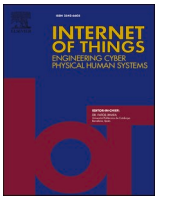
Internet of Things 19 (2022) 100564

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.sciencedirect.com/journal/internet-of-things



A survey on security in internet of things with a focus on the impact of emerging technologies

Phillip Williams [a,*], Indira Kaylan Dutta [b], Hisham Daoud [a,c], Magdy Bayoumi [a,c]

[a] Center for Advanced Computer Studies, University of Louisiana at Lafayette, LA, 70503, USA
[b] College of Engineering & Applied Science, Arkansas Tech University, Russellville, Arkansas, 72801, USA
[c] Department of Electrical & Computer Engineering, University of Louisiana at Lafayette, LA, 70503, USA

ARTICLE INFO

Keywords:
IoT
Security
Machine learning
Blockchain
Threats
Security solutions

ABSTRACT

Internet of Things (IoT) have opened the door to a world of unlimited possibilities for implementations in varied sectors in society, but it also has many challenges. One of those challenges is security and privacy. IoT devices are more susceptible to security threats and attacks. Due to constraints of the IoT devices such as area, power, memory, etc., there is a lack of security solutions that are compatible with IoT devices and applications, which is leading this world of securely connected things to the "internet of insecure things." A promising solution to this problem is going beyond the standard or classical techniques to implementing the security solutions in the hardware of the IoT device. The integration of emerging technologies in IoT networks, such as machine learning, blockchain, fog/edge/cloud computing, and quantum computing have added more vulnerable points in the network. This paper introduces a comprehensive study on IoT security threats and solutions. Additionally, this survey outlines how emerging technologies such as machine learning and blockchain are integrated in IoT, challenges resulted from this integration, and potential solutions to these challenges. The paper utilizes the 4-layer IoT architecture as a reference to identify security issues with corresponding solutions.

1. Introduction

The term Internet of Things (IoT) was coined by Kevin Ashton at a presentation he made at Proctor and Gamble in 1999 [1]. Since then, there has been a dramatic increase in the number of connected devices installed on the internet. A clear and concise definition of IoT found in literature is by the European Union Agency for Network and Information Security. IoT is defined as "a cyber-physical ecosystem of inter-connected sensors and actuators, which enable decision making [2]." The IoT concept has been a part of our lives for several decades before the term became popular in the early 2010s [3]. Before the 2010s several sectors in society were using IoT devices and applications, but on a small scale. IoT offers users the ability to integrate devices, data, and applications employing internet protocols.

The number of sectors implementing IoT applications is on the increase, which means the number of IoT devices and applications created will also increase. One such sector is consumer IoT, which is introducing wearable technology with sensors to monitor and transmit the activity and health data of a person. The healthcare industry is introducing IoT devices and applications [4] such as

* Corresponding author: Mr Phillip Williams, Center for Advanced Computer Studies (CACS), University of Louisiana at Lafayette, 301 E. Lewis Street, Lafayette, Louisiana 70503, United States
E-mail address: phillip.williams1@louisiana.edu (P. Williams).

https://doi.org/10.1016/j.iot.2022.100564

Available online 5 July 2022
2542-6605/© 2022 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

remote patient monitoring, hospital operations management, glucose monitoring, connected inhaler, connected contact lens, robotic surgery, efficient drug management, cancer detection, and augmented reality headsets and smart hearing aids to their patients. “Smart home” IoT devices and applications [5] available on the market today include smart door locks, smart heating, smart gardening, video doorbells, personal assistants to smart bulbs, smart coffee machines, and smart refrigerators. The "Smart city" sector has created IoT devices and applications that are used in smart parking, smart street lights, and smart waste management [6, 7]. Companies leading the charge of introducing new applications for industrial IoT are Alibaba Cloud in partnership with Siemens. They are working on industrial IoT operating systems [8]. DHL is a logistics company, working on IoT technology for supply chain businesses [9]. Konux is working on end-to-end IoT solutions for operation, monitoring, and predictive maintenance [10]. Nexiot is working on ultra-low-power embedded technology [11]. Other companies include, Scandid in Switzerland [12], Apple in the USA [13], Cognigy in Germany [14], Huawei in China [15], Samsung Electronics in South Korea [16].

The growing number of IoT devices available on the market is an indication of a successful IoT industry, but many of these devices suffer from resource constraints. As a result, classical security solutions are not applicable to many IoT devices and it is strongly required to provide the IoT devices with lightweight security solutions. [17] classifies security constraints as limitations based on hardware, software, and networking of IoT devices. Limitations based on hardware include computational, storage, power, and memory constraints. Limitations based on software include embedded software constraints. Limitation based on networking includes, mobility, scalability, slow intermittent network connections which is due to the implementation of low power radios which results in low data rates.

1.1. Related works and contribution

In this section we demonstrate some of the recently published work related to surveys on IoT security and privacy.

Hassija et al. [18] provided a detailed review of the security-related challenges and sources of threat in the IoT applications. The paper gave detailed and realistic recommendations to improve the IoT infrastructure to facilitate secure communications. Finally, the author discussed how existing and upcoming technologies such as blockchain, fog computing, edge computing, and machine learning can be used to increase the level of security in IoT. Similarly, Jurcut et al [19] discussed the problems related to safety and security in IoT. This is done by identifying general threat and attack vectors against IoT devices and also highlighting vulnerabilities that can lead to a breach of security. Additionally, this paper presented some solutions to compromised devices along with methods for prevention and security improvements to minimize risks.

In [20] Mishra et al. reviewed the evolution of IoT, applications, and challenges of IoT. A layered perspective was used to highlight the security issues faced in IoT. A comparison of anomaly detection techniques and the most recent Intrusion Detection System (IDS) was utilized to improve IoT security. Noor [21] presented information on recent research trends in IoT security from 2016-2018. This paper looked at relevant tools and simulators, outlined simulation tools, modelers, and computational and analysis platforms tools used by researchers in the field of IoT security.

HaddadPajouh et al. [22] discussed IoT security threats, challenges, limitations, requirements, and potential solutions based on a three-layer IoT architecture. Al-Garadi et al. [23] discussed Machine earning (ML) and Deep Learning (DL) methods that can transform IoT security from “facilitating secure communication between devices to security-based intelligent systems.” This paper discussed the potential vulnerabilities and attack surfaces of IoT systems. [24] Zaman et al. presented a survey on IoT security threats, along with Artificial Intelligence-based security models as a counter-measure to the security threats based on a layered IoT architecture.

Kouicem et al. [25] discussed the security benefits emerging technologies such blockchain and software defined network (SDN) bring to IoT networks. The main security benefits of these two systems are flexibility and scalability. The paper also looked at the security requirements and challenges in different IoT applications. Security solutions are categorized as classical and new approaches.

In [26] Harbi et al. analyzed IoT security based on a taxonomy of security requirements that included data security, communication security, and device security. For a list of IoT applications the paper discussed the challenges and proposed security solutions.

Hamad et al. [27] discussed security threats faced by IoT and the possible countermeasures. Identified the major security requirements needed to mitigate the security challenges of IoT such as resource constraints and IoT heterogeneous nature. The paper classified security solutions into security services such as access control, integrity, authentication and anonymity, confidentiality, and privacy. In [28] Thakor et al. reviewed lightweight cryptographic algorithms for constrained IoT devices. Lightweight cryptographic algorithms are classified in two main classes: symmetric and asymmetric. Hardware and software performance metrics for resource requirement are introduced. The paper gave a brief description of lightweight algorithms found in literature. The best lightweight algorithm is one that has a proper balance of performance, cost, and security.

Hameed and Alomary [29] reviewed lightweight encryption algorithms and authentication methods in IoT that mitigate several types of attacks. The authors proposed that more research is needed to enhance security in IoT devices. Lu and Xu [30] discussed the security attacks on IoT using the four-layered cybersecurity-oriented architecture for IoT and displayed a taxonomy of cybersecurity attacks on IoT. They discussed the application in different industries and countermeasures against attacks.

Although there are several of the works introduced as mentioned above on IoT security, they are specific to certain limited aspects of IoT. Therefore, there is the need for a more detailed surveys on aspects not covered such as the security challenges of integrating emerging technologies in IoT and security solutions through hardware to fit the resource-constrained IoT devices. The contribution of this work is listed below:

- Analysis on the integration of emerging technologies in IoT, challenges, and possible solutions.
- Present lightweight hardware security solutions as a viable option for constrained IoT devices.

- Review IoT security threats from several perspectives (i.e., hardware, software, and data in transit).
- Identify and provide a description of popular IoT security primitives along with other technologies used to protect devices and IoT networks from threats or attacks.

2. IoT security threats

Security threats can be classified into three groups. Threats in the form of attacks on the hardware (which involves the IC applications). Secondly, threats in the form of using malicious software to gain full control of devices. Finally, threats that capture and modify data in transit. These are shown in Fig. 1. A brief description of the three forms of common IoT security threats are listed below:

2.1. Hardware threats

1 Hardware Trojan – the attacker monitors, modifies, or disables either the data stored in the circuit or the communication of the circuit using trojan. This is done during the design or fabrication of the device. Figure 2 displays the general structure of a hardware trojan in a design [31]. In [32], hardware trojans are classified as:
2 Combinational/Sequential. Combinational: the trojan activation depends on the occurrence of a particular condition at certain internal nodes of the circuit.
3 Sequential: the trojan activation depends on the occurrence of a specific sequence of rare logic values at internal nodes.
4 Attributes. Physical, activation, and action.
5 Trigger and Payload Mechanism. The trigger mechanisms can be of two types: digital and analog.

Side Channel Attack – is when an attacker exploits the leakage of physical information from a system during the execution of an application. The adversary carries out non-invasive hardware-based attacks by monitoring and measuring power usage, electromagnetic radiations, timing information, and sound. The information gathered can be analyzed to extract private information such as cryptographic keys. Techniques used to perform a side channel attack are differential fault analysis [33], power monitoring attack [34], electromagnetic analysis attack [35], and acoustic cryptanalysis key extraction attack [36]. Figure 3 shows the architecture of a differential power analysis attack to extract secret data from the smart card.

- Tampering – occurs when an attacker alters the data associated with an IC after it is involved in an application. Most IoT devices will be placed in an environment without any physical safeguards to protect against an attacker gaining physical access to the device or wirelessly tampering with the software on the firmware of the device. The attacker can install malicious hardware or software to modify the behavior of an IC or the device.
- Denial of Service (DoS) or Distributed DoS (DDoS) – is when attackers meddle with the internal structure of an IC to block users from accessing the service.

2.2. Software threats

- Botnet – these are devices with malicious software installed and are connected to the internet. The lack of strong security solutions in resource-constraint IoT devices make them easy targets for cyber-criminals. The cyber-criminal can turn these devices into botnets that are completely under their control. Botnets are used by cyber-criminals to carry out phishing attacks, spamming, malware delivery, as well as Distributed Denial of Service (DDoS). Botnet architecture can be a peer-to-peer architecture, centralized architecture, or a hybrid of both architectures [37]. The basic botnet architecture is shown in Figure 4 below
- Spoofing – happens when an attacker impersonates a valid IoT device or authenticated user in order to gain access to a network. This is done by utilizing the Media Access Control address or Internet Protocol address of the legitimate user.
- DoS – is when attackers use computer(s) to flood or overload a target with massive amounts of messages or data, that result in a denial of service. Some of the more common DDoS attacks are user datagram protocol (UDP) Flood, Internet Control Messaging Protocol (ICMP), or ping flood. SYN flood attacks, ping of death, slowloris, NTP amplification, HTTP flood, and zero-day DDoS attack are examples of DDoS attacks as well [38]. Figure 5 provides a visual example of a DoS attack architecture [39].

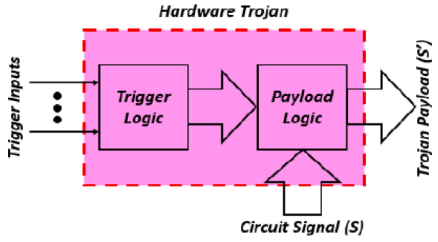


Fig. 1. General Structure of Hardware Trojan in a Design

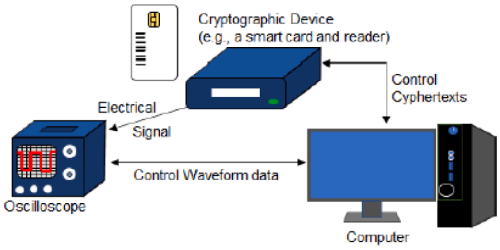


Fig. 2. Differential Power Analysis

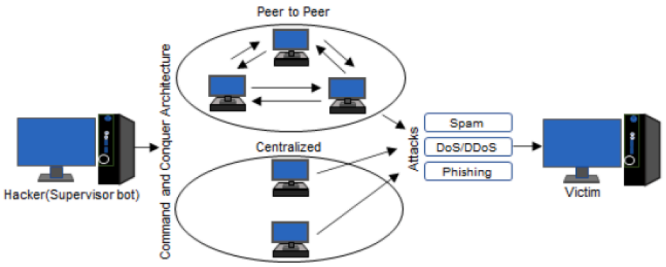


Fig. 3. Basic Botnet Architecture

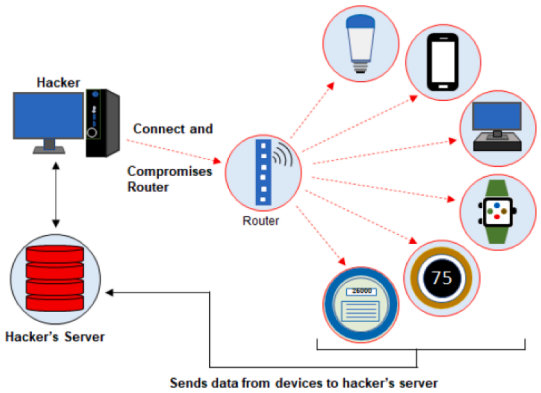


Fig. 4. Iluustrates an example of Spoofing

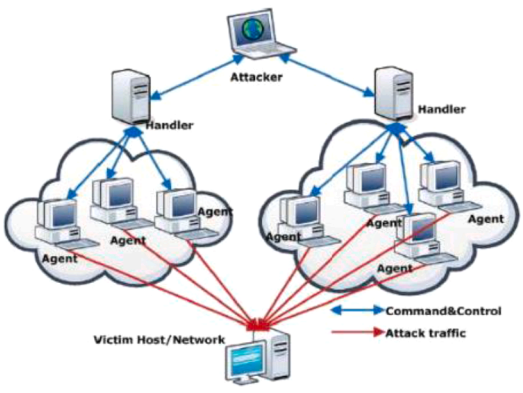


Fig. 5. Example of a DoS Attack Architecture

2.3. Data in transit

- Eavesdropping/Sniffing – is carried out by an attacker that uses a program or software (such as Wireshark) to capture data in transit. Hackers use programs that are developed to locate and record private data communications. The captured data is analyzed using cryptographic tools or in the case where a device lacks adequate security solutions, the attacker just needs to read or listen to obtain valuable information. Security attacks that utilize eavesdropping are Soundcomber and artificial intelligent assistants (e.g., Alexa, and Siri) [40]. Figure 6 is an architecture of the Soundcomber application for smartphones.
- Replay Attack – is when the attacker captures packets of information from an authenticated device, stores it, then delays or re-transmits it later as if the attacker is an authenticated device.
- Traffic Analysis – is examining captured network traffic to deduce useful information based on patterns in communication. Two types of traffic analysis attacks are link-load analysis attack that is used to discover the traffic rate on a network communication link and flow-connectivity attack with the goal of discovering the flow connectivity between a sender and a receiver [41].
- Man in the Middle Attack – is defined as an attack in which the attacker is located in the middle of the communication as a relay/proxy between a sender and a receiver. In this position, the attacker can intercept and alter the communications between the sender and receiver. Figure 7

2.4. Hardware threats

Table 1 displays reported security attacks based on the classifications of IoT threats [42, 18], along with a description of each attack.

3. IoT security solutions

The standard approach to dealing with different threats and attacks on IoT devices and IoT networks is to utilize the best encryption techniques to protect data at rest (data stored on a device) and data in transit (data sent over a communication link). Then we need proper authentication techniques to verify the identity of an entity requesting access to a network, device, or service. After that, we need to look at how to implement different security protocols from the perspective of the IoT security architecture layers (as described above). We also need proper security solutions to protect IoT devices and networks from sophisticated threats and attacks that may be application-specific.

3.1. Cryptographic solutions for data protection

Different types of cryptographic solutions are available to protect our important data, but unfortunately, not all of them are suitable for resource-constrained environments like IoT devices.

Both commercial and industrial IoT devices are vulnerable to IoT specific attacks [52]. If we continue to utilize the existing IoT device design flow where security is addressed as an afterthought, we will face many more security disasters in the near future. Lightweight cryptographic solutions are being researched with the goal of creating a strong cryptographic solution for constrained IoT devices.

- Lightweight Cryptography. The perceptual layer, network layer, and application layer require data encryption to securely protect the data generated and transferred by the layers. Among these three layers, the perceptual layer has all the constrained components, which demands a lightweight cryptographic system. There are two criteria used to determine the lightweight of a cryptographic algorithm [53]. The first criterion is the software weight of the cipher: this is determined by the cipher's time and memory complexities. Time complexity is the time it takes the cipher to turn plaintext into ciphertext and memory complexity is the amount of memory needed to carry out the task of creating the cipher. The second criterion is the hardware weight of the cipher: this is determined by the cipher's area and power consumption. The area of the cipher is represented by the number of gate equivalencies (GE) used to implement the cipher and power consumption is the power used during the cipher's execution. GE is determined by dividing the area (in micrometer squared: μm2) by the area of a two-input NAND gate. The algorithm needs to meet the lightweight standards whilst having a similar performance for security attacks and security standards when compared to conventional algorithms. The current cryptographic primitives can be divided into two categories: asymmetric key cryptography and symmetric key cryptography.

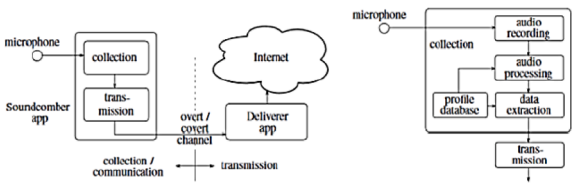


Fig. 6. Architecture of a Soundcomber Application

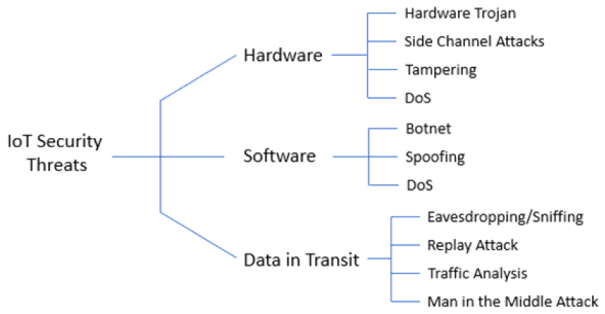


Fig. 7. IoT Security Threats

Table 1
Security Threats/Attacks on IoT with Recent Reported Attacks based on the Security Architecture of IoT

Security Threats/Attacks		Reported Attacks	Description
Hardware	Tampering	Jeep hack [43]	Hackers were able to exploit a vulnerability in the firmware update mechanism of the Jeep.
	Tampering	Voice-Controllable System [44]	Laser-based Audio Injection commands is used to obtain connection to devices such as a thermostat.
Software	Botnet DDoS	Malware attack [45]	BrickerBot used to permanently incapacitate IoT devices.
	Botnet	Silex malware [46]	Brickerbot wiped the firmware of 2000 IoT devices.
	Botnet DDoS	Mirai botnet [47]	This attack brought down the internet's domain name system infrastructure controlled by Dyn.
	Botnet	Malware attack [48]	VPNFilter botnet infects 500K network routers and network-attached storage devices.
Data in Transit	Traffic Analysis	Sybil attack on Tor Network [49]	Exploited a vulnerability in the Tor protocol to uncover the IDs of website operators using Tor hidden services.
	Eavesdropping/ Sniffing MITM Attack	Tesla Model S key fob attack [50]	Hacker is able to clone the key fob by wirelessly reading signals from its communication to obtain the fob cryptographic key.
	Eavesdropping/ Sniffing	Target's data br-each involving IoT HVAC system [51]	This breach compromised over 41 million customers' credit card information.

- Asymmetric Key Cryptography. Few lightweight cryptography researchers have been working on asymmetric cryptographic algorithms, but unfortunately the results are not yet steady and fruitful like symmetric cryptographic algorithms. Lightweight asymmetric cryptographic algorithms are complex in terms of operation, as a result these are often not area or power efficient. With the advancement of attack models, these algorithms are becoming vulnerable [54]. Asymmetric algorithms are typically based on trapdoor functions like, prime and semiprime factorization, and the Euler's totient function [55]. Asymmetric algorithms can be divided into encryption algorithms and key distribution algorithms. A prime example of an asymmetric encryption algorithm is Rivest-Shamir-Adleman (RSA). Elliptical Curve Cryptography (ECC), and Diffie-Hellman are representative examples of asymmetric key distribution algorithms. ECC, and Digital Signature Algorithm (DSA) work in the framework of public key cryptosystems to generate a digital signature. Figure 8

Rivest-Shamir Adelman (RSA). RSA is highly secure because the reverse procedure in RSA is very difficult for an attacker and producing the private key from the public key is also difficult, but key generation is complex, and the process is very slow [56].

Elliptical Curve Cryptography (ECC). ECC is more complex and difficult to implement, but it consumes relatively less power than other asymmetric algorithms. As a result, ECC is most favorable for implementation in constrained devices [57].

Diffie-Hellman (DH). RSA is highly secure because the reverse procedure in RSA is very difficult for an attacker and producing the private key from the public key is also difficult, but key generation is complex, and the process is very slow, making it vulnerable to man in the middle attacks [56].

Digital Signature Algorithm (DSA). DSA is relatively more beneficial and faster than other asymmetric algorithms. The process of

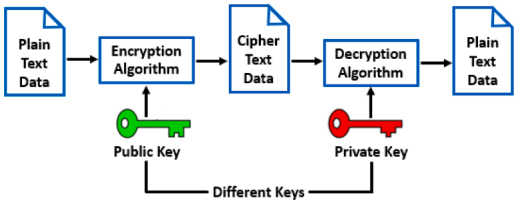


Fig. 8. Asymmetric Key Cryptography

sharing the signature is complicated and also the digital signatures have a short lifespan [56].

Of all the asymmetric algorithms, ECC consumes less power than other algorithms. This approach for implementation in IoT has recently become an important research topic, but mostly from a software perspective. An open source ECC for the Contici OS for IoT has been implemented and evaluated by [57]. It was released under the Berkeley Software Distribution (BSD) license. An ECC approach was applied by [58], where they implemented a Zero Knowledge Protocol in an open-source and generic programming library called Wiselib. [59] showed an ECC computation can be efficiently protected against side channel attacks. This approach followed the lightweight requirements with a minimal security level. [60] have shown a comparative study among RSA, Diffie-Hellman, and Elliptical Curve Cryptography with Diffie-Hellman (ECDH). According to their findings, ECDH performs better than other algorithms in terms of power and area. The results of evaluating public key cryptography are not complete as symmetric key cryptography.

- Symmetric Key Encryption. Symmetric cryptography algorithms do not use many resources and are faster in operations, because primarily these operations are based on bitwise functions such as XOR and permutations. As a result, these algorithms are more suitable for IoT applications [56]. A very important distinction in symmetric algorithms is among stream ciphers, hash functions, and block ciphers. Fig. 9 shows the typical architecture of symmetric key cryptography.

Stream Cipher. Stream Cipher. Trivium, Chacha, WG-8, and Espresso are few of the common lightweight stream ciphers with high throughput gain. Grain 128 is relatively more suitable as a lightweight cipher for constrained devices, but it has low throughput [61]. The encryption and decryption operation of a stream cipher is described in Figure 10.

Hash Functions. A good hash function will satisfy two basic properties, fast computing and minimizing duplication of output values. Some lightweight hash functions have been researched recently which could be suitable from an IoT perspective. [62] describe ways to use PRESENT block cipher in hashing modes of operation. Other examples [63] of lightweight hash functions in research are Spongent, PHOTON, and GLUON. Fig. 11 shows the typical architecture of a has function.

Block Cipher. This technique is very helpful for IoT applications. This process has almost symmetrical or identical encryption and decryption methods. Due to the Block ciphers have low latency of the block ciphers, as a result these are considered as the most researched and modified solutions for IoT security [61]. Advanced Encryption Standard (AES), Data Encryption Standard (DES), 3DES, Blowfish, Twofish are a few of the examples of different kinds of Block Ciphers. Researchers have been working on different approaches to make the block ciphers lightweight and IoT suitable. Curupira, PRESENT, KATAN, TEA, Hummingbird, RECTANGLE, SIMON are a few lightweight block ciphers [64] which are being researched. A typical block cipher model is illustrated in Figure 12.

- Key Management. A key management protocol is considered secure if it has security characteristics such as availability, integrity, confidentiality, authentication, and non-repudiation. IoT key management protocols can be divided into three classes: centralized, decentralized, and distributed. Centralized key management protocol employs the Key Distribution Center, which acts as a server that is consulted before communication takes place among group members. It is also responsible for assigning encryption keys to each group member. To avoid a single point of failure, a decentralized key management protocol can be used to distribute the encryption group key to all group members. For the distributed protocols, members of a group cooperate to establish a common session key [65].

3.2. Authentication solutions

- Device or Identity Authentication. This is the process of confirming the identity of objects in the network. From an IoT perspective, every object is required to have the ability to identify and authenticate all other objects in the system or a part of the system with which it interacts. Four proposed authentication methods [66] are hardware-based, token-based, non-token based, and procedural.

Procedural Authentication. This can be one-way, two-way or three-way authentication. One-way authentication is where a principal or trusted device authenticates an untrusted entity. Figure 13 shows an overview of the Physical Unclonable Functions (PUFs) protocol, which is a one-way authentication technique. Two-way authentication (or mutual authentication) is where both the trusted and the untrusted entity authenticates each other. Three-way authentication is where a third party is involved in the authentication process of two entities that identifies the third party as trusted.

Token-based. This method authenticates an object based on a piece of data created by a server. A token is often obtained when the user enters a valid username and password. With this token, an entity is given access to the authenticator's resources. Protocols used to issue tokens include OAuth2 protocol and open ID. Figure 14 shows a description of how token-based authentication is utilized in an

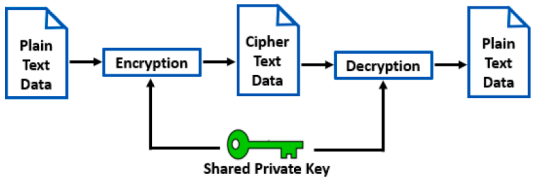


Fig. 9. Symmetric Key Cryptography

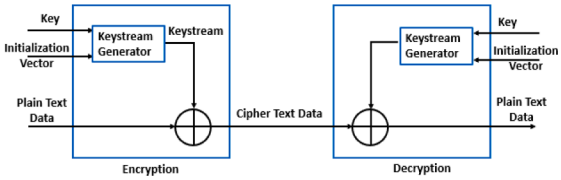


Fig. 10. Stream Cipher

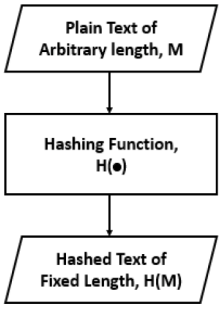


Fig. 11. Hash Function

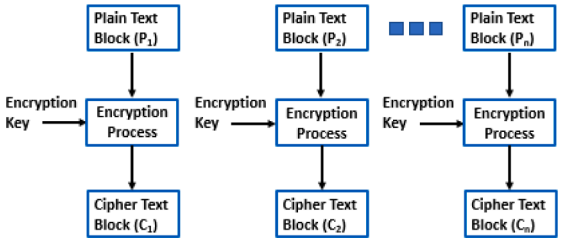


Fig. 12. Block Cipher

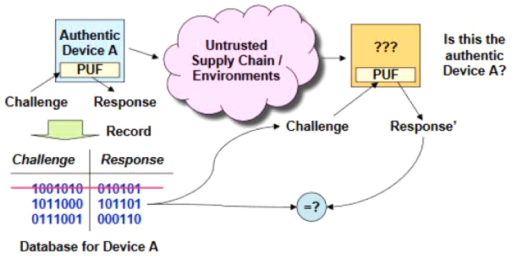


Fig. 13. PUF-based One-way Authentication [67]

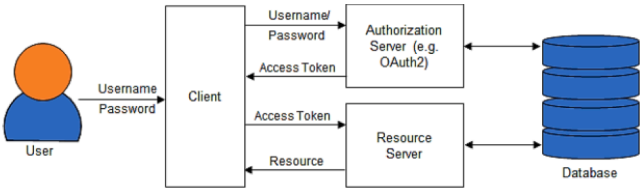


Fig. 14. Token-based Authentication

IoT application.

Non-Token-based. This method requires the user to enter credentials to send or receive data. A common protocol for this technique is TLS/DTLS (Transport Layer Security/Datagram Transport Layer Security). DTLS is used to handle the unreliable nature of UDP; due to the fact that it does not check for packet loss. DTLS on its own is computationally expensive. Figure 15 shows how the DTLS handshake is utilized for mutual authentication. Lightweight DTLS versions for constrained devices can be found in literature.

Hardware-based. This method utilizes the physical characteristics of the hardware to carry out authentication. These methods can be classified into two groups [66]: Implicit and explicit. Implicit hardware-based methods utilize the unique physical characteristics of the hardware to implement the authentication method. Two common methods found in the literature are the True Random Number Generator (TRNG) and Physical Unclonable Functions (PUFs). PUFs areis one of the most researched lightweight hardware-based authen-tication techniques in the literature. Explicit hardware-based methods use keys stored on an IC (or a trusted platform module) to carry out authentication.

- Data Origin/Message Authentication. Message authentication pertains to the integrity of the data/message in transit. That is, the message is not modified during transit and the individual receiving the message can verify the origin of the data/message. Message authentication can be achieved using: message authentication codes [68], digital signatures, or PUF [69]. Figure 16 depicts the "LightMAC" algorithm utilized in Message M authentication.

3.3. Security solutions for IoT platforms

IoT software platform is defined as a software that facilitates the sharing of data and services among IoT devices on a network. The features [71] of a platform include connectivity and network management, device management, processing analysis and visualization, application enablement, security, event processing, monitoring, integration and storage, and data acquisition. The security solutions for a platform can be divided into four areas: integrity of data while in transit, secure data storage, identifying devices requesting a connection and sending data, and authorization of users or entities. IoT software platforms can be divided into two categories, cloud-based platforms, and open-source platforms.

- Closed-Source IoT Platform. IoT closed-source platforms integrate the functionalities of IoT devices and cloud computing as a service over an end-to-end to the platform. Table 2 provides an overview of the security solutions, discovery and communication protocols available in common cloud-based IoT platforms available on the IT market today.
- Open-Source IoT Platform. Open-source platforms allow users to download, modify, and share the platform source code. Table 3, details the security solutions, discovery, and communication protocols available in common open-source IoT platforms.

3.4. Promising security solutions for IoT

- Artificial Intelligence and Machine Learning. The heterogeneous nature of IoT coupled with the additional complexity of security attacks challenges classical security methods used to secure devices, applications, and networks from intrusion. Machine learning algorithms build a model based on sample data, which can be used to predict, identify, and classify patterns in the data. As a result, it is used to provide access control, secure identity, and malware detections in computer systems. In [72], a survey is presented to describe studies utilizing ML algorithms to create models running at the access gateway of large-scale networks to detect IoT malware activity based on inbound scanning traffic patterns. Among these ML algorithms are SVM, decision trees, naive Bayes, artificial neural network, k-means clustering, fuzzy logic, genetic algorithms, and stacked auto-encoder. [73] describes studies that utilized ML models to identify, predict, and classify DoS, malware, and eavesdropping attacks.
- Blockchain. Blockchain was designed to record cryptocurrency financial transactions among its users (or nodes). There are four main technologies found in blockchain: distributed ledger, consensus algorithm, cryptography, and smart contracts. The main security characteristics found in blockchain are decentralization, immutability, non-repudiation, transparency, pseudonymity, and traceability. The presence of these technologies and security characteristics makes it a suitable candidate to be used in securing IoT networks. Blockchain improves the security of IoT systems by encrypting data at rest and stored data and digitally signing them

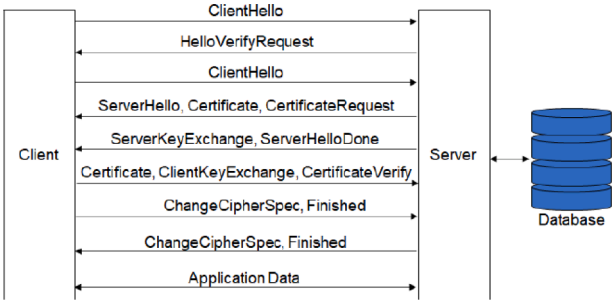


Fig. 15. DTLS Handshake Protocol

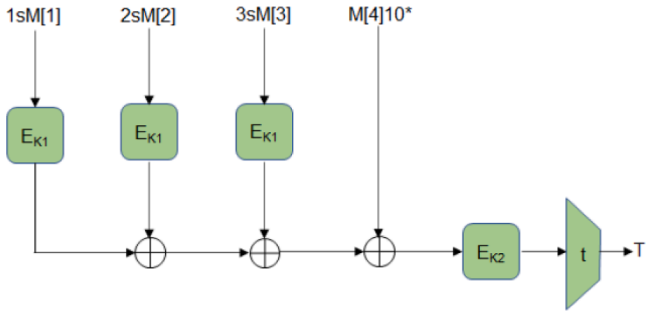


Fig. 16. Structure of LightMac [70]

Table 2
Characteristics of Common Cloud-based IoT Platforms

Company	Platform as Service (PaaS)	Security Features	Comment
Microsoft	Azure IoT [120]	Access Control/Authorization: Role-based Access Control to secure storage. Authentication: Shared Access Signature (SAS) to delegate access to resources in storage. Secure data in transit: HTTPS. Wire Encryption: SMB 3.0 encryption. TLS preserve data integrity between device and cloud platform. Encryption of data at rest: storage service Encryption, client-side encryption, and Azure disk encryption. AES-256 used in storage encryption.	Azure IoT Platform Security Services: Azure Security Center for IoT. Azure Sentinel
Amazon	Amazon Web Services IoT (AWS IoT) [121]	Secure data in transit: X.509 client certificates and TLS. TLS encrypts the connection between the device and the broker. Authorization: using IAM policy. Encryption of data at rest: Amazon DynamoDB encrypts your data using AES-256.	Shared Security model: Security of the cloud (AWS IoT). Security in the cloud (AWS service used).
Google	Google Cloud IoT [122]	Secure data in transit: TLS encrypt communication between device and cloud platform. Authentication: Digital Certificate. Encryption of data at rest: database use AES-256 or AES-128.	All security and manage-ment services are built into the Google Cloud IoT Core.

Table 3
Characteristics of Common Open Source IoT Platform

Platform as a Service (PaaS)	Security Features	Comment
Thingsboard [123]	Secure data in transit: SSL/TLS used for transferring data over the network. Authentication: X.509 certificates and access tokens.	The Professional Edition is a closed source advanced version of open source IoT platform with several features.
SiteWhere [124]	Authentication and Authorization: Security Assertion Markup Language (SAML), OAuth. Encrypt data in storage: InfluxDB that utilizes AES-256.	Spring security framework: focuses on providing both authentication and authorization.
DeviceHive [125]	Basic Authentication using JSON Web Tokens (JWT). Secure data in transit: SSL/TLS for devices and applications communication. Access Control/Authorization: Role-based Access Control	IoT platform is published on the Microsoft Azure Marketplace.

with cryptographic keys. In some cases, vulnerabilities are identified in the firmware of devices, but most devices do not provide a means to securely upgrade the firmware to patch the vulnerability. Blockchain utilizes smart contract technology that can securely facilitate automatic updates that can remedy any vulnerabilities found in the firmware of IoT devices [74].

3.5. Applications-based security solutions

[75] lists the security threats faced by some of the common IoT devices available in today's IT market. Below is a list of common IoT devices with comments on security issues and security solutions for these devices.

- RFID Tags. RFID tags main constraints are area, power, processing power, and storage. Therefore, ultralightweight and lightweight security methods are introduced. Security Solutions: Hash functions: Keccak and Spongent, block ciphers: TEA and KATAN, stream ciphers: Grain and Trivium [76].
- Smart Watches. Hackers use security threats as a means of gaining access to your smartphone, which has sensitive data. Security solutions: Pairing-Based Cryptography, SHA 1, SHA 2, RSA 1024, RSA 2048 E and RSA 2048 D [77].
- Universal Plug and Play (UPnP) Enabled Devices. UPnP can allow a malicious program to bypass the firewall entirely. Security solution: disabling UPnP on routers unless necessary.
- Small Automobile. Cars today are equipped with devices that use Bluetooth, radiofrequency, and internet connectivity for communication. These devices implement security in different ways; this heterogeneity makes the car vulnerable to external remote attacks. Security solutions: Authenticated Routing for Ad hoc Network (ARAN), Secure and Efficient Ad hoc Distance (SEAD), Ariadne, Holistic, and ECDSA [78].
- Children's Toys. Personal data stored on these devices must be encrypted in the event a hacker gets access to the device. Security solutions: toys need to be compliant with relevant security standards and regulations by groups such as GDPR, COPPA, and PECR. Transport Layer Security for authenticated sessions [79].
- IoT Camera. Many of the vulnerable devices lack authentication of protocols utilized in streaming video and also the encryption of all communication between the camera, applications, and servers. Security solutions: encrypt data at rest with a standardized method (such as AES), use Transport Layer Security (TLS) for encryption of data in transit, and account authentication over HTTPS [80].
- IoT Medical Devices. There are no reports of malicious hacking of IoT medical devices. But there are reports of successful hacks. Security solutions: secure data flowing to and from devices using AES-256, cryptographically sign data to protect against tampering, such as the storing private key on Trusted Platform Module (TPM). Use multi-factor authentication for user access. Use Hardware Security Modules (HSMs) with server-side applications to prevent unauthorized access [81].
- IoT Thermostat. The secure operation of this device can be a matter of life or death: there are places with sub-freezing temperatures in the winter and high temperatures in summer seasons. Security solution: Google Nest uses 2-factor authentication, Wi-Fi Protected Access II (WPA2) & WPA3 security to join the Wi-Fi network. Communication is secured by TLS [82].
- Smart Lock. A smart lock does not make a lock more secure than a traditional mechanical lock. Security solution: August smart lock. Two-factor authentication, two-layer encryption using Bluetooth Low Energy (BLE) encryption, and a lost phone feature that allows the user to disable all virtual keys stored on the application installed in the phone [83].
- Air Quality Sensor. These devices are deployed as autonomous wireless sensor nodes. Security methods are needed to protect them from security threats that may affect data integrity. Security solutions: WPA2 or WPA3 should be used to protect Wi-Fi, and HTTPS used for end-to-end encryption. Flash encryption to protect sensitive data on the flash [84].
- Industrial IoT Devices. In the implementation of security methods in a SCADA system an "egg shell" network model must be avoided at all cost. Security solutions: Hash algorithm (SHA2 to SHA5) along with Hash-based Message Authentication Code (HMAC) can be used for symmetric keys. Elliptic Curve Digital Signature Algorithm (ECDSA) for asymmetric keys. Secure mutual authentication, a hardware root of trust (HRoT) mechanism such as a hardware security module (HSM), and a TPM [85].
- Voice Activated Assistants. Research has shown that with or without physical access to the device a hacker can give it audible commands. Security solutions: multi-factor authentication methods. WPA2 or WPA3 for encryption [86].

3.6. *Hardware security solutions*

Since the IoT node devices are constrained in nature in terms of area and power, it necessitates a lightweight cryptographic solution. The crypto-graphic solution can be implemented either in hardware or in software. Hardware implementation versus software implementation of cryptographic systems is a continued topic of debate [87]. Both approaches have certain pros and cons. Hardware implementations of cryptographic systems are much faster in operation, and once built they cannot be tampered easily. Although implementations of software cryptographic systems are cost-effective and more flexible. Uncontrolled memory access and the vulnerabilities allowed by the operating systems make software cryptographic systems to provide much lower levels of security than their hardware equivalents [88]. Overall, a hardware design would have stronger security and better performance than a software solution and hardware implementation of lightweight cryptographic solutions would be the best approach for adding cryptographic measures in constrained IoT devices. Encryption block, authentication block and Random Number Generator (RNG) are the most important parts of a hardware cryptographic module.

Encryption is one of the most important methods of protecting data both in transit and at rest [89]. The hardware implementation of encryption systems for IoT devices should be designed in a manner which is not only more secure but also area and power efficient. For hardware implementation to date the required gate equivalent for lightweight block ciphers is generally less than 2K GE (Gate Equivalents) [90]. Asymmetric encryption algorithms are not area efficient and complex in nature. Considerable research has been done on the state-of-the-art symmetric algorithm Advanced Encryption Standard (AES) [91], which has been considered to be a standard and relatively safe for a long time. Unfortunately, AES implementations are not lightweight, even an area efficient AES requires an area of about 3400 gate equivalents [90]. For constrained IoT devices, there are some lightweight 8-bit AES implementations under research, which could be a candidate for securing the constrained IoT devices [92]. But, 8-bit lightweight AES has relatively very slow performance in constrained devices compared to a standard 128-bit AES [90]. For AES, blocks of information are encrypted thousands of times with the same key, which is a weak point. Apart from resource and performance constraints, AES or any other block ciphers are vulnerable to language frequency analysis, when they are used for encrypting long files [93, 94].

To minimize the effects of frequency analysis, Cipher Block Chaining (CBC mode) [91] methods were introduced. But researchers have found ways to detect patterns by analyzing the information bleeding through these blocks [95]. Side channel attacks [96, 97] and exposure to some level of collisions can also make AES weak [98]. Implementation of additional modes with AES makes it more unsuitable for lightweight solutions. In [99], authors proposed a lightweight polymorphic encryption technology as an attractive alternative encryption solution for constrained devices. To this date nobody is regularly securing the data associated with IoT devices. Part of the reason for this lack of security is the complexity and area required to implement strong encryption for the device. The results were directly compared with the state of the art encryption, AES and with PRESENT [100], which is another lightweight hardware encryption candidate. The hardware implementation of [99] showed that it is not only more secure but also area and power efficient.

All cryptographic solutions require a strong RNG. Nowadays, cryptographic solutions that depend on deterministic forms such as Pseudo-Random Number Generators (PRNGs) to produce randomness are not secure since they depend on deterministic algorithms. A stronger arrangement is to use Hardware Random Number Generation. But not all of them are secure enough. Numerous still depend on classical material science processes that run in an uncontrolled and chaotic way. Moreover, they intensely depend on post-processing calculations that are deterministic. These are not secure to supply randomness as the quality of their entropy source isn't stable [101, 102]. Quantum Random Number Generators (QRNG) is currently in development for smaller silicon devices such as dedicated hardware security modules [89].

As mentioned above hardware security solutions such as authentication are dominating the research landscape as researcher believe that hardware security solutions are the answer to providing security (especially encryption and hardware) for resource constrained devices. PUF primitives are the proposed hardware security solution for authen-tication.

4. Emerging technologies: challenges and countermeasures

4.1. Machine Learning (ML) security risks

Despite the variety of the machine learning system designs, all the machine learning systems composed of the same pipeline which impose some security risks that are applied to any specific machine learning system. Figure 17 shows the pipeline of generic machine learning system. This figure illustrates nine basic components of any machine learning system: 1) raw data in the world, 2) dataset assembly, 3) datasets, 4) learning algorithm, 5) evaluation, 6) inputs, 7) model, 8) inference algorithm, and 9) outputs. In this section, we identify the risks associated with each of the nine components of the generic machine learning system. Some of these risks are cross-referenced between the different components. Additionally, a set of controls are suggested to mitigate the security risks due to employing each of the machine learning components.

Raw Data in the World: When it comes to sec-urity, data plays as important role in machine learning system as the learning

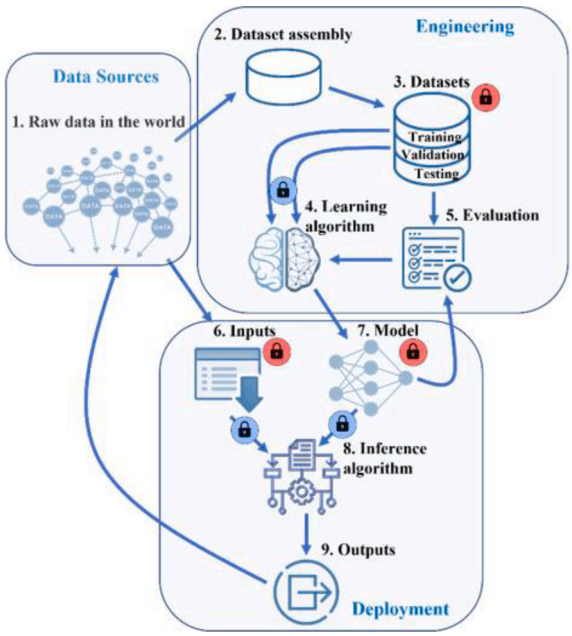


Fig. 17. Components of a generic machine learning system with attacks associated to some components or dataflow. Arrows represent the information flow. Manipulation attacks are pictured in red at the site of attack: data manipulation attack at (3. Dataset), input manipulation attack at (6. Inputs), model manipulation attack at (7. Model). Extraction attacks are pictured in blue, showing the flow of information: data extraction attack (dataflow form the Dataset to the Learning algorithm), input extraction attack (dataflow form the Inputs to the Inference algorithm) and model extraction attack (dataflow form the Model to the inference algorithm).

algorithm and any technical deployment. Raw data here means any kind of the data used in the machine learning system rather than training data only.

Potential Security Risks:

- Data Confidentiality: maintaining data confid-entiality in a machine learning system is more challenging than in a standard computing system. This is because a machine learning system that is trained up on sensitive or confidential data will have some aspects of those data represented into it through training. Attacks to extract confidential and sensitive information from machine learning systems are well known [103]. Note that even feature extraction might be useful since that can be used to enhance adversarial input attacks [104].
- Trustworthiness: Data sources are not reliable, suitable, and trustworthy. The attacker might tamper with or poison raw input data [105].
- Storage: Sometimes, data are managed and stored in an insecure manner. This security risk could be mitigated by access control. Such controls are not attainable when employing public data sources which make it easy for an attacker to control the data sources.
- Encoding Integrity: The raw data are not representative of the problem that is being solved by the machine learning system. The encoding done on these data should be designed in a way to preserve the integrity of the data.
- Representation: Data representation plays an important role in input to machine learning system. Representation schemes should be carefully considered, especially in cases of text, video, API, and sensors.
- Looping: The machine learning model might be confounded by feedback loops when data output from the model are later used as input back into the same model.
- Data Entanglement: data entanglement is considered a security risk in machine learning system. This can be avoided by knowing which part of the data can change and which should not ever change [106].
- Metadata: Not always metadata can help improve the performance of machine learning models. Metadata may be a “hazardous feature” which seems useful on the face of it, but actually deteriorate generalization. Metadata may also be vulnerable to tampering attacks that can confuse the machine learning model [107].
- Time: In some IoT applications, time of data arrival is crucial. In these systems, the attacker could control the network lag.
- Sensor: The technical source of input should be highly considered. Reliability of the sensors used to gather the data is an important factor. Sensor blinding attacks are one example of a risk encountered by poorly designed data collecting systems.
- Utility: The data and the learning model should be well chosen to reach a correct conclusion regarding the machine learning approach. It is worth noting that machine learning systems can fail due to data problems.

Proposed Controls: We suggest some of the controls that could be applied to mitigate the above-mentioned risks related to the raw data in general. These controls include protecting the data source if possible. One important thing is to check the data algorithmically before feeding it into the model (e.g., using range distribution analysis, outlier detection, mismatched unit discovery, etc.). Another control is to apply some transformation to the data to preserve data integrity and to create some features so that the data is consistently represented. Additionally, using version control technology to manage the dataset is a required control to secure the data. Aside from giving general controls, we highlight some controls for specific data risks. First, to protect data confidentiality, the machine learning system should be designed so that data extraction from the model is expensive. For example, consider using mathematical properties of the raw input space when choosing the model. Second, regarding to the data presentation, a review and periodic validation of the data representation is a good control. Third, avoiding the loops in the data streams to reduce the looping risk. Finally, sensor risks could be reduced with overlapping and correlated sensors that build and maintain a redundant data stream.

Dataset Assembly: The raw data is pre-processed before being fed into the learning algorithm. This step introduces some risks that are mentioned below:

Potential Security Risks:

- Encoding Integrity: As can be seen in the raw data, encoding can introduce some security risks in the pre-processing step. For instance, normalization of Unicode to ASCII may introduce problems when encoding.
- Annotation: The way data are annotated into features can be directly attacked. This introduces attacker bias into a system.
- Normalize: The nature of the raw data could be changed by normalization and as a result data might become exceedingly biased and easily attacked.
- Partitioning:Splitting the datasets into training, validation and testing may be done in a bad way to facilitate data bias.
- Fusion: In some cases, the machine learning system could be more robust if the input is provided from multiple sensors. In these cases, data sensitivity will be a big risk and should be monitored.
- Filter: If the raw data filtration method is known to the attacker, this knowledge may be leveraged into malicious input later in system deployment.
- Adversarial Partitions:If by some way, the attackers can influence the dataset partitioning into training and testing, they can control the whole machine learning system. An interesting attack is boosting an error rate in a sub-category. Because some machine learning systems are “opaque,” setting up special trigger conditions as an attacker may be more easily achieved through manipulation of datasets than through other ways [105].
- Random: In stochastic models, Randomness plays an important role. It is recommended to use cryptographic randomness sources instead of normal ones like Monte Carlo. In addition, random generation of dataset parts may be at risk if the attacker can control the source of randomness.

Proposed Controls: One general control that helps in reducing the security risks related to the data assembly process is to ensure the data sanity checks that look at ranges, outliers, probabilities, and other aspects of the data to detect anomalies before they are incorporated in the datasets. A suggested control of the fusion risk is to monitor the data from the sensor to take an action in case of sensor failure or dirty data comes out from the sensor.

Datasets: The assembled data is portioned into training, validation and testing groups. A special care must be taken in the creation of these groups in order to avoid predisposing the machine learning algorithm to future attacks. The most important part is the training set because it highly influences the system's future behavior. Below are some potential security risks associated with the datasets:

Potential Security Risks:

- Poisoning: The poisoning attacks is happened when the attacker intentionally manipulates the data to cause abnormal training process. In this sense, the designer should consider how much of the training data could be under control of the attacker and to what extent [108].
- Transfer: The transfer attack happens when the transfer learning technique is employed, especially when the pretrained model is widely available. One should consider whether the machine learning system could possibly be a Trojan that includes sneaky machine learning behavior that is unexpected [109].
- Dissimilarity: The training, validation, and test sets should be similar from trustworthiness, data integrity and mathematical perspective for the machine learning system to work properly in a secure manner.
- Supervisor: In supervised training models, the model could be incorrectly trained by Malicious introduction of misleading supervision.
- Online: In online learning, the dataset manipulation could be tricky where an attacker can slowly retrain the model in wrong way.

Proposed Controls: One general control that helps in reducing the security risks related to the dataset is to characterize the statistical overlap between training and validation sets, because of the high similarity between them will yield to overfitting. A suggested control of the dissimilarity risk is to ensure data similarity between the three groups of the dataset using mathematical methods.

Learning Algorithm: The learning algorithms suffer from security risks but much less than the data used to train, test, and operate the machine learning system. Machine learning system could be divided into two main categories, offline and online learning systems. From a security perspective, there is some advantage to an offline system because the online system increases exposure to a number of data vulnerabilities over a longer period of time.

Potential Security Risks:

- Online: In the online learning systems, the attackers could nudge the system in the wrong direction by getting access to the training data.
- Randomness: In machine learning models, setting thresholds and weights randomly must be done with care. Using the wrong sort of random number generator may introduce to a security problem. Cryptographic randomness is preferred for a secure system.
- Blind Spots:The learning algorithms may have blind spots which could open the system up to easier attack through methods that include adversarial examples.
- Confidentiality: Some learning algorithms are not suitable for dealing with confidential information. For instance, using non-parametric algorithm like k-nearest neighbors to process sensitive medical records is not a good ides since the examples should be stored on servers. Algorithmic leakage is an issue that should be considered carefully [104].
- Oscillation: If the learning algorithm is using gradient descent in a space where the gradient is misleading, the model may end up oscillating and not converging.
- Hyperparameters: One of the possible risks to the learning algorithm is hyperparameters tuning by the attacker. In some cases, the attacker can tweak, hide, or even introduce the hyperparameters to the system.
- Hyperparameter sensitivity:Sensitive hyper-parameters present high security risk, especially if they are not locked in.
- Sensitive Hyperparameters not accurately evaluated and explored can cause overfitting. Additionally, hard to detect changes to hyperparameters would make an ideal insider attack.

Proposed Controls: To reduce the randomness risk, it is recommended to take a look at the randomness technique used. Representational robustness can help mitigate some blind spot risks (for instance, using word2vec encoding Instead of one-shot encoding in an NLP system). The model choice should preserve representational integrity to protect the confidentiality. In addition, if the history of queries to the system is kept in a log and reviewed periodically, this will ensure that the system is not unintentionally leaking confidential information. In order to combat the hyperparameters risk, the hyperparameters should be chosen carefully and locked in. Also, performing a sensitivity analysis on the chosen hyperparameters will reduce the security risk against hyperparameter sensitivity.

Evaluation: Machine learning system evaluation is a process that comes in after the learning process to ensure the proper operation of the system. This is done using evaluation data. Below are the potential security risks associated with this process:

Potential Security Risks:

- Overfitting: The overfit models can be easily attacked through the input patterns since adversarial examples need only be similar to the training examples.
- Bad Evaluation Data: Evaluation data must be carefully designed and used. If they are too small or too similar to the training data, this imposes a potential risk [105].

- Cooking the Books:In some cases, evaluation data might be structured in a way to make the system working properly even when it's not.
- Catastrophic Forgetting: This risk arises When a model is crammed too full of overlapping information. Online models are more susceptible to this risk.
- Data Problems: The training and evaluation processes could be difficult due to the upstream attacks against data.

Proposed Controls: In order to control the bad evaluation data risk, public data sets with well-known generalization rate may be recruited. When the evaluation data and results are public, it will be much harder for the bad evaluation data and cooking the books risks pulling off. The researchers are beginning to move toward releasing the source code and data for reproducible results.

Inputs: The input data fed into the trained model may introduce some security risks. Below are some of these risks.

Potential Security Risks:

- Adversarial Example:Malicious input is considered one of the most important classes of the computer security risks. When it comes to the machine learning systems, it is known as adversarial examples. Adversarial examples have received so much attention that they swamp out all other risks in most people's imagination [110].
- Controlled Input Stream:An attacker may purposefully manipulate the input data to a trained machine learning model that takes its input data from outside.
- Dirty Input:Noise is everywhere in the real world. If the input data are dirty and noisy, it will be hard to process. A malicious program can leverage this susceptibility by merely adding up noise to the input data.

Proposed Controls: To mitigate the risk of controlling input stream, a multi-modal input might be considered in the design of the machine learning system. This could be achieved by using multiple sensors that are not similarly designed or that do not have the same failure conditions. To control the dirty input risk, sanity checks, data cleaning and filtering can be carried out.

Model:The model risks appear when the trained model is ready to be deployed. Note that some of the evaluation risks are applied in this step (for example, overfitting and catastrophic forgetting).

Potential Security Risks:

- Trojan: If the model is transferred, there is a possibility that the reused model is Trojaned or damaged version of the original model [103].
- Training Set Reveal:Most machine learning algorithms learn a great deal about input and store its representation internally. The problem appears when the input data contains sensitive information. Choosing the right algorithm could help control this risk. Also, care should be taken on the output produced by the system as it may reveal sensitive aspects of the training data.
- Steal the Box: Through direct input/output observation, an attacker can steal the machine learning system knowledge. This is like reversing the model.

Proposed Controls: In order to reduce the steal the box risk, the output provided by the model should be watched carefully.

Inference Algorithm:The inference risks are considered in the inference stage when the fully trained model is ready for use.

Potential Security Risks:

- Online:In an online learning system, the model can be easily pushed past its boundaries by the attacker.
- Inscrutability: it is very real risk to integrate a machine learning system into an IoT environment to do its task without understanding about how it works.
- Hyperparameters: The learning system could be controlled if the attacker can modulate the hyperparameters of the inference model after the evaluation process is done.
- Confidence Scores: In many instances, confidence scores can help an attacker. For example, if the machine learning model is not confident about its decision, that provides a feedback to the attacker with regards to how to tweak the input to make the model misbehave.
- Hosting: As many models are operated on hosted or remote servers. One should take care to protect these models against machine learning related attacks.

Proposed Controls: It is recommended to protect the system against the online attack by refreshing the deployed model to a known state, resetting, or otherwise cleaning it periodically. One control against the hosting attack is to isolate the engineering machine learning system from the production system.

Outputs: One of the important attacks against machine learning systems deployed in an IoT environment is attack the system's output.

Potential Security Risks:

- Direct: This security risk happened when an attacker tweaks the output streams directly. This could be done by interposing between the output stream and the receiver.
- Provenance: machine learning systems should be trustworthy to put into use. Even, some partial or temporary attacks against the model's output can cause trustworthiness.

- Miscategorization: A fallacious output could be resulted by adversarial examples. If those outputs are utilized without detection, it will lead to big problems.
- Transparency:Transparency is achieved when the system's decision is explained well. On the other hand, opaque systems are not explained and hence are much easier of being attacked because it is harder to recognize any unusual activity.
- Eroding Trust: A whole system trust could be eroded if the deployed machine learning system misbehaves. For example, a GAN that outputs uncomfortable images or sounds [111].

All the mentioned attacks could be classified into two main groups. The first group is manipulation attacks which attack the operational integrity to manipulate the behavior. The second group is extraction attacks which attack confidentiality by extracting information from the system. If we can apply these attacks on the three pillars of the machine learning system (Input, data and model) we can get six different general attacks which are illustrated in Figure 18.

4.2. *Blockchain technology security risk*

As discussed in Section V, the main security characteristics found in blockchain are transactional privacy, decentralization, immutability of data, non-repudiation, transparency, pseudonymity, and traceability, integrity, authorization, system transparency, and fault tolerance. Even with these beneficial security characteristics there are some security threats or risks in implementing blockchain technology in IoT. Reviewed a list of potential risks to the components of blockchain along with associated controls to mitigate the effects of these threats. Figure 18 shows nine (9) components of blockchain: 1) system management, 2) blockchain consensus, 3) blockchain network, 4) membership services, 5) events, 6) ledger, 7) smart contract, 8) system integration, and 9) wallet.

In this section we discuss the risks faced by each blockchain component along with suggested controls to mitigate these risks. Our aim is to have blockchain that can easily be integrated in an IoT network without any concerns about security.

System Management. This provides the ability to create, change, and monitor the blockchain components. With this important function this component has some security risks associated with it.

Potential Security Risks:

- Transaction Integrity: the integrity of the transaction may be compromised if the account address of the peer node carrying out the transaction is not unique. To preserve the transaction integrity the private keys should not be reused in another transaction. Normally an ECC algorithm is used to generate the public key using the private key. An attacker who knows the reused private key will be able to determine the public key and has the ability to read the transaction data in its raw form. Verifying a new transaction block means that each peer node compares the stored hash of the blockchain with the hash of the new transaction block. In 2020 Leurent and Peyrin [112] showed that hash functions (e.g., SHA-1 and MD5) are vulnerable to collision attacks.
- Fault Tolerance: blockchain is robust even if there is a peer node or a component failure. But there is a problem if the peer node that fails is the point in the network that is connection point between the network and the gateway. A DoS attack on this node would render the blockchain inoperable. DoS attack on multiple blockchain will affect the reliability or availability of the network. How many nodes is required for the blockchain to be considered reliable when multiple nodes fail or are maliciously attacked at the same time?
- Monitoring a Single Blockchain Node: just monitoring a single peer node to obtain the number of transactions, processed and integrated into a block to join the blockchain. This monitoring does not provide information on resource usage at the peer node.

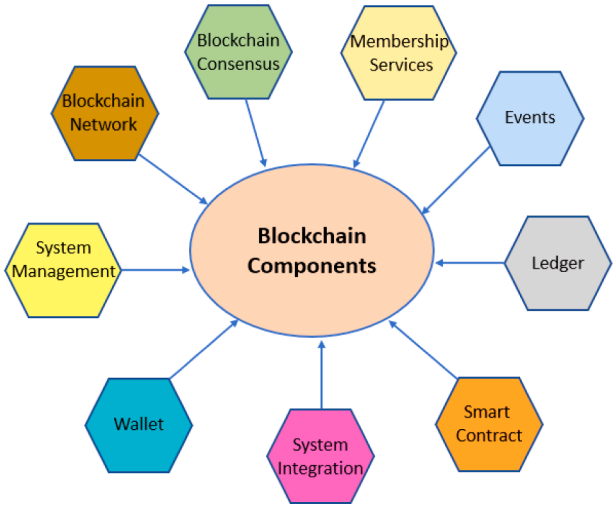


Fig. 18. Blockchain Components

This monitoring also doesn't provide information about the health of other peer nodes or information about the bandwidth, throughput, and latency of the peer network.

Proposed Controls: The integrity of the data stored in each block of the blockchain is very important. Once a transaction block is added to the blockchain it cannot be removed. SHA-256 is the main hashing function used in public and private blockchain. The gateway into the blockchain network needs redundancy, by having multiple gateways. The health of the blockchain network hinges on the systems' performance. Therefore, employ tools to monitor all aspects of the blockchain. [113] proposed a solution to DoS attacks: by designing a reputation model to score the behavior of all the consensus nodes in the consensus process, and the faulty nodes will get lower reputation if any malicious behavior is detected.

Blockchain Consensus. This is a set of data and processing peers located on the blockchain that continually maintain the replicated ledger. In blockchains, consensus between nodes determines how a new block is chosen to be added to the blockchain.

Potential Security Risks:

- Poor Performance. Depending on the consensus algorithm, network latency, network instability, malicious nodes, and the complexity of the block, the nodes may require more time to process a transaction and converge towards a single chain. This poor performance means the security and immutability of the blockchain is decreased.
- Liveness Attack. This attack delays the transaction confirmation time. Occurs in three stages [114]: preparation, transaction denial, and blockchain delay.
- Double Spending Attacks. This is a common blockchain attack when a transaction is duplicated. This attack is made possible by the fact that the same digital token can be spent 2 times [115]. Example, 51% vulnerability, race, vector46, and Finney attacks.

Proposed controls. In order for a blockchain to perform efficiently the consensus algorithm should converge very quickly. The consensus algorithm is always improving and these improvements help to decrease network latency, network instability, and the complexity of the block. Blockchain core network protects against double spending attack by verifying each transaction with the use of Proof-of-Work (PoW) or Proof-of-Stake (PoS) or a hybrid of both. mechanism [116].

Blockchain Network. Contains compute nodes that are connected to each other (via a mesh network).

Potential Security Risks:

- Blockchain Nodes System Upgrade. There needs to be secure software upgrades for especially computers in the peer network.
- DoS Attack. A DoS attack on the peer network is intended to bring down the entire blockchain network or make it difficult for new transactions to be processed. This is done at the gateway or router that receives transactions from users to be forwarded to the peer network.
- Transaction Malleability Attacks. This attack results in the user paying or sending a transaction twice. The attacker alters the transaction's ID and broadcast this transaction with a changed hash to the peer network. If this is confirmed before the original transaction, it will fail. The user sees this failure and resends another transaction. This results in the attacker compromising the integrity of the blockchain.
- Time-jacking. This can be carried out in two different scenarios. 1) Where the attacker has the ability to place fake peer nodes in the network and 2) The attacker has access to several peer nodes in the network with inaccurate timestamps. With this influence the attacker can alter the network time in order to force the acceptance of an alternative blockchain. However, this attack can be prevented by restricting how the system time is used.
- Routing Attacks. The gateway to the peer network if hacked the attacker can modify the transaction data before sending it to the peer network.
- Delay Attack. The attacker tampers with messages sent to peers in the network.
- Sybil Attacks. A sybil attack assign identifiers to nodes in the peer network. A simple solution to this attack is to require peer nodes to prove their identity before joining the network.
- Eclipse Attack. In this attack an attacker can exploit the connections in the peer network. With a successful hack of a blockchain node the attacker has access to the data sent to node, which can be used maliciously.

Proposed Controls: the computers holding the blockchain ledger is called the peer node. The peer nodes should be closely monitored for malicious attacks. If attacker is able to change or add a block to the blockchain, there is no way to revert the block to its original form before this attack. A promising solution to DoS or botnet attack is to use ML techniques at the gateway to the network by identifying and discarding malicious packets before they enter the block-chain network. A message acknowledge-ment from the intended destination can be used to determine if the peer got the message. [117] proposed the ''Beaver'' system, which protects users' privacy while resisting Sybil attacks by charging fees.

Membership Services. Manages identity and transactional certificates and access rights. The blockchain nodes have three roles: 1) Client (execute smart contract), 2) Peers (maintains a copy the blockchain (or ledger) and validate transactions), and 3) Orderer (establishes the order of all transactions).

Potential Security Risks:

- Fault Tolerance. No single blockchain node should be tasked with carrying out all these roles and the others have no roles.

- Transaction Certificate Authority (TCA) trust. Once a blockchain node is enrolled in the network the node a request is sent to the TCA. These certificates will be used for invoking a transaction on the blockchain. For a more secure transaction it is recommended that the certificate is only used in a single transaction. If a transaction certificate gets in the hands of an attacker may create blocks and send to the other nodes for verification.
- Enrollment Certificate Authority (ECA) trust. Allows new nodes to register with the blockchain network. It enables the blockchain node to request two enrollment certificates: one for data signing and the other for data encryption. If the certificate for data encryption can be compromised when used in Elliptic Curve Integrated Encryption System (ECIES) scheme that is vulnerable to side channel attacks and twist-security attack.

Proposed Controls. Blockchain is a distributed system. Therefore, for fault tolerance each blockchain node should be tasked with carrying out all 3 roles. There are simple countermeasures to protect the network from side channel attacks. Instead of recommending the transaction certificate be used for a single transaction, this should be a standard for all blockchain implementations.

Events. Generates notifications when blockchain carries out important actions that includes the generation of new block and smart contracts sign a contract.

Potential Security Risks:

- No Notifications. An attacker with access to the blockchain nodes may remove notifications which creates security issues. For e.g. An attacker with access to the blockchain nodes can remove notifications to hide data transactions made to the data in the blockchain. Notification builds trust, whether it is warning or just a notification of activities on the network.
- Notification Leakage. Sending unsecure notifications can result in an attacker gaining knowledge of the blockchain network activities.

Proposed Controls: The cryptographic technique used to protect transaction blocks and the ledger should be used to secure the notification message.

Ledger. This contains the current state of the blockchain transactions. Each device in the blockchain network has a copy of the ledger.

Potential Security Risks:

- Double Spending Attack. This would result in the ledger containing a block that was not created by an authorized user.
- Tampering. Since the ledger is stored on the peer nodes, then the ledger is susceptible to tampering.

Proposed Control. An access control mechanism on the ledger would protect it from individuals attempting to modify the ledger in hopes of not being found out. Physical security of the peer nodes should be considered to protect devices from tampering.

Smart Contracts. Smart contract is a program stored on a blockchain and runs when predetermined conditions agreed upon by the participants are met. These conditions or rules governs how transactions and data are represented on the blockchain network. The rules are typically used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without a third party's involvement. User accounts submits transactions that executes a function defined in the smart contract.

Potential Security Risks:

- Vulnerabilities in Contract Source Code. A smart contract with vulnerabilities in its source poses a risk to the parties using it to sign the contract. The programming language Solidity is used write smart contracts for blockchain. It introduces the possibility of a re-entry attack, where contract A calls a function from contract B with undefined behavior. Then contract B can call a function from contract A which can be used for malicious purposes [118]. It is imperative to employ coding best practices as bad coding can result in an attacker getting access to private transactions. In the case of digital currencies, the cost of smart contract can be high. A security risk may not be known until a successful attack has taken place.
- Vulnerabilities in Virtual Machines. Most blockchains are hosted on virtual machines in a datacenter on the cloud services. The virtual machine may be susceptible to immutable defects, access control problems and short address attacks [118]:
- Immutable Defects. Smart contract with bugs in the code are impossible to fix once it is created. An attacker can exploit this exploit to the code vulnerabilities to gain access to the blockchain network.
- Short Address Attacks. VM accepts any padded argument whether it is correct or incorrect. Attacker exploit this vulnerability by send modified addresses to blockchain nodes. Therefore, a blockchain node would view address as valid and send data to the modified address of the attacker.
- Access Control Problems. Smart contracts with missed modifier bug give attacker access to secret information in smart contract.
- DoS Attack. When the flow of control is transferred to an external contract, the execution of the caller contract can fail, which may result in the caller contract entering a DoS state as the caller contract execution is disrupted. [119].

Proposed Controls. Known flaws in programming can be averted when practicing best coding practices for smart contract: stay up to date on bugs found in smart contract and make changes to your coding. Check external contract calls for malicious codes. Know that any public function can be called maliciously by an attacker. The private data in smart contracts can be viewed by anyone. A SmartScan [119] is proposed by Fatima et al as a means of detecting DoS attacks on smart contracts. Integrating ML tools with smart contract to

scan smart contracts for bugs is a viable option to consider for identifying and mitigating malicious coding attacks.

System Integration. Function is to integrate blockchains in a bidirectional manner with external systems.

Potential Security Risks:

- Insecure Communication. Data sent to external systems without proper encryption can be intercepted by an attacker listening to the communication channel. Some attacks that can be carried out includes:
- Replay Attack. the aim of this attack is to spoof the identities of two parties, intercept their data packets, and relay them to their destinations without modification.
- Man-in-the-Middle Attack: By exploiting some vulnerabilities like: private key leakage and 51% vulnerability, an attacker by spoofing the identities of two parties can secretly relay and even modify the communication between these parties, which believe they are communicating directly, but in fact the whole conversation is under the control of the attacker.

Proposed Control. Any communication between the blockchain and external systems need to be secure. Monitoring the gateway for malicious packets can mitigate these attacks. Integrate robust multi-factor authentication techniques to ensure communication is between the intended parties.

Wallet. Manages the security credentials of the blockchain.

Potential Security Risks:

- Susceptible to Phishing Attack. Attackers send messages to an unsuspecting user asking them to enter private key information. With the private key they have access to your wallet.
- Private Key Security. the user's private key is regarded as the identity and security credential, which is generated and maintained by the user not a third party. If the attacker has the private key of a user, the attacker can modify the data on the device by creating a block to be added to the blockchain.
- Dictionary Attack. Brute force attack where hacker tries to guess the cryptographic hash by using a database of hashes for common passwords. If successful, the attacker can obtain the login credentials for the wallet.
- Vulnerable Signatures. Cryptographic algo-rithms that have vulnerabilities due to low levels of randomness (or entropy) in the values it creates. With this value an attacker may be able to read messages using this similar value for encryption.
- Flawed Key Generation. Poor randomness in key generation makes it easier for an attacker to obtain private keys which can be used to obtain the wallet credentials.
- Attacks on Cold Wallet. These are hardware wallets. By exploiting bugs in the wallet software an attacker can gain access to the private information of all the users of this software.
- Attacks on Hot Wallet. Hot wallet is an internet-connected application used to store private key. The cloud database is vulnerable to attacks. Even if the wallet is disconnected to the web, it is susceptible to physical attacks.

Proposed Controls. By adopting multi-factor authentication, the user can make it difficult (by requiring a lot of processing power and time) for these attacks to succeed in getting access to the data stored in the wallet. Strong passwords for wallet credentials are recommended. For phishing attacks, make sure you have a record of email addresses or website links used in communication with external systems. Cryptographic algorithms used for generating keys should have high randomness (entropy), this increases the difficulty to break the cryptographic hash.

5. Lessons learned

As mentioned earlier, security threats to IoT devices could be categorized into three main groups which are hardware threats, software threats, and threats to data in transit. A secure IoT device is both robust and handles any of these security threats. Data at rest is another aspect of security threats that should be looked at by the security experts. This aspect has shown up due to the emergence and integration of different technologies into the IoT systems. Data at rest represents an important concern from the security point of view. These data could be the saved parameters of an AI-powered device or data that are stored in the database for cloud-based applications. Additionally, the data stored in the IoT device itself should be protected against any potential threat. Protection of such data is significant for a secure IoT device.

Other lessons learned from the different security threats discussed earlier are summarized below:

- Corrupted data or software can compromise the security of IoT devices. Firmware updates and encryption of data are the best approaches to increase the security of the IoT device.
- Many protocols are secure on their own against certain threats but when they are integrated with a system, it might be vulnerable to the security threat it was initially secure against. As developers may uninstall security options due to resource constraints.
- It is difficult to guarantee security against some of the hardware threats as the IoT devices are mostly placed in environments that are not readily physically accessible to the owner.
- There is a growing concern about protecting IoT devices from software threats. This is a result of botnets evolving faster than solutions to protect the IoT device.

Regarding the security solutions investigated in this paper, cryptographic solutions are considered the most important security

solution for data protection. Cryptography can ensure the confidentiality and integrity of both data in transit as well as data at rest. It also performs authentication for senders, recipients, and one another to protect against repudiation. Continuous advancement of artificial intelligence, machine learning, and blockchain technology is also showing promising solutions to secure the internet of insecure things. We have also thoroughly discussed different application-based security solutions that could be considered when designing a robust application. Furthermore, hardware cryptographic solutions could be considered in real-time sensitive applications due to their high-speed operation and high-security performance compared to software cryptographic solutions. This will be at the cost of power consumption and the size of IoT devices. To alleviate these drawbacks, lightweight hardware security solutions have risen over the last decade. In summary, there are many proposed security solutions for IoT devices, applications, and platforms but we need to further explore promising security solutions to secure IoT systems, especially when emerging technologies are employed within the IoT system.

6. Conclusion

IoT has opened a door to a world of unlimited possibilities for implementations in varied sectors in society, but it also has many challenges. One of those challenges is security and privacy. IoT devices are more susceptible to security threats and attacks, due to their constraints. There is a lack of proper security solutions for IoT applications and is leading this world of securely connected things to the internet of insecure things. In this review article, we presented the current state of security in IoT, and what solutions need to be put in place to persuade users that IoT's image is not only about providing low-cost devices; but equally important is providing the best security solutions that address the security threats and privacy concerns. To keep it secure, the stakeholders: consumers, security administrators, and future IoT developers need to be knowledgeable about the security aspects of IoT. The role the developer plays is to ensure that security is the main goal during the design and development of the device or application.

In this paper, we presented and reviewed IoT security threats from several perspectives (i.e., hardware, software, and data in transit) with highlighting precautions related to different security threats. A review of the current security solutions is also demonstrated. In order to deal with the constraints IoT devices suffer from, we review the existing hardware security solutions with a comparative analysis that targets providing security to the IoT-constrained devices. Introducing emerging technologies to the IoT environment adds more vulnerabilities to the entire network security. We demonstrated two of the main emerging technologies namely, Machine Learning and Blockchain, their impact on the security when being employed in an IoT platform, and proposed solutions to mitigate these risks. This could help researchers with new directions to contribute to the field.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships

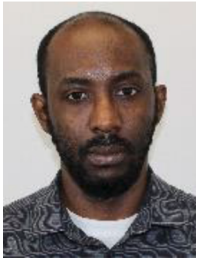
References

[1] K. Ashton, "That 'Internet of Things' Thing," June 22, 2009. [Online]. Available: https://www.rfidjournal.com/that-internet-of-things-thing. [Accessed on Jul. 4, 2020].
[2] European Union Agency for Network and Information Security (ENSIA), "Baseline security recommendations for IoT," November 20, 2017. [Online]. Available: https://www.ensia.europa.eu/publications/baseline-security-recommendations-for-iot. [Accessed: Jul. 4, 2020].
[3] S. Madakam, R. Ramaswamy, S. Tripathi, Internet of Things (IoT): a literature review, J. Comput. Commun. 3 (5) (2015) 164.
[4] M. Hasan, "IoT in healthcare: 20 examples That'll make you feel better," April 2, 2020. [Online]. Available: https://www.ubuntupit.com/iot-in-healthcare-20-examples-thatll-make-you-feel-better. [Accessed: Nov. 5, 2020].
[5] Lanner. "Examples of IoT devices in your next smart home," September 10, 2018. [Online]. Available: https://www.lanner-america.com/blog/5-examples-iot-devices-next-smart-home. [Accessed: October 10, 2020].
[6] A. Grizhnevich, "IoT for smart cities: use cases and implementation strategies," May 3, 2018. [Online]. Available: https://www.scnsoft.com./blog/iot-for-smart-city-use-cases-approaches-outcomes. [Accessed: Oct. 10, 2020].
[7] S. Chaudhary; R. Johari, R. Bhatia, K. Gupta and A. Bhatnagar, Craiot: concept, review, and application(s) of IoT. In 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), 2019.
[8] Alibaba, "Siemens and Alibaba cloud partner to power industrial Internet of Things in China," [Online]. Available: https://www.alibabagroup.com/en/news/press_pdf/p180709.pdf. [Accessed: Oct. 11, 2020].
[9] DHL, "Internet of Things," [Online]. Available: https://www.dhl.com/global-en/home/insights-and-innovation/thought-leadership/trend-reports/internet-of-things-in-logistics.html. [Accessed: Oct. 12,2020].
[10] Konux, "Transform railway operations for a sustainable future," [Online]. Available: https://www.konux.com. [Accessed: Oct. 12, 2020].
[11] Nexiot. [Online]. Available: https://nexxiot.com. [Accessed: Oct. 12, 2020].
[12] Scandit. [Online]. Available: https://www.scandit.com. [Accessed: Oct. 14, 2020].
[13] Apple. [Online]. Available: https://www.apple.com. [Accessed: Oct. 14, 2020].
[14] Cognigy. [Online]. Available: https://www.cognigy.com. [Accessed: Oct. 14, 2020].
[15] Huawei. [Online]. Available: https://www.huawei.com/us. [Accessed: Oct. 17, 2020].
[16] Samsung Electronics, "Samsung electronics to jointly build SKT World-First nationwide LoRaWAN network dedicated to IoT," [Online]. Available: https://www.samsung.com/global/business/networks/insights/press-release/samsung-electronics-to-jointly-build-skt-world-first-nationwide-lora-wan-network-dedicated-to-iot. [Accessed: Nov. 4, 2020].
[17] M.M. Hossain, M. Fotouhi, R. Hasan, Towards an analysis of security issues, challenges, and open problems in the Internet of Things, 2015 IEEE World Congress Serv. (2015) 21–28.
[18] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, B. Sikdar, A survey on IoT security: application areas, security threats, and solution architectures, IEEE Access 7 (2019) 82721–82743.
[19] A. Jurcut, T. Niculcea, P. Ranaweera, et al., Security considerations for internet of things: a survey, SN Comput. Sci 1 (2020) 193.
[20] N. Mishra, S. Pandya, Internet of things applications, security challenges, attacks, intrusion detection, and future visions: a systematic review, IEEE Access 9 (2021) 59353–59377.
[21] M. Noor, W. Hassan, Current research on Internet of Things (IoT) security: a survey, Elsevier: Computer Netw. 148 (2019) 283–294.

[22] H. HaddadPajouh, A. Dehghantanha, R. Parizi, M. Aledhari, H. Karimipour, A survey on internet of things security: requirements, challenges, and solutions, Elsevier: Internet of Things 14 (2021).
[23] M.A. Al-Garadi, A. Mohamed, A.K. Al-Ali, X. Du, I. Ali, M. Guizani, A survey of machine and deep learning methods for Internet of Things (IoT) security, IEEE Commun. Surv. Tutor. 22 (3) (2020) 1646–1685.
[24] S. Zaman, et al., Security threats and artificial intelligence based countermeasures for internet of things networks: a comprehensive survey, IEEE Access 9 (2021) 94668–94690.
[25] D.E. Kouicem, A. Bouabdallah, H. Lakhlef, Internet of Things security: a top-down survey, Comput. Netw. 141 (Aug. 2018) 199–221.
[26] Y. Harbi, Z. Aliouat, S. Harous, A. Bentaleb, A. Refou, A review of security in Internet of Things, Wirel. Pers. Commun. 108 (1) (Sep. 2019) 325–344.
[27] S.A. Hamad, Q.Z. Sheng, W.E. Zhang, S. Nepal, Realizing an internet of secure things: A survey on issues and enabling technologies, IEEE Commun. Surveys Tuts. 22 (2) (2020) 1372–1391, 2nd Quart.
[28] V.A. Thakor, M.A. Razzaque, M.R.A. Khandaker, Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities, IEEE Access 9 (2021) 28177–28193.
[29] A. Hameed and A. Alomary, "Security issues in IoT: A survey,'' in Proc. Int. Conf. Innov. Intell. Inform. Computing Technol. (ICT), Sep. 2019, pp. 1-5.
[30] Y. Lu, L.D. Xu, Internet of Things (IoT) cybersecurity research: a review of current research topics, IEEE Internet Things J 6 (2) (Apr. 2019) 2103–2115.
[31] S. Roy, Hardware Trojans – a cause of concern in SafetyCritical electronic systems, Int. J. Modern Trend. Eng. Sci. 4 (5) (2017) 110–119.
[32] S. Simranjeet, J.M. Bassam, H. Thaier, Hardware security in IoT devices with emphasis on hardware trojans, J. Sensor Actuator Netw. 8 (3) (2019) 42.
[33] J. Breier, W. He,. "Multiple fault attack on PRESENT with a hardware Trojan implementation in FPGA." In Proceedings of the IEEE International Workshop on Secure Internet of Things (SIoT), Vienna, Austria, 21–25 September. 2015; pp. 58–64.
[34] D. Das, S. Maity, S. B. Nasir, S. Ghosh, A. Raychowdhury and S. Sen, "High efficiency power side-channel attack immunity using noise injection in attenuated signature domain," 2017 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), McLean, VA, 2017, pp. 62-67.
[35] W. Zhou and F. Kong, "Electromagnetic side channel attack against embedded encryption chips," 2019 IEEE 19th International Conference on Communication Technology (ICCT), Xi'an, China, 2019, pp. 140-144.
[36] D. Genkin, A. Shamir, E. Tromer, Acoustic cryptanalysis, J. Cryptol. 30 (2017) 392–443.
[37] I. Ullah, N. Khan and H. A. Aboalsamh, "Survey on botnet: Its architecture, detection, prevention and mitigation," 2013 10th IEEE International Conference on Networking, Sensing and Control (ICNSC), Evry, 2013, pp. 660-665.
[38] Imperva. "DDoS Attacks." [Online]. Available: https://www.imperva.com/learn/application-security/ddosattacks. [Accessed: Jun. 5, 2020].
[39] R. Alnakhalny, M. Anbar, S. Manickam, E. Alomari, An intelligent ICMPv6 DDoS flooding-attack detection framework (V6IIDS) using back-propagation neural network, IETE Tech. Rev. (2015).
[40] R. Schlegel, K. Zhang, X.Y. Zhou, M. Intwala, A. Kapadia and X. F. Wang, "Soundcomber: a stealthy and contextaware sound trojan for smartphones," In Proceedings of the ISOC Network and Distributed System Security Symposium (NDSS'11).
[41] X. Fu, "On traffic analysis attacks and countermeasures." 2005. [Online]. Available: https://core.ac.uk/download/pdf/4271895.pdf. [Accessed: Jul. 29, 2020].
[42] I. Makhdoom, M. Abolhasan, J. Lipman, R.P. Liu, W. Ni, Anatomy of threats to the internet of things, IEEE Commun. Surv. Tutor. 21 (2) (2019) 1636–1675. Second quarter.
[43] WIRED. "Hackers remotely kill a jeep on a highway,"YouTube, Jul., 2015. [Video file]. Available: https://www.youtube.com/watch.v=MK0SrxBC1xs. [Accessed: Jun. 5, 2020].
[44] S. Takeshi, B. Cyr, S. Rampazzi, D. Genkin, K. Fu, Light commands: laser-based audio injection attacks on voice-controllable systems, Nov. (2019) [Online] Available, https://light.commands.com/20191104-Light-Commands.pdf [Accessed: Jun. 20, 2020].
[45] K.K.E. Silva, Vigilantism and cooperative criminal justice: is there a place for cybersecurity vigilantes in cybercrime fighting? J. Int. Rev. Law Comput. Technol. (2018) 21–36.
[46] Bitdefender, "Silex malware wrecks 2,000 IoT devices in four hours," Jun., 2019. [Online]. Available: https://www.bitdefender.com/box/blog/iot-news/silex-mal.ware-wrecks-2000-iot-devices-four-hours. [Accessed: Jun. 5, 2020].
[47] J. David, "Massive cyber-attack 'sophisticated, highly distributed', involving millions of IP addresses," CNBC, para. 3-7, Oct., 2016. [Online]. Available: https://www.cnbc.com/2016/10/22/ddos-attack-sophisticated-highly-distributed-involved-millions-of-ip-addressesdyn-html. [Accessed: Jun. 5, 2020].
[48] Talos Group, "New VPNFilter malware targets at least 500K networking devices worldwide," May, 2018. [Online]. Available: https://blogs.cisco.com/security/talos/vpnfilter. [Accessed: Jun. 15, 2020].
[49] D. Goodin, "Active attack on Tor network tried to decloak users for five months," Jul., 2014. [Online]. Available: https://arstechnica.com/information-technology/2014/07/active-attack-on-tor-network-tried-to-decloak-users-forfive-months. [Accessed: Jun. 20, 2020].
[50] Computer Security and Industrial Cryptography. "COSIC researchers hack tesla model S key fob," YouTube, Sept., 2018. [Video file]. Available: https://www.youtube.com/watch.time_continue=1&v=aVlYuPzmJoY&feature=emb_logo. [Accessed: June 5, 2020].
[51] G. Wallace, "HVAC vendor eyed as entry point for Target breach," February 7, 2014. [Online]. Available: https://money.cnn.com/2014/02/06/technology/security/target-breachhvac/index.html. [Accessed: Jul. 29,2020].
[52] J. Wurm, K. Hoang, O. Arias, A. Sadeghi and Y. Jin, "Security analysis on consumer and industrial IoT devices," 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC), Macau, 2016, pp. 519-524.
[53] W. Julian Okello, Q. Liu, F. Ali Siddiqui and C. Zhang, "A survey of the current state of lightweight cryptography for the Internet of things," 2017 International Conference onComputer, Information and Telecommunication Systems(CITS), Dalian, 2017, pp. 292-296.
[54] C.A. Lara-Nino, A. Diaz-Perez, M. Morales-Sandoval, Elliptic curve lightweight cryptography: a survey, IEEE Access 6 (2018) 72514–72550.
[55] N. Krzyworzeka, "Asymmetric cryptography and trapdoor one-way functions," Automatics. vol. 20, pp. 39-51.
[56] S. Chandra, S. Paira, S. S. Alam and G. Sanyal, "A comparative survey of symmetric and asymmetric key cryptography," 2014 International Conference onElectronics, Communication and Computational Engineering (ICECCE), Hosur, 2014, pp. 83-93.
[57] O. P. Piñol, S. Raza, J. Eriksson and T. Voigt, "BSD-based elliptic curve cryptography for the open Internet of Things," 2015 7th International Conference on New Technologies, Mobility and Security (NTMS), Paris, 2015, pp. 1-5.
[58] P. G. Spirakis, I. Chatzigiannakis, A. Pyrgelis and Y. C. Stamatiou, "Elliptic curve based zero knowledge proofs and their applicability on resource constrained devices," Proc. - 8th IEEE Int. Conf. Mob. Ad-hoc Sens. Syst. MASS, 2011.
[59] T. Backenstrass, M. Blot, S. Pontie and R. Leveugle."Protection of ECC computations against side-channel attacks for lightweight implementat," In IEEE International Verification and Security Workshop (IVSW), 2016.
[60] T. K. Goyal and V. Sahula, "Lightweight security algorithm for low power IoT devices," 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, 2016, pp. 1725-1729.
[61] M. A. Philip and Vaithiyanathan, "A survey on lightweight ciphers for IoT devices," 2017 International Conference on Technological Advancements in Power and Energy (TAP Energy), Kollam, 2017, pp. 1-4.
[62] A. Bogdanov, G. Leander, C. Paar, A. Poschmann, M.J. Robshaw, Y. Seurin, Hash functions and RFID tags: mind the gap. International Workshop on Cryptographic Hardware and Embedded Systems, Springer, 2008, pp. 283–299.
[63] B.T. Hammad, N. Jamil, M.E. Rusli, M.R. Z`aba, A survey of lightweight cryptographic hash function, Int. J. Sci. Eng. Res. (7) (2017) 806–814, 8July.
[64] I. K. Dutta, B. Ghosh and M. Bayoumi, "Lightweight cryptography for internet of insecure things: a survey,"2019 IEEE 9th Annual Computing and CommunicationWorkshop and Conference (CCWC), Las Vegas, NV, USA, 2019, pp. 0475-0481.
[65] S. Naoui, M. E. Elhdhili and L. A. Saidane, "Security analysis of existing IoT key management protocols," IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Agadir, pp. 1-7, 2016.
[66] M. El-hajj, A. Fadlallah, M. Chamoun, A. Serhrouchni, A survey of Internet of Things (IoT) authentication schemes, Sensors 19 (5) (Mar. 2019) 1141.
[67] G. E. Suh and S. Devadas, "Physical Unclonable functions for device authentication and secret key generation," 2007 44th ACM/IEEE Design Automation Conference, San Diego, CA, 2007, pp. 9-14.

[68] H. Nicanfar, P. Jokar, K. Beznosov, V.C.M. Leung, Efficient authentication and key management mechanisms for smart grid communications, IEEE Syst. J. 8 (2) (June 2014) 629–640.
[69] I. A. B. Adames, J. Das and S. Bhanja, "Survey of emerging technology based physical unclonable functions," International Great Lakes Symposium on VLSI (GLSVLSI), Boston, MA, pp. 317-322, 2016.
[70] G. Saldamli, L. Ertaul and A. Shankaralingappa, "Analysis of lightweight message authentication codes for IoT environments," 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC), Rome, Italy, 2019, pp. 235-240.
[71] G.L.N. López de Lacalle, J. Posada, Special issue on new industry 4.0 advances in industrial IoT and visual computing for manufacturing processes, J. Appl. Sci. 9 (2019) 4323.
[72] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, P. Faruki, Network intrusion detection for IoT security based on learning techniques, IEEE Commun. Surv. Tutor. 21 (3) (2019) 2671–2701, third quarter.
[73] L. Xiao, X. Wan, X. Lu, Y. Zhang, D. Wu, IoT security techniques based on machine learning: how Do IoT devices use AI to enhance security? IEEE Signal Process. Mag. 35 (5) (Sept. 2018) 41–49.
[74] H. Dai, Z. Zheng, Y. Zhang, Blockchain for Internet of Things: a survey, IEEE Internet Things J. 6 (5) (Oct. 2019) 8076–8094.
[75] P. Williams, P. Rojas and M. Bayoumi, "Security taxonomy in IoT – a survey," 2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS), Dallas, TX, USA, 2019, pp. 560-565.
[76] B. Aboushosha, R.A. Ramadan, A.D. Dwivedi, A. El-Sayed, M.M. Dessouky, SLIM: a lightweight block cipher for internet of health things, IEEE Access 8 (2020) 203747–203757.
[77] A. Ometov et al., "Feasibility characterization of cryptographic primitives for constrained (wearable) IoT devices," 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), Sydney, NSW, 2016, pp. 1-6.
[78] A.K. Jadoon, L.C. Wang, T. Li, M.A. Zia, Lightweight cryptographic techniques for automotive cybersecurity, J. Wireless Commun. Mobile Comput. 2018 (2018).
[79] G. Chu, N. Apthorpe, N. Feamster, Security and privacy analyses of internet of things Children's toys, IEEE Internet Things J. 6 (1) (Feb. 2019) 978–985.
[80] Federal Trade Commission, "Using IP cameras safely," [Online]. Available: https://www.consumer.ftc.gov/articles/0382-using-ip-cameras-safely. [Accessed: Nov. 28, 2020].
[81] Device Authority, "Securing a Connected/IoT Medical Device: a guide for device manufacturers and medical professionals." 2018. [Online]. Available: https://www.deviceauthority.com/sites/deviceauthority/files/medical_device_insight_guide_2018.pdf. [Accessed: Oct. 12, 2020].
[82] Google Nest, "Google Nest Wifi security features." [Online]. Available: https://support.google.com/googlenest/answer/9547625?hl=en. [Accessed: Oct. 12, 2020].
[83] August Smart Lock. [Online]. Available: https://august.com/products/august-smart-lock-3rd-generation. [Accessed: Nov. 28, 2020].
[84] L. Luo, Y. Zhang, B. Pearson, Z. Ling, H. Yu, X. Fu, On the security and data integrity of low-cost sensor networks for air quality monitoring, J. Sensors 18 (2018) 4451.
[85] T. Gebremichael, et al., Security and privacy in the industrial internet of things: current standards and future challenges, IEEE Access 8 (2020) 152351–152366.
[86] Federal Trade Commission (FTC), "How to secure your voice assistant and protect your privacy." February 2020. [Online]. Available: https://www.consumer.ftc.gov/articles/how-secure-your-voice-assistant-and-protectyourprivacy#secure. [Accessed: Jul. 13,2020].
[87] N. Sklavos, O. Koufopavlou, Mobile communications world: security implementations aspects – a state of the art, CSJM J. Inst. Math. Comput. Sci. 11 (32) (2003) 168–187.
[88] N. Sklavos, K. Touliou, and C. Efstathiou. Exploiting cryptographic architectures over hardware Vs . software implementations: advantages and Trade-Offs 2 software security limitations. In Proceedings of the 5th WSEAS International Conference on Applications of Electrical Engineering, volume 2006, pages 147–151, 2006.
[89] T. Miyachi, Protecting industrial control systems, J. Inst. Electric. Eng. Japan 132 (6) (2012) 354–358.
[90] V. K. Jha. Cryptanalysis of Lightweight Block Ciphers. Master's thesis, Aalto University, 2013.
[91] P. Jorgensen. Applied cryptography: Protocols, algorithm, and source code in C, volume 13. John Wiley and Sons Inc., New York, 2nd edition, 1996.
[92] I. K. Dutta, B. Ghosh, and M. Bayoumi. Lightweight cryptography for internet of insecure things: a survey. In 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019, pages 475–481, 2019.
[93] B. Cambou. A XOR data compiler: Combined with physical unclonable function for true random number generation. In Proceedings of Computing Conference 2017, volume 2018 -Janua, pages 819–827, 2018.
[94] B. Cambou. Multi-factor authentication using a combined secure pattern, 2015.
[95] P. C. Kocher. Timing attacks on implementations of diffie-hellman, RSA, DSS, and other systems. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 1109, pages 104–113, 1996.
[96] L.A. Tawalbeh, H. Houssain, T.F. Al-Somani, Review of Side Channel Attacks and Countermeasures on ECC, RSA, and AES Cryptosystems, J. Internet Technol. Secur. Trans. (JITST) 5 (2016) 515–525, pages.
[97] Francois-Xavier Standaert. Introduction to side-channel attacks. In Secure Integrated Circuits and Systems. pp. 27-42, 2010.
[98] A. Carlson, P. Doherty, I. Eichen, and J. Gall. Using collisions to break CBC. In ShowMeCon, 2016.
[99] I. K. Dutta, B. Ghosh, A. H. Carlson, and M. Bayoumi, "Lightweight polymorphic encryption for the data associated with constrained internet of things devices. In IEEE World Forum on Internet of Things, WF-IoT 2020 - Symposium Proceed-ings. Institute of Electrical and Electronics Engineers Inc., jun 2020.
[100] H Bar-El. Security implications of hardware vs. software cryptographic modules. In Discretix White Paper, pages 1–3, 2002.
[101] Hardware Random Number Generators | Blogs. (n.d.). Retrieved December 9, 2021, from https://cerberus-laboratories.com/blog/random_number_generators/.
[102] Quantum Random Number Generation (QRNG) - ID Quantique. (n.d.). Retrieved December 9, 2021, from https://www.idquantique.com/random-number-generation/overview/.
[103] Shokri, R., M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in Proc. 2017 IEEE Symp. Security Privacy, pp. 3–18, 2017.
[104] Papernot, Nicholas, "A Marauder's map of security and privacy in machine learning," arXiv:1811.01134 [cs], 2018.
[105] M. Barreno, Blaine Nelson, D. Anthony, J.D. Joseph, The security of machine learning, Mach. Learn. 81 (2) (2010) 121–148.
[106] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, and M. Young. "Machine learning: the high interest credit card of technical debt.", 2014.
[107] M. T. Ribeiro, S. Singh, and C. Guestrin. "Anchors: High-precision model-agnostic explanations." In Thirty-Second AAAI Conference on Artificial Intelligence, 2018.
[108] S. Alfeld, X. Zhu, P. Barford, "Data poisoning attacks against autoregressive models." AAAI Conference on Artificial Intelligence, North America, 2016.
[109] McGraw, Gary, Richie Bonett, Harold Figueroa, and Victor Shepardson. "Securing engineering for machine learning," IEEE Comput., Volume 52, Number 8, pages 54-57.
[110] Yuan, Xiaoyong, Pan He, Qile Zhu, and Xiaolin Li, "Adversarial examples: attacks and defenses for deep learning." IEEE Transactions on Neural Network Learning Systems, pp. 1–20, 2019.
[111] Shane, Janelle, You look like a Thing and I Love You, Voracious, 2019.
[112] G. Leurent T. Peyrin, "SHA-1 is Shambles," (2020).
[113] K. Lei, Q. Zhang, L. Xu and Z. Qi, "Reputation-based byzantine fault-tolerance for consortium Blockchain," 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), 2018, pp. 604-611.

[114] S. W. Kim. (May 24, 2018). Safety and liveness—blockchain in the point of view of FLP impossibility. [Online]. Available: https://medium.com/codechain/safety-and-liveness-blockchain-inthe-point-of-view-of-flp-impossibility-182e33927ce6, [Accessed Dec. 5, 2021].
[115] S. Singh, A.S.M.S. Hosen, B. Yoon, Blockchain security attacks, challenges, and solutions for the future distributed IoT network, IEEE Access 9 (2021) 13938–13959.
[116] N.A. Akbar, A. Muneer, N. ElHakim, S.M. Fati, Distributed hybrid double-spending attack prevention mechanism for Proof-of-Work and Proof-of-Stake Blockchain consensuses, Future Internet 13 (2021) 285.
[117] K. Soska, A. Kwon, N. Christin, S. Devadas, 'Beaver: A decentralized anonymous marketplace with secure reputation, IACR Cryptol. ePrint Arch. 2016 (2016) 464–479.
[118] A. Katrenko, Mihail, S."Blockchain attack vectors: vulnerabilities of the most secure technology", May 6, 2020, [Online]. Available: https://www.apriorit.com/dev-blog/578-blockchain-attack-vectors. [Accessed: Dec. 7, 2021].
[119] N. F. Samreen and M. H. Alalfi, "SmartScan: an approach to detect denial of service vulnerability in Ethereum smart contracts," 2021 IEEE/ACM 4th International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), 2021, pp. 17-26.
[120] Azure IoT. [Online]. Available: https://docs.microsoft.com/en-us/azure/iot-fundamentals/iot-security-deployment. [Accessed: Nov. 28, 2020].
[121] Amazon Web Services IoT (AWS-IoT). [Online]. Available: https://docs.aws.amazon.com/iot/latest/developerguide. [Accessed: Nov. 28, 2020].
[122] Google Cloud IoT. [Online]. Available: https://cloud.google.com/blog/products/gcp/securing-cloud-connected-devices-with-cloud-iot-and-microchip. [Accessed: Nov. 28, 2020].
[123] ThingsBoard and DeviceHive. [Online]. Available: https://www.hamk.fi/wpcontent/uploads/2019/03/Project-Report_bitencourt_anjos.pdf. [Accessed: Nov. 20, 2020].
[124] D. Díaz-López, M. Blanco Uribe, C. Santiago Cely, D. Tarquino Murgueitio, E. Garcia Garcia, P. Nespoli, F. Gómez-Mármol, Developing secure IoT services: a security-oriented review of IoT platforms, Symmetry 10 (12) (Nov. 2018) 669.
[125] Security in DeviceHive. [Online]. Available: https://docs.devicehive.com/docs/security-in-devicehive. [Accessed:Nov. 28, 2020].



Phillip Williams received B.Sc. in Electrical Engineering in 2008 and M.Sc. with an Electrical Engineering Concentration in 2010. He is a member of the VLSI lab at the University of Louisiana at Lafayette His research work focuses on Hardware Security for resource constrained Internet of Things (loT) devices, Cybersccurity, and Authentication. He is currently working on Physically Unclonable Functions (PUFs) and their resilience against modeling attacks.

Indira Kalyan Dutta received her BSE degree from American Internationl University-Bangladesh in Electrical and Electronics Engineering in 2012. She joined the VLSI lab in University of Louisian at Lafayette as a Graduate Student in 2013 and received her MS degree in Computer Engineering in 2015. Her research work focuses on Hardware Security,Cryptography and Internet of Things (loT). She received her PhD degree in Computer Engineering from University of Louisiana at Lafayette in 2021. In her PhD thesis, she pioneered an implementation of a Novel architecture of Lightweight Polymorphic Encryption system for constrained loT devices

Hissham Daoud received the B.sc. and M.sc. degree from Cairo University Giza Egypt in 2004 and 2007 respcetiveiy, and the Ph.D. degree from Ain Sham University, Cairo, Egypt in 2014,all in electronics and communications enginecering. Since 2004 he has held multiple positions in both industry and academia. He is currently with the University of Louisiana at Lafayette, LA, USA. His research interests include blomedical signal processing, machine learning, deep learning, and neuromorphic computing. Dr. Daoud wae the recipient of the IEEE CASS Student Travel Award, the Charles Desoer Award in IEEE Biomedical CriCuits and Systems Conference (BioCAS 2019), and the Best Paper Award in the 14th IEEE Colloquium on Signal Processing and its Applications conference (CSPA 2018). He has served as a Reviewer for several IEEE conferences and journals

Magdy Bayoumi (Life Fellow,IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from Cairo University, Egypt, the M.Sc. degree in computer engineering from Washington University at St. Louis, and thye Ph.D. degree in electrical engineering from the University of Windsor ON. He is currently the Department Head of the Electrical and Computer Engineering Department, University of Louision at Lafayette, Lafayette, LA, USA. His research interests include VLSI design and architectures, digital signal processing, and wireless and hoc and sensor network. He was a recipient of the 2009 IEEE Circuits and Systems Meritoricus Service Award and the IEEE Circuits and Systems Society 2003 Education Award. He was the Vice President for Conferences of the IEEE Circuits and Systems Society