man-in-the-middle, this attack refers to the intercepting of the traffic by the adversary. However, in contrast to MITM, a sniffing attack is passive and does not change the network traffic.
2. **Malicious Virus/Worm:** An attacker can introduce malicious self-propagating attacks, such as worms or Trojan horses, into IoT applications to gain access to or manipulate sensitive data. Effective defensive measures such as strong firewalls and virus detection systems should be implemented to prevent and combat these types of attacks in IoT applications.
3. **Access Control Attacks:** Access control involves allowing authorized individuals to handle authenticated entities. This form of attack on the application layer is particularly effective, because once the access to the software in the application layer is compromised, then the complete IoT application may become vulnerable.

Common methods include:

a. Buffer overflow
b. Cross-site-scripting (XSS)
c. Password attacks (e.g., brute force, dictionary attacks)
d. Backdoors
e. Social engineering

The mitigations techniques that are applicable to this layer include a diverse approach ranging from enhancing the application code base to employing OS-level, compiler-level and hardware-level mitigations. With *input validation* we can ensure that user inputs do not include special characters or patterns that could be used to inject malicious code. *Address space layout randomization (ASLR)* randomizes the location of program code and data in memory on the OS-level. *Run-time protection mechanisms* such as Memory Protection Extensions (MPE) can be used in hardware-level. Finally, advanced compilers can employ techniques such as Bounds checking to prevent memory-related vulnerabilities.

4. Edge Computing in IoT Security

4.1. Edge Computing

Edge computing is a type of distributed computing model that data is processed closer to the edge devices, rather than being transmitted to a centralized location. Edge computing architecture of the IoT system mainly consists of three layers, terminal layer, edge layer and cloud layer as shown in Figure 8. In this type of computing model, there is no necessity to send all data collected from terminal nodes to the cloud platform since distributed local computing and controlling can be conducted in the edge layer.

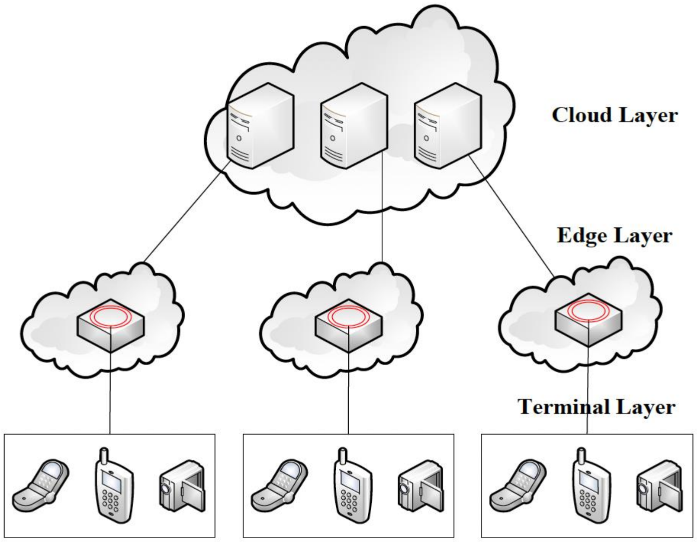

Figure 8. Edge computing architecture [16]

4.2. Edge Computing based IoT security.

There are several ways that edge computing can provide enhanced security for an IoT infrastructure. Some of these techniques are discussed in the subsequent sections [16].

4.2.1. Secure Boot

The secure boot process ensures that only trusted software is executed on an edge device. Here, the device firmware is verified by the edge layer before booting up to prevent any unauthorized access [17].

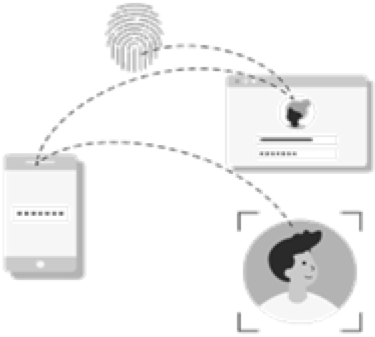

Figure 9. Secure boot in edge computing

4.2.2. Identity and Access Management

In an action called identity and access management (IAM), the edge layer can act as a mediator which inspects requests and performs authentication and authorization ensuring only valid users or devices can access the data or control the devices.

4.2.3. Data Encryption

Data encryption can be used to secure data at rest and in transmit between edge devices and the cloud. Encryption keys can be stored on the edge devices to ensure that data is encrypted at the source and remains encrypted even when transmitted to the cloud.

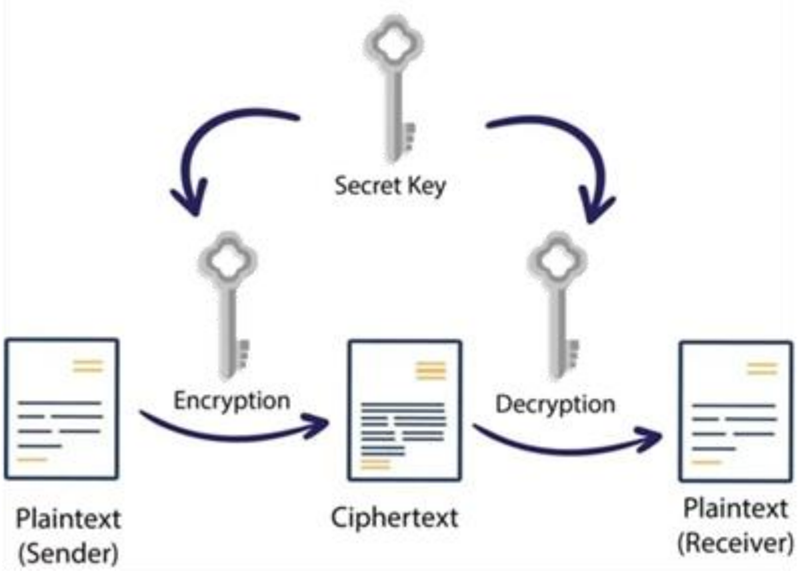

Figure 10. Data encryption in edge computing

4.2.4. Intrusion Detection Systems

An intrusion detection system (IDS) is a security software that observes network traffic and system events to detect any indications of unauthorized access or malicious activity. An instance of these systems can be deployed at the edge layer to provide additional real-time and on-premises security features.

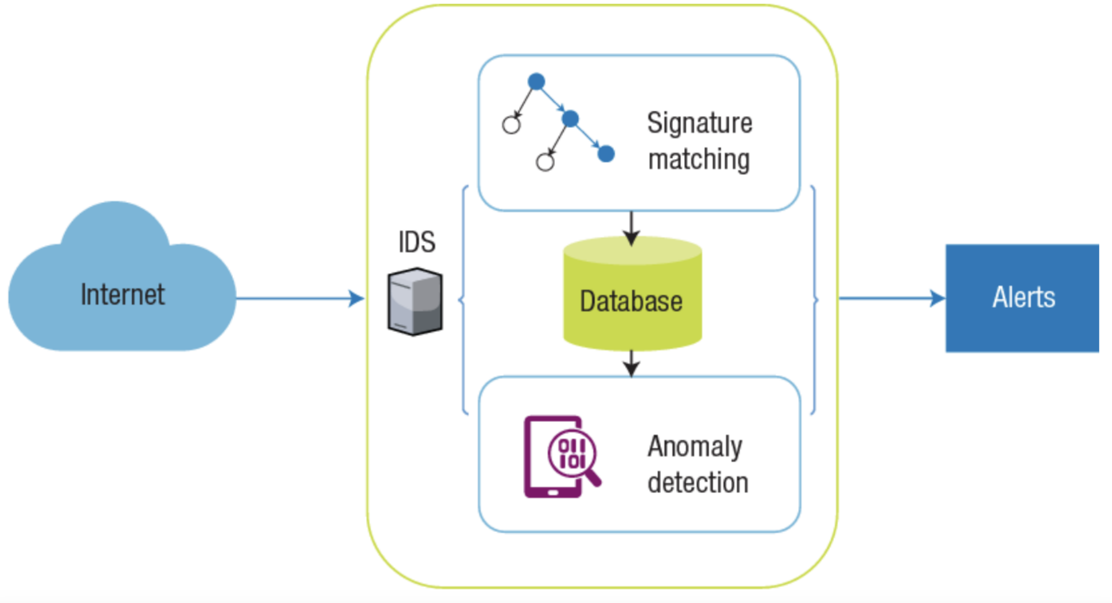

Figure 11. IDS in edge computing

4.2.5. Threat Intelligence

Edge devices can be protected by regularly updating threat intelligence feeds to identify and prevent new threats. This can help to detect and mitigate potential attacks before they can cause damage.

4.2.6. Physical Security

Physical security measures can be used to protect edge devices from physical tampering or theft. This can include using security cameras, access control systems, and alarm systems.

5. Blockchain-based IoT security

5.1. Blockchain Technology

Blockchain technology is a database system that enables transparent and secure sharing of information among members of a network. In this technology, the blocks of data are linked together in a series of connected entities in a tamper-proof and decentralized fashion. An example shown in Figure 12 shows an overview of the steps taken place in these systems.

A secure transaction is performed in the following steps:

1. An IoT device initiates and broadcasts a transaction to the network.
2. The network then reaches the agreement of the new block creator.
3. All IoT devices insert a copy of the new block into their local ledgers.
4. If there is any smart contract, The transaction stored in the blockchain triggers a smart contract.
5. The IoT device carries out the transaction.

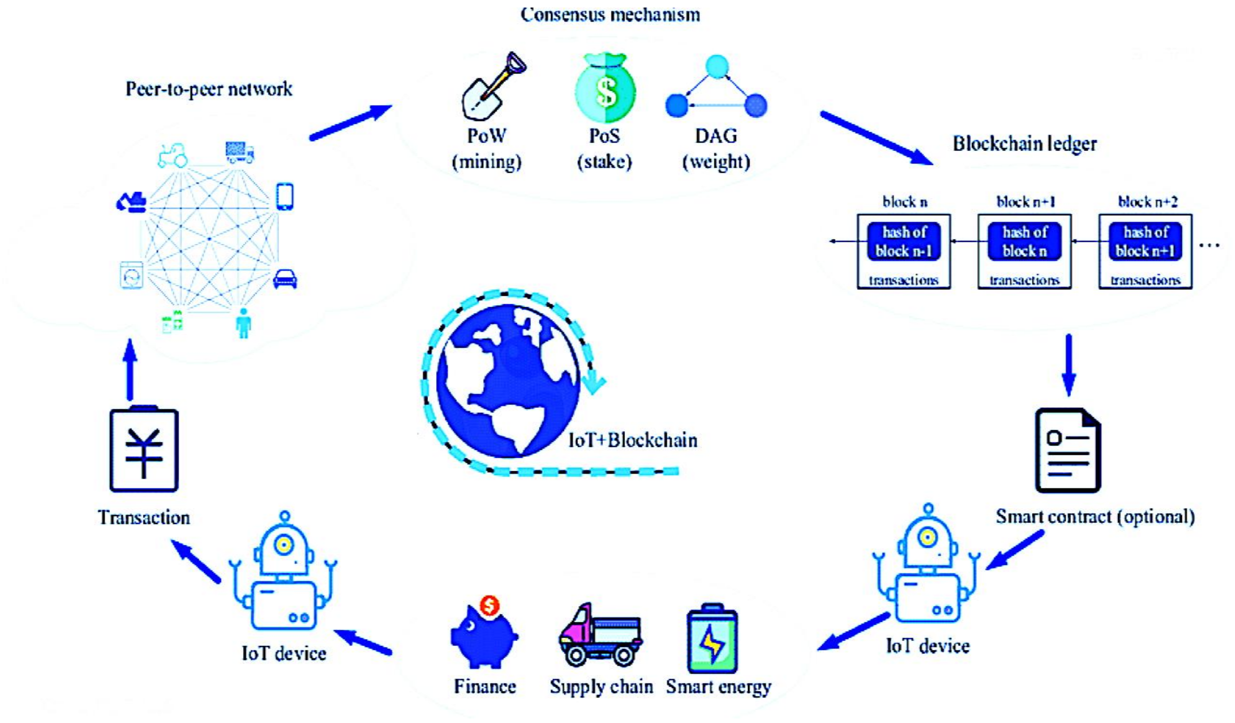

Figure 12. An example of Blockchain Technology [18]

5.2. Blockchain-based IoT Security

Blockchain technology is relatively new in the field of computing and there are many applications to the security of IoT systems yet to be researched and discovered. In what follows, some of the proven methods for this technology are discussed [19] [20].

5.2.1. Distributed Ledger

The data transfer between IoT devices can be secured using a distributed ledger. Since each transaction is recorded in a distributed environment, it makes it impractical for the attacker to modify the data from various decentralized locations. Blockchain-based solutions allow IoT data to be distributed, allowing for greater security, and reducing the risk of a single point of failure.

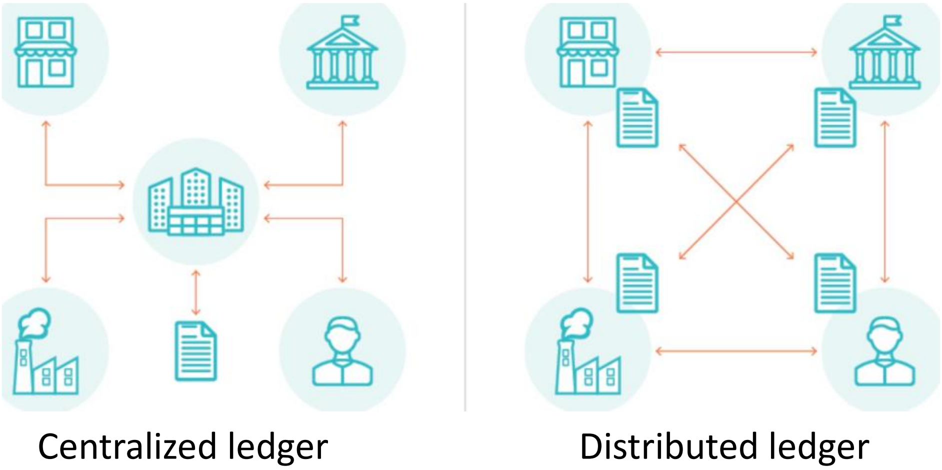

Figure 13. Distributed ledger in Blockchain

5.2.2. Immutable Records

Blockchain's immutable record ensures that once data is stored on the blockchain it cannot be modified. An application of this attribute provides a guarantee that the software that IoT devices run would remain unchanged and uncompromised.

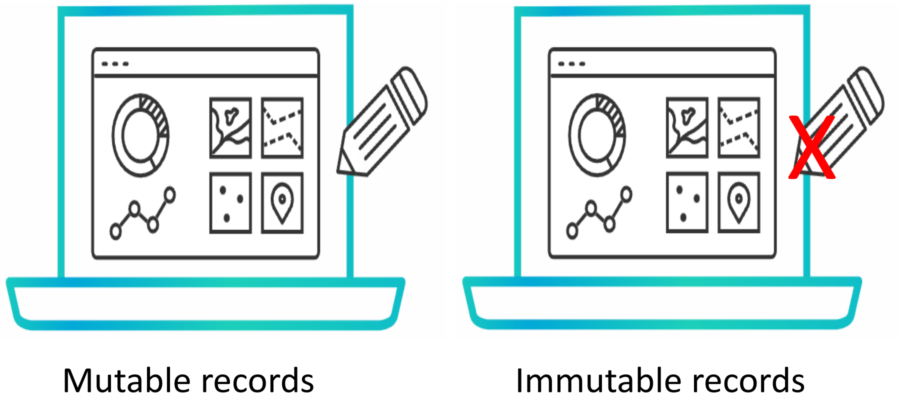

Figure 14. Immutable Record in Blockchain

5.2.3. Smart Contracts

By implementing smart contracts, the process of updating IoT device firmware can be automated and perform it securely and automatically.

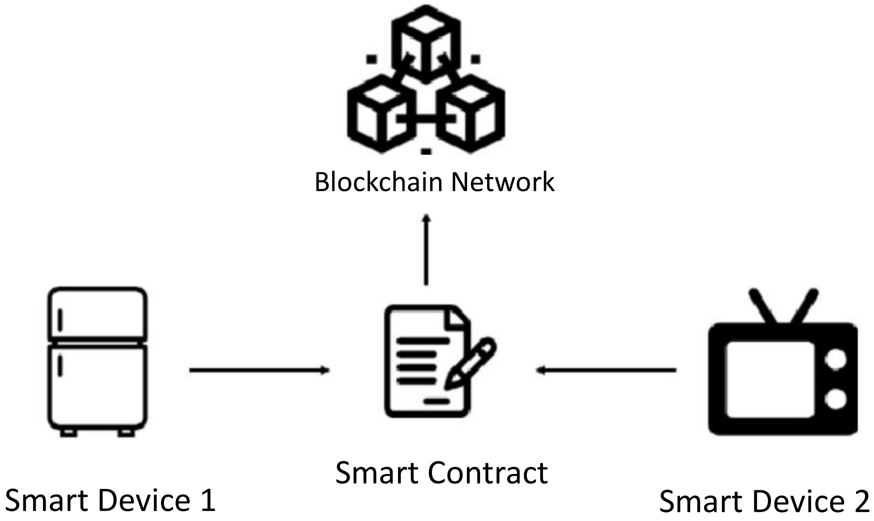

Figure 15. Smart contract in Blockchain

5.2.4. Identity Management

Blockchain-based identity management solutions can be used to authenticate IoT devices and ensure that only authorized devices are allowed to access the network. This can be done by assigning a unique digital identity to each IoT device, which can be verified by blockchain technology.

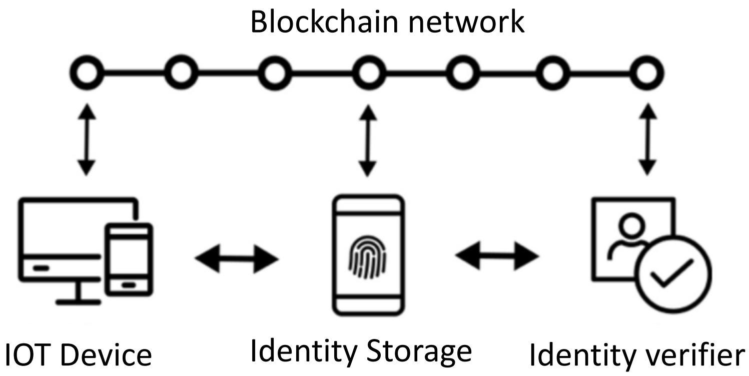

Figure 16. Identity management in Blockchain

5.2.5. Secure Communication

Secure communication between IoT devices can be achieved by encrypting data using public key cryptography and then using the blockchain network to store cryptographic keys involved in the communication.

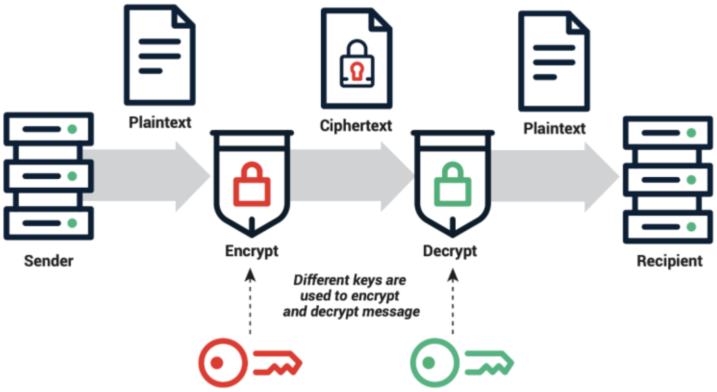

Figure 17. Secure communication in Blockchain

6. Case studies

6.1. Mirai Botnet Attack

The Mirai botnet incident, which occurred in late 2016, was a turning point in the history of IoT-related cyberattacks. This malware specifically targeted IoT devices, mainly IP cameras, DVRs, printers, and routers, to infect and control them for launching Distributed Denial of Service (DDoS) attacks [21]. The botnet scanned the internet using pseudorandom IPv4 network addresses and ten randomly selected username and password pairs to attempt a Telnet or SSH connection. It then used a preconfigured list of 62 credentials to establish a connection with the device.

The infected device was remotely managed by the control server, enabling the attacker to launch DDoS attacks and scan for new vulnerable devices concurrently. The Mirai botnet infected 600,000 devices by the end of its life cycle in February 2017. The damages caused by Mirai were significant, with the cost of $13.50 per device for consumers and around $4,207.03 per hour that the network bandwidth is overflown with traffic.

Several industries were affected, including game servers, anti-DDoS providers, telecommunication firms, political websites, and some suspicious sites hosted in Russia. The impact was much higher for domain name system (DNS) providers, which should have invested in DDoS countermeasures and network resource redundancy. The countries that were particularly impacted included the United States, the United Kingdom, Germany, France, Italy, Brazil, Australia, Japan, and China.

To avoid botnet attacks, randomized passwords in consumer electronics devices, avoiding default network configurations, applying address space layout randomization (ASLR), automatic updates under secure frameworks, and notification alerts when suspicious behavior is detected are encouraged. The adoption of immune-to-fragmentation attacks operating systems and withdrawal and upgrade of outdated systems can also impede the proliferation of botnet attacks.

6.2. Healthcare Devices

Healthcare infrastructure is part of the commercial IoT pillar and is vulnerable to cyberattacks [21]. The COVID-19 crisis has seen a significant increase of 50% more cyberattack incidents targeting the healthcare industry. Cyberattacks targeting medical IoT devices can range from confidentiality-type compromises to system-wide events that can harm patients' well-being. A hospital in Germany lost a patient in critical condition due to a ransomware attack that caused system access unavailability. More than 80% of the 1.2 million IoT devices located in US healthcare organizations were found to be operating using outdated OS, which do not support security updates anymore.

The majority of medical IoT devices lack potent cybersecurity features, paving the way for adversaries attempting to penetrate these mission-critical healthcare systems. Internet-connected medical systems are exposed to the public domain, such as anesthesia systems, cardiology systems, infusion systems, magnetic resonance imaging (MRI) machines, picture archiving and communication systems (PACSs), nuclear medicine, and pacemakers. Most of these systems are inadequately protected against critical vulnerabilities, such as default credentials, emergency account login, Telnet-root access, FTP-Admin, SSL key password manager, etc.

Attackers can exploit medical IoT devices with detailed information about the existing healthcare system and its exact location and functionality within the hospital network to execute physical attacks and mount device type of attacks. In another instance in 2015, researchers set up honeypot servers to research security attacks on the healthcare systems. The honeypots recorded 55,416 successful logins and 24 successful exploitations during their active period, and the most common exploit was MS08-067 which is a remote execution vulnerability present in Microsoft Windows Server 2003, 2008, XP, and Vista. Using the exposed medical device characteristics (e.g., models, connections, IPs, etc.), the honeypots were installed mimicking the healthcare organization architecture. Six months later, hackers discovered these counterfeit devices, exploited most of their vulnerabilities, gained access to various emulated medical devices, and even left malware on the honeypots.

Healthcare organizations should prioritize updating their outdated operating systems and implementing potent cybersecurity features on their medical IoT devices to prevent adversaries from penetrating into their mission-critical healthcare systems. Also, healthcare organizations should take steps to protect their systems from physical attacks and device-type attacks by limiting access to detailed information about the existing healthcare system and its exact location and functionality within the hospital network. Finally, the use of honeypots can provide healthcare organizations with detailed information on vulnerabilities, which can be used to improve cybersecurity measures.

6.3. Aircraft Avionics

The Boeing incident is another example of the potential risks of insecure IoT devices and highlights the importance of proper security measures to prevent attacks [21]. The discovery of an unprotected server on Boeing's network allowed hackers to download avionics system provider data and code intended for use on the company's 737 and 787 passenger jets. This incident demonstrates the potential consequences of unsecured IoT devices, as even a single vulnerability can lead to a series of destructive attacks.

Vulnerabilities discovered in some of the obtained .vex files include stack and buffer overflows, remote code execution, a vulnerable trivial file transfer protocol (TFTP) server, an insecure system-call handler, and return-oriented programming (ROP) exploits. The security analysis revealed many insecure code functions on the crew information system-maintenance system (CIS/MS) module that could potentially cause service-type attacks such as integer and buffer overflows.

Leveraging these vulnerabilities, two communication-based attack scenarios were depicted, targeting the network infrastructure of the 787 aircraft. The discovered vulnerabilities, including zero days, can be exploited leading to destructive attacks. Attack scenarios can attain unauthorized access to the common data network (CDN), which connects most of the airplane's systems.

The solutions to mitigate these attacks include the use of hardware mitigation and the adoption of compiler-level mitigation for insecure functions. Secure firmware updates are also recommended for mission-critical subsystems attested by integrity checks and controls to verify the authenticity of the firmware. However, Boeing and Honeywell stated that such vulnerabilities cannot be exploited, and security mechanisms are in place to counter them. Despite this, the incident highlights the ongoing need for continued research in aircraft cybersecurity, as aircraft security is far from a solved area of cybersecurity research.

6.4. Baby Monitoring Cameras

Baby Monitoring Cameras (BMCs) are widely used consumer electronic devices. However, a security analysis of a BMC revealed multiple vulnerabilities that could compromise users' privacy and allow unauthorized access to customers' local networks [21]. The vulnerabilities include the use of default passwords that are neither device-specific nor randomly generated, open TCP ports 554 and 5000 that can be exploited without authentication, and two Universal Asynchronous Receiver/Transmitter (UART) interfaces that can grant complete administrator control to anyone with physical access to the device.

Attackers can exploit the default-enabled peer-to-peer (P2P) cloud feature to compromise the device and violate users' privacy. Default credentials can lead to possible attacks targeting the service aspect of the device, locking authorized users out or providing adversaries with access to customer local networks via compromised devices.

The compromise of a BMC can provide adversaries with access to customer local networks and, as such, presents a risk to users' privacy. Researchers, manufacturers, and service providers recommend countermeasures to safeguard customer security, such as changing default credentials, disabling unused features, patching with the latest firmware updates, and enabling encrypted traffic.

Conclusion

Securing IoT infrastructure is an exceptionally challenging task due to the various technologies involved in these systems. This report provided a general overview of the security aspects of IoT systems and discussed the solutions proposed to address these challenges. First, by examining various attack surfaces and prevention mechanisms, the report emphasized the necessity of adopting a multi-layered security approach in IoT systems. We then explored the usage of emerging technologies such as edge-computing and blockchain technology, and some of the proven methods they can provide to enhance security. Lastly, the report concludes by providing case studies of prominent security incidents in the IoT industry, showcasing the security flaws, damages, and lessons learned from them. Security in IoT remains a challenge and there are still open areas for research. However, some of the methods provided in this study have been proven to be effective against security flaws and can be used by practitioners to implement in the system they design.

References

[1] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," in IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1125-1142, Oct. 2017, doi: 10.110.

[2] Mahmoud, Rwan, et al. "Internet of things (IoT) security: Current status, challenges and prospective measures." 2015 10th international conference for internet technology and secured transactions (ICITST). IEEE, 2015..

[3] Bellman, Christopher, and Paul C. van Oorschot. "Analysis, implications, and challenges of an evolving consumer iot security landscape." 2019 17th International Conference on Privacy, Security and Trust (PST). IEEE, 2019..

[4] Morgner, Philipp, and Zinaida Benenson. "Exploring security economics in IoT standardization efforts." arXiv preprint arXiv:1810.12035 (2018)..

[5] Hahm, Oliver, et al. "Operating systems for low-end devices in the internet of things: a survey." IEEE Internet of Things Journal 3.5 (2015): 720-734..

[6] Wolf, Marilyn, and Dimitrios Serpanos. "Safety and security in cyber-physical systems and internet-of-things systems." Proceedings of the IEEE 106.1 (2017): 9-20..

[7] Miettinen, Markus, et al. "Iot sentinel: Automated device-type identification for security enforcement in iot." 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2017..

[8] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications," in IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1125-1142, Oct. 2017, doi: 10.110.

[9] "Lin, Jie, et al. "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications." IEEE internet of things journal 4.5 (2017): 1125-1142.".

[10] Hassija, Vikas, et al. "A survey on IoT security: application areas, security threats, and solution architectures." IEEE Access 7 (2019): 82721-82743..

[11] "Seshadri, Arvind, et al. "SWATT: Software-based attestation for embedded devices." IEEE Symposium on Security and Privacy, 2004. Proceedings. 2004. IEEE, 2004.".

[12] "Yang, Xinyu, et al. "A novel en-route filtering scheme against false data injection attacks in cyber-physical networked systems." IEEE Transactions on Computers 64.1 (2013): 4-18.".

[13] "Cho, Chang-Hyun, et al. "Design of rfid mutual authentication protocol using time stamp." 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology. IEEE, 2009.".

[14] "Cammarano, Alessandro, Chiara Petrioli, and Dora Spenza. "Pro-Energy: A novel energy prediction model for solar and wind energy-harvesting wireless sensor networks." 2012 IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2012). I".

[15] Rassam, Murad A., et al. "A sinkhole attack detection scheme in mintroute wireless sensor networks." 2012 International Symposium on Telecommunication Technologies. IEEE, 2012..

[16] Sha, Kewei, et al. "A survey of edge computing-based designs for IoT security." Digital Communications and Networks 6.2 (2020): 195-202..

[17] Tiburski, Ramao Tiago, et al. "Lightweight security architecture based on embedded virtualization and trust mechanisms for IoT edge devices." IEEE Communications Magazine 57.2 (2019): 67-73..

[18] Bansal, Saloni, and V. K. Tomar. "Challenges & Security Threats in IoT with Solution Architectures." 2022 2nd International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC). IEEE, 2022..

[19] Alkurdi, Fahad, et al. "Blockchain in IoT security: a survey." 2018 28th International Telecommunication Networks and Applications Conference (ITNAC). IEEE, 2018..

[20] Shammar, Elham A., Ammar T. Zahary, and Asma A. Al-Shargabi. "A survey of IoT and blockchain integration: Security perspective." IEEE Access 9 (2021): 156114-156150..

[21] Xenofontos, Christos, et al. "Consumer, commercial, and industrial iot (in) security: Attack taxonomy and case studies." IEEE Internet of Things Journal 9.1 (2021): 199-221..

[22] Mahmoud, Rwan et al. “Internet of Things (IoT) Security: Current Status, Challenges and Prospective Measures.” 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST). Infonomics Society, 2015. 336–341. Web..