Cyber-physical security for IoT networks: a comprehensive review on traditional, blockchain and artificial intelligence based key-security

Ankit Attkan¹ · Virender Ranga¹

Received: 25 September 2021 / Accepted: 28 January 2022 / Published online: 24 February 2022
© The Author(s) 2022

Abstract

The recent years have garnered huge attention towards the Internet of Things (IoT) because it enables its consumers to improve their lifestyles and professionally keep up with the technological advancements in the cyber-physical world. The IoT edge devices are heterogeneous in terms of the technology they are built on and the storage file formats used. These devices require highly secure modes of mutual authentication to authenticate each other before actually sending the data. Mutual authentication is a very important aspect of peer-to-peer communication. Secure session keys enable these resource-constrained devices to authenticate each other. After successful authentication, a device can be authorized and can be granted access to shared resources. The need for validating a device requesting data transfer to avoid data privacy breaches that may compromise confidentiality and integrity. Blockchain and artificial intelligence (AI) both are extensively being used as an integrated part of IoT networks for security enhancements. Blockchain provides a decentralized mechanism to store validated session keys that can be allotted to the network devices. Blockchain is also used to load balance the stressing edge devices during low battery levels. AI on the other hand provides better learning and adaptiveness towards IoT attacks. The integration of newer technologies in IoT key management yields enhanced security features. In this article, we systematically survey recent trending technologies from an IoT security point of view and discuss traditional key security mechanisms. This article delivers a comprehensive quality study for researchers on authentication and session keys, integrating IoT with blockchain and AI-based authentication in cybersecurity.

Keywords Authentication · Authorization · Session key · Key management system · Key distribution · Artificial intelligence · Blockchain · Sensor networks · IoT security · Cyber-physical systems

Introduction

Over the last decade of technological enhancements in the area of IoT security, it is observed that the number of attacks and types of attacks has risen exponentially as compared to earlier days of IoT technology. Edge devices have grown smarter and powerful in terms of receptibility of instructions and computational power. However, as the computational power and resourcefulness are increased using advanced boards like Arduino [1], Raspberry Pie [2] and Nvidia Jetson Xavier [3], the human greed for perfor-

mance and efficiency also rises and it results in the term “resource-constrained” become persistent in IoT networks. Rising technologies give rise to numerous ways for attackers to exploit the edge devices and network. It is then obvious to run heavier security mechanisms and algorithms supporting stronger cryptographic keys for better security. An IoT network of heterogeneous nodes is vulnerable to a variety of attacks and exploit threats, because of the varying communicational technical standards and data formats being handled by these devices. A gateway node may perform smoothly for a complex algorithm whereas the smaller computing devices may collapse due to their limited resources [4]. There are numerous vulnerabilities in the network and these act as an opportunity point for an attacker to exploit. For instance, multiple heat-sensing sensor nodes are deployed in mountain terrain on a battlefield to track down the hidden threat soldiers. A technically skilled highly trained soldier might capture one of the sensor nodes and exploit its vulnerabilities

✉ Ankit Attkan
ankit_62000070@nitkkr.ac.in
Virender Ranga
virender.ranga@nitkkr.ac.in

¹ Department of Computer Engineering, National Institute of Technology, Kurukshetra 136119, India

to bring down the whole network of sensors to turn the odds against the enemy. Sensor capture attack and session key Hijacking are discussed in detail in Sect. 4 of this article. There is a need for improvement in IoT security from a wide area of applications point of view as IoT is applicable in almost every field from agriculture [5, 168], healthcare [6], smart home deliveries [7] to battlefields as discussed earlier. Security of devices can be maintained on a certain metric of “Trust”. Just like real people, who prefer building trust among each other before sharing sensitive information, IoT network components like sensor nodes, gateways, edge devices, actuators, etc., also need to establish a verifiable trust among each other before sharing data. Now, how to build this so-called “Trust” among non-living machine hardware? Authentication parameters and proof of possessing the security entity is one way of many other ways. Suppose symmetric key cryptography [8] is being used for two IoT devices to communicate, and these keys were provided by the IoT network administrator during the registering process of that particular node. Now the proof that both these peers trying to communicate with each other legitimate and safe to share data depends on whether these devices possess the same symmetric key which was distributed by the administrator. One peer encrypts a certain randomly chosen number with the symmetric key and the other peer performs decryption using its key. Now if the decryption is successful then the proof of “Trust” is established among the two peers. Then it is by default assumed that data sharing is safe after authentication. In any scenario, the authorization of IoT network devices can be only done after successful authentication takes place between communicating nodes. By using the term “authentication” we mean to convey and keep intact the following:

- Authentication is a process that validates a device that is trying to communicate via messages or transactions with another device is legitimate, authentic, trustworthy of sharing sensitive information.
- Any information exchanged among peers as sent and received is exactly from the source it claims to be.
- Authentication ensures that there are no third-party intermediates that may generate vulnerabilities. Two devices mutually authenticate each other at a particular time duration, say, nodes A, B, and C want to authenticate each other, then any combination pair of only two nodes will authenticate at a time to avoid third-party intervention.
- Authentication means ability/proof of trust among peer-to-peer or machine-to-machine interaction.

IoT networks run on an architecture that consists of three main layers which are the AL-application layer, ML-middleware layer, and PL-perceptron layer or also known as edge layer. The modern-day architecture of IoT is discussed in detail in Sect. 6 of this article. The edge layer constitutes the

components like sensors, RFID tags, actuators, etc. Deployment of perceptron layer edge devices can be done in two ways, random and regular deployment. Regular deployment is where there is a discrete constant distance is kept among the devices spread over a geographical area in uniformity. In random deployment, there are irregular distances among the edge layer devices, thus there are varying signal strength and network range issues in farther located devices whereas there is good connectivity observed in devices nearer to each other. Some of the major challenges while building an IoT network involves:

- *Heterogeneous components*: managing and exchanging various formats of information among heterogeneous devices is a major challenge. Such IoT edge devices operate on varying platforms, utilizing different protocols to authenticate, authorize, share data and communicate with each other. It is a challenge to manage heterogeneous devices when the number of nodes is large [9].
- *Scalability of the network size*: it has issues related to information management, device re-location, deploying newer edge devices into an older IoT network with different keys to authenticate each other while maintaining forward and backward secrecy [10].
- *Sharing data*: It includes challenges relevant to availability, mutual authentication, network traffic congestion, malware attack vulnerabilities, and passive attacks [11].
- *Sustainable (efficient) IoT algorithms*: due to hardware variation, some devices perform optimally while others underperform due to load congesting over low computational edge devices. Most of the energy is wasted to handle network congestion and selecting better converters to convert one data format to another. Constructing energy-efficient session key generation and distribution algorithms, that perform better with higher securities under lower computational overhead is a challenge. The challenge is to minimize the usage needed for communication among various edge devices of an IoT network aiming for minimum wastage of battery power [12].
- *Network component tracing capability* network components in a sensor node deployed a cluster of edge devices are always vulnerable to be stolen physically by the attacker to obtain data collected and stored in the device. This occurs mostly in cases of wireless sensor networks where sensors with less to no security are deployed. Hardware security for such devices is less to minimize the deployment cost of the IoT network. The edge device must be identified using their IDs allotted during the registration phase and tracing of such edge devices is a challenge while maintaining the connectivity range. If the device goes out of range it has to be discarded for security purposes when Ad-Hoc networks are observed [13].

Our contribution

In this article, we have categorized the present authentication protocol-based schemes, IoT key generation, distribution, and management frameworks. It provides a systematic comprehensive survey for a variety of researchers in the field of Internet of things security. This article is constructed in a way to provide a basic background of the technologies involved (blockchain and AI) in integration with IoT, then dive into the security vulnerabilities, attacks, and issues in security while addressing some of the possible solutions to the research problems identified. The major contribution of this review article can be summarized as follows:

- It provides a concise introduction which gives the readers a basic overview of the research area, the role of authentication in IoT security, and discusses very briefly the challenges related to IoT deployment.
- It provides a wide taxonomy of IoT attacks and classifies these attacks into broad categories. Discusses these attacks that are still persistent throughout the initiation, deployment, maintenance, and termination phases of the IoT networks.
- This article distinguishes between authentication, authorization, and trust metric to clear the mythical assumptions related to terminologies around fresh scholars by providing a better understanding of the concepts.
- It identifies the major session and authentication key-oriented attacks and discusses in detail how these attacks are performed by the adversary to exploit the numerous vulnerabilities to compromise the network.
- This article discusses the traditional key generation and distribution systems, their perks, and their limitations towards IoT attacks. Provides a comparison between various traditional authentication frameworks. It gradually proceeds with the basics of Blockchain and Artificial Intelligence technologies for fresh readers and then provides their integrated role in the Internet of things security.
- The work identifies the shortcomings that still exist in the present authentication mechanisms based on trending blockchain and artificial intelligence technologies in IoT security.
- The article provides possible future directions for IoT network security boosted by blockchain and AI.

Organization of the article

This article aims to provide a basic background of technologies being discussed in each section before discussing the core components, problems, solutions, and the limitations related to them. In "Attacks on IoT devices", we provide the taxonomy of the huge number of attacks that are found in vulnerable IoT networks. "Attacks on IoT

devices" also highlights certain categories of cryptosystems that are used frequently in IoT security to overcome device/node/peer/gateway authentication issues. "IoT devices & their mutual-authentication, authorization, trust metric" discusses the role of authentication and why it is different from authorization to clear terminological misunderstandings among new researchers, it also provides the working mechanism of authentication keys and authorization procedure for peer-2-peer devices. "IoT edge level session keys: vulnerabilities, attacks and preventive solutions" discusses how an attacker exploits the whole IoT network by compromising the session keys and impersonating fake identities to breach network security. "Internet of things and blockchain: basics, IoT keys and authentication" introduces blockchain concepts, its integrated use in IoT edge networks for security enhancements, and ways this technology is being used to authenticate devices, securely store data/keys, and their vulnerabilities towards identified attacks. "Artificial intelligence and IoT: architecture, attack analysis, and learning security" introduces the integration of Artificial Intelligence in IoT security and IoT key management systems, their advantages, and limitations towards certain attacks identified during the survey. The article discusses future directions in "Future scope of improvement and conclusion" and Sect. 8 concludes the article followed by significant references.

Attacks on IoT devices

The advantages of IoT networks have their impact on smart technology adaptations in human lives from offices to homes and everywhere else. They are used frequently in day-to-day lives as well as in industries, to make things easier, smarter, and simpler to interact with. IoT technology is smarter and saves time and resources. The emergence of IoT has brought up new opportunities in the field of health care, transportation, smart home, farming, etc., and bringing up new opportunities for innovation and communication technologies between entities. The fast development of IoT and adopting the technology in daily lives require addressing the security threats in this field. The different attacks are discussed in Fig. 1 showing the attack taxonomy in the IoT field.

The IoT network architecture supports different devices and protocols for the devices to communicate. Therefore, based on the layer, type of communication, and device connected attacks possess different behaviour by exploiting the vulnerabilities in the associated entities.

Physical attacks

The physical attacks are associated with the hardware devices in the network affecting the physical functionality of the system.

Fig. 1 Attack taxonomy in internet of things network (device and network perspective)

1. *Micro-probing* This is a kind of invasive attack that requires physical manipulations on semiconductors. The microscopic needles are attached to the internal wiring of the chip to read the internal secrets of the chip and to carry out fault attacks [14].
2. *Reverse Engineering* Re-coding and manipulating the algorithm or complete source code of security architecture to reveal sensitive information [15].
3. *Sensor Capture* The attacker gains access to the sensor device for compromising the network, removes the node from the network, and redeploys it as malicious nodes in the network [16].
4. *Physical Communication Channel tapping* The attacker uses channel tapping to extract the signals from the channel, without breaking the connection. The inconvenience may be caused due to diverting of traffic by the attackers [17].
5. *Physical damaging* This sort of attack tampers the sensor devices. The sensor devices are often left unattended to operate in outdoor environments like in farming sectors that often cause physical damage to the devices [18].
6. *Stolen smart card* When a smart card gets lost, the attacker can easily guess the password by applying an

offline password-guessing technique, which is possible due to the smart card's fixed output for the same input [19].

7. *Biometric fingerprint recreation* The attackers gain access to the system and steal the biometric template on the device for manipulating the stored data of users and implementing the false acceptance attacks posing a serious risk to the system [20].

Side-channel attacks

Side-channel attacks are associated when a security exploit gathers information from the influence of program execution by measuring or exploiting the indirect effects of the system instead of targeting the program directly.

1. *Time Analysis* The time analysis side-channel attack occurs when an attacker observes the execution time taken by the operations of a cryptosystem. The precise measurements of time for each operation allow the attacker to find the inputs for the system which is an easier task than cryptanalysis and brute-force attack [21].

2. *Power Analysis* The physical properties of a device are studied by the attackers to calculate the power consumption of a hardware device. The methodology is governed by different laws of physics, voltage fluctuations, semi-conductor properties of the device, and many other factors. It also involves the calculation of power traces or constructing a graph of electrical activity against time [22].
 3. *Fault Analysis* The attackers conduct this active side-channel attack by inducing faults in the implementation of the cryptographic operation to find its internal state. For instance, inducing the processor to a high voltage or high-temperature state, which results in false output, allowing the attacker to deduce the instructions of the processor and its internal behaviour [23]
 4. *Electromagnetic Analysis* The motive behind conducting the electromagnetic analysis is to capture the encryption keys being used in the cryptographic functions. The electromagnetic signals emitted from the device are captured and analyzed by the attackers to carry out this attack [24].
 5. *Denial of Service* The DoS in side-channel attacks occurs due to different parameters that are considered for measuring the performance in cryptanalysis. The timing attack tends to slow down the process by flooding the network with signals which results in shutting down of the network [25].
 6. *Malicious Code Injection* The information leakage from the side channels allows the malicious user to retrieve the secrets and thereby, running unprivileged code under attack or exploiting remote code execution vulnerabilities [26].
 7. *Sleep Denial* This attack affects each sensor node in the network by forcefully sending the nodes in the sleep mode and hence, affecting the network performance [27].
 8. *Fake Node Injection* The attacker adds a fake node in the network which intends to inject malicious code in the network and consume the power of other sensor nodes by keeping them busy which in turn collapses the network. The fake node also acts as a man-in-the-middle to gather other secrets and information [28].
- text, still having some knowledge about the language or statistical distribution of characters in the plain text [29].
2. *Known plain text (KPA)* The KPA occurs when the cryptanalyst has access to both plain text and the ciphertext. This attack model is used to reveal the secret keys used for encryption and codebooks [30]
 3. *Chosen Plain text (CPA)* The CPA generally assumes that the attacker is capable of calculating the ciphertext for any arbitrary plain text to reduce the security of the cryptosystem. In this model, the adversary interacts with the encryption machine (viewed as a black box) to reveal the encryption key [31].
 4. *Man-in-the-Middle (MITM)* The model has associated with active eavesdropping and intercommunication between entities believing that they are communicating with each other. The attacker acts as a third party between the entities without being noticed and makes an individual connection with the victim and controls the entire communication between them. The attacker intercepts the messages and injects new messages to circumvent the mutual authentication between two parties [32].
 5. *Replay Attack* A replay attack (playback attack) is a passive attack performed by delaying the data transmission maliciously or fraudulently. The attack can be performed by either of the parties, that is, the originator or an adversary intercepting the data and re-transmitting it after modification [33].
 6. *Sybil Attack* The attacker tries to control the network by creating multiple fake identities to control the network. In sensor networks, a fake sensor node with true identity communicates with other neighbouring nodes and distributes multiple identities which are not lawful. The fake identities degrade the system's reliability, data integrity, and security [34].
 7. *Key Hijacking* When key hijacking (often called session hijacking) is performed to gain unauthorized access to devices in a network. This technique is used to provide access for the remote server to an unknown user [35].
 8. *Brute force* The hit and trial techniques are common in guessing easy passwords. The attacker performs multiple attempts submitting passwords and paraphrases systematically to guess the security key created using the key derivation function [36].
 9. *DoS/DDoS* For cryptanalysis, the DoS attack is observed through hash collisions exploiting the worst-case runtime of looking up hash tables. The system time is consumed for searching instead of performing critical tasks [37].

Cryptanalysis attacks

The attacker uses different techniques to break the cryptosystem of a network. A similar technique is the cryptanalysis attack technique in which attackers do a thorough study of the cryptographic technique to determine the weak point of the system. The attacks rely on the nature of the algorithm and the general characteristics of the plaintext.

1. *Cipher Text only* This attack model is used by the attacker when it has access only to the ciphertexts and not to plain

Software attacks

The devices are not only the target for hackers but also to gain system access and run unprivileged code on the system to

perform various suspicious activities that hinder the normal operation of the system.

1. *Virus* The virus is a program code or a macro that attaches itself to a legitimate code to get executed. The virus program doesn't execute till it gets the favourable circumstances for its code. A virus has damaging effects on the system software caused due to data corruption [38].
2. *Worms* To execute an attack, attackers make use of standalone software which propagates themselves without any human intervention of other host software. The device is targeted by exploiting the vulnerability making use of system transport services to execute through multiple sensor devices [39].
3. *Trojan horses* It is a software program that looks legitimate and gets into execution with the help of user interaction (email attachment or file downloading). The attackers trick the users to load and execute it on their systems resulting in damage to the host machine by stealing data, corrupting files, and supporting other malware spread [40].
4. *Spyware* These software aims to collect critical information of an individual or organization, pretending to be legitimate and sending the information to a third-party entity without the user's knowledge. This technique allows gaining access to the device without the consumer's consent [41].
5. *Adware* The software that demands "user clicks" to generate revenue through advertisements present on a legitimate web page. Once the user clicks on the advertisement a backdoor is installed on the system stealing sensitive data [42].
6. *Key-logger* The user credentials are often stolen by the attacker using a program that records keystrokes of the keyboard. All the typed information can be retrieved and passwords are accessed using key-logging programs [43].
7. *Logic Bomb* The software is injected with a certain piece of malicious code and executes its payload and functionality after fulfilling a certain condition [44].
8. *Malware* The attackers seek information of the network to find out its vulnerabilities to execute malicious software known as malware that infects the system to steal confidential information, destroying data, or disabling the network. The known vulnerabilities are exploited for malicious software to execute and propagate through the system and then through the network [45].
9. *Root Exploit attack* The root exploit software requires the attacker to gain privileges granted by the system administrator to execute a sequence of commands resulting from known vulnerability to harm software and associated devices in the network [46].

10. *Phishing* Humans are tricked by attackers by sending fraudulent messages or emails (spoofed) for deploying malicious software or to retrieve sensitive information. The attacks are operated through legitimate sites as well as observing user's actions to gather more information [47].

Network attacks

The sensor nodes are communicated through the network for sharing data and passing other information about the network. The attackers adopt different advanced techniques to gain access to an individual system or an organizational network. Many possible attacks must be taken addressed in an IoT environment to build defense strategies.

1. *Routing attacks* The attacker defines certain vulnerabilities in the network routing software using which it tries to break the authentication process. The router spoofing attack, routing table amendments with fake entries and redirecting packets from the router take place under the routing attacks [48].
2. *Traffic sniffing* Traffic sniffing involves continuous monitoring of the network by attackers and capturing the packet on an unintended target. The attacker analyses the packet and retrieves the information to gain insight into the devices [49].
3. *Flooding attack* This attacking technique is used to avoid the network to perform its normal functioning. The network is bombarded with a large number of packets and commands to exhaust the network resources in processing those packets [50].
4. *Eavesdropping* It is a kind of sniffing attack associated with retrieving information being transferred among the devices connected through an insecure channel. The information theft occurs due to a lack of security in the channel [51].
5. *Packet cloning* One of the network attacks is a packet replication attack (or) packet clone attack. An adversary can easily capture one packet from the network traffic and extract information from it. Then reprogram it to create a clone of a captured packet. Then these clones can be sent to all network areas, where they can be considered legitimate. The receiver response to the packet disclosing its secrets and other sensitive information [52].
6. *Sinkhole attacks* In this attack, the network traffic is attracted by the compromised nodes by advertising the routing information updates. The attackers benefit themselves by launching other related attacks like spoofing, altering router tables, selective forwarding, and similar attacks [53].

7. *Blackhole attack* The communication in the network is stopped when all the packets get dropped at a particular black hole node. The situation is worsened if the black hole node is also the sink hole node. In such a scenario the traffic flowing through the network is completely stopped and no communication happens [54].
8. *Selective forwarding* This type of attack happens when the ‘neglect and greed’ strategy is followed by the nodes. The nodes ignore the original traffic flowing the network, drop it and instead send their traffic through the network. The malicious nodes cause the Denial of Service by inhibiting the flow of information through the network [55].
9. *IP spoofing* In a communication network, the packets are exchanged between the network nodes having a sender and a receiver of the packet. The IP header in the packet consists of a source address, which during IP spoofing is forged with the purpose to hide the identity or impersonate a device to perform a DDoS attack within the network infrastructure [56].
10. *Masquerading* A masquerade attack is performed to gain unauthorized access to an individual host system or network to attain the information using a legitimate network identity. The attack can be performed by an individual belonging to the organization or an external group when the organization is on a public network [57].
3. *Modification* It is an active attack, based on manipulation and interception of the data being exchanged. The data can be altered by the hackers (attackers) to change the message perception and also dropping off the data from the receivers’ site especially when there is congestion in the network [59].
4. *Obfuscation* The obfuscation is not a direct attack but an attack enabler that provides details or rules or techniques to hide the attack from IDS or other attack detection components dedicated for the purpose. The DoS attack usually fails when strict schema validation (rule validation) is applied in the network [60].
5. *Relocation* Data are migrated from one system/storage to another system/storage. The attackers continuously sniff the communication channel for data and weak security leads to data theft. The attackers extract, transform and duplicate the data to misuse the information for further attacks [61].
6. *Redirection* Redirection is a vulnerability that allows an attacker to trick users of an application to follow a link that leads to an untrusted external site or source. The link to such a site is usually provided on a real webpage or user interface of an application [62].
7. *DoS/DDoS* The intention of the attacker to perform a DoS attack is to disrupt the normal functioning of a network or a system within the network by overwhelming a node with huge data traffic, which makes the server unavailable to provide the service. If multiple compromised machines are used to target the server by sending a request to achieve the effectiveness of the attack by attacking the network infrastructure as well, the scenario is suitable for distributed Denial of Service attack [63].
8. *Insider attacks* This type of malicious attack is performed by the person known as an insider who is authorized for system access. The attack is easy for the insider to operate than an outsider due to the knowledge of the network infrastructure. [64]
9. *SQL Injection* This type of attack finds the web security vulnerability to allow the attackers in a network to inject malicious queries and gain access to the database. The attacker, after the execution of the query, gains access to the data for which he is not authorized to [65].
10. *Data Breach* The intention of the attacker is to retrieve private and confidential information which may result through theft, loss of storage drives, stolen hardware, etc. A data breach may be unintentional which can occur due to accidental corruption of storage media resulting in privacy and data leakage [66].

Data attacks

The IoT devices and network is making an individual give value to convenience over security. The compromise in the security has made the sensitive information the most attractive area for the attacker. The attackers make use of intelligent systems to gain knowledge about the vulnerabilities in the network and system. Due to the vulnerabilities in the system, the threat to the data may lead to several serious attacks.

1. *Data corruption* Any damage to the hardware without the role of an attacker can cause data to be corrupt. The corrupted data when fed to the system results in faulty outputs which are considered unusual for required applications. When the malicious software modifies the files the data loss might lead to system failure. The data corruption caused by the malware might be as a part of their process to execute the payload and overwrites the files with garbage codes which is unusable to be considered for other operations [46].
2. *Data cloning* Data cloning is done by making a digital copy of the data and further using it to perform an attack. For instance, credit card theft in which the card is cloned by the fraudsters using an electronic scanner to perform fraudulent attacks [58].

Fig. 2 Various cryptographic authentication protocols for IoT network

Cryptographic authentication protocols: cryptographic key generation mechanism

The exchange of information in an IoT environment is required for the devices to work in coordination and communicate for any required action such as message passing, starting of the required task (procedure), connection establishment, and more. This information also involves important information about the network, user credentials, content, and devices. The data being transferred needs to be protected using an authentication scheme. The most popular schemes are based on cryptosystems that making use of the concept of keys. In the Fig. 2 different IoT authentication protocols are presented based on the type of cryptosystem used for the protection of data and network.

Symmetric cryptosystem based protocols [67]

The IoT environment allowing multiple devices to communicate with each other follows the symmetric cryptosystem to protect the data from tampering. The requirement for this category of cryptography is that the same key is employed for all the operations, that is, encryption and decryption. These are the main operations of cryptography where all the concept of the keys and key management is necessary. The key that is used to encrypt the data on the sender's end, is the same key that is used to decrypt the data on the receiver's end. There exist different protocols that are used to employ symmetric key cryptography in a system. The key size usually depicts the level of security and management of the cryptosystem that is served by the protocols. There exist various protocols in cryptography, like skipjack where the algorithm of encryption and decryption is based on an 80-bit key for the 64-bit data block. It is regarded as an unbalance Feistel network with 32-rounds. The application of skipjack is for implementing tamperproof hardware making use of a key escrow scheme.

The blowfish block cipher cryptography technique also lies in the same category of the symmetric cryptosystem. The algorithm makes use of fixed size length blocks of 64-bits, allowing padding for the blocks that aren't multiples of 8. The above said 64 bits plaintext message is first divided into 32-bits and the resulting bits XORed with the last two entries of the original bit array, which later are recombined to produce the 64-bit ciphertext. The algorithm is widely used to add data integrity and privacy in the embedded systems. The Twofish cryptographic algorithm also belongs to the family of symmetric cryptosystem supporting block-size of 128-bits and key sizes up to 256-bits. Similar to the earlier blowfish, one half of the n-bit key is used as the actual encryption key and the other half of the n-bit key is used to modify the encryption algorithm. It also has a Feistel network structure which can be used in smart cards, embedded chips, and hardware security. The Serpent algorithm of this family, initially started with 16 rounds to conserve the security approach to counter the known attacks. Later, 32 rounds were implemented with block size 128-bits and key size variants as 128, 192, and 256 bits to encounter the futuristic attacks. The serpent algorithm is suppressed due to the outstanding performance of AES cryptographic encryption using substitution-permutation for producing a ciphertext. The AES algorithm incorporates 10 rounds for a 128-bit key, 12 rounds for a 192-bit key size, and 14 rounds for a 256-bit key. The bytes of the block text are substituted based on rules dictated by predefined substitution boxes. The interesting part of the algorithm is that it internally makes use of hill cipher to jumble up the messages in block's columns. The AES cryptosystem possesses five modes of operation being Electronic Code Book (ECB), Cipher Block Chaining mode (CBC), Cipher Feedback mode (CFB), Output feedback mode (OFB), and counter mode (CTR). The Cast5 algorithm also makes use of S-boxes (substitution boxes) of sizes 8*32-bit based on bent functions, XOR operations, rotations, addition, and subtraction. The

cast5 is also a Feistel network of 12 or 16 rounds bearing a block size of 64-bit and key size of 40 to 128 bits. The super-computing area demands more secure encryption. Therefore, the advanced version of DES (key size 56 bits) is Triple DES (TDES or 3DES) algorithm that applies the DES algorithm to each data block. The TDES algorithm is widely used by vendors, users, and cryptographers.

Asymmetric cryptosystem based protocols [68]

The verification of the messages between the participants (sender and receiver) can be made based on the keys which ensure data integrity, confidentiality, and reliability of data. The symmetric cryptosystem uses only a single key which is shared between sender and receiver, but the asymmetric cryptosystem or public-key cryptosystem makes use of two keys private and public keys. The keys are used for encryption and decryption, that is, if a private key is used for encryption, then the public key will be used for decryption and vice-a-versa. The credibility and verification of the data are made based on digital signatures. In the procedure, first, the sender obtains the public key of the receiver, encrypts the message using its public key, transfers the encrypted data, and the sender can decrypt the data using its private key. This kind of cryptosystem is beneficial in case of non-repudiation so that the sender does not deny sending a message. To implement the asymmetric cryptosystem, there exist numerous protocols that make the task of encryption and decryption more secure and less complex. One such algorithm to achieve a smooth cryptosystem is Diffie-hellman (DH) key exchange. It is known to serve two unknown parties to communicate securely over a public channel through the shared secret key. The communicating parties agree upon a benign piece of information to be mixed with their privileged information. Their secrets are mixed with the public key and exchanged with the information to be shared. As they decipher the other's message, they can extract the public information and with knowledge of their secret, deduce the new information that was carried along. The use of long number strings makes it almost impossible for the eavesdroppers to steal the information even after sufficient resources. The ElGamal cryptography is based on DH key exchange that can be defined on any group of integers modulo n . The ElGamal cryptography is unconditionally malleable and needs modification in the padding scheme if at all required for the security scheme. The computation of any protocol for securing the data decides the level of protection it provides like elliptic curve cryptography (ECC). ECC generates key pairs for public-key cryptography by using the mathematics of elliptic curves possessing algebraic structures over finite fields. ECC features a faster generation of keys and signatures as compared to the RSA algorithm. In the RSA cryptosystem, both the public and private keys are integers and comparatively double the size of ECC to remain

secure, hence making the process of cryptography slower, but easier and simpler than ECC. The RSA algorithm uses the concept of factorization of large prime numbers, the DSA protocol belonging to the asymmetric family, uses discrete logarithm and modular exponentiation. The complex computation of DSA is applicable to be used for digital signature and verification in four operations key generation, key distribution, signing, and signature verification. The resemblance of Rabin cryptosystem security is with RSA cryptosystem based on the difficulty of integer factorization. The Rabin cryptosystem is much secure when applying for a known plain text attack. The complexity of the algorithm lies in the identification of the true plain text if four possible inputs generate each output as ciphertext. Rabin cryptography finds its application in generating digital signatures like RSA and ElGamal algorithms.

Hybrid authentication cryptographic protocols [69]

The maximum efficiency of the cryptosystem can be gained by combining the public key and symmetric cryptosystem to build a hybrid protocol cryptography system. The advantage of an asymmetric cryptosystem of not sharing a common key between sender and receiver is inherited by a hybrid system. In the key encapsulation scheme of the hybrid system, the public and the private key are the same. The hybrid cryptosystem can be constructed either using a key encapsulation scheme as in public-key cryptography or a data encapsulation scheme as in symmetric cryptography. To secure the large messages and data content, the symmetric key is used for encryption or decryption, while for shorter values is encrypted or decrypted using public-key cryptography. The biometric cryptosystem is one of a kind that takes benefits from both the cryptosystems. The advantage of biometric cryptosystem is the convenience of authentication via face, palm, voice, etc. taking the form of a secret key. In this model, users are required to authenticate by matching their biometric with the template on their devices and once it is matched, the device communicates with the service using tokens so biometric information is never transmitted over the wire. Biometric cryptography uses a decentralized model which abandons the risk associated with central biometrics storage. The advanced security systems make use of biometric authentication in addition to username and password credentials. The hashing system is a multistep authentication mechanism that makes use of both biometric and hashing techniques. The randomized token used in the system guards the systems against biometric hacks. The smart token system used in smart cards provides the security and flexibility required by the organizations to authenticate the user for accessing data stored in data centres. The smartcard cryptosystem is a two-step authentication system that stores a user's public key details and a personal identification num-

ber (PIN), which acts as the secret key to authenticate the user to the smart card. The hacks result in collision attacks that are resolved using secure cryptographic hash functions like access polynomial, which constitutes modern cryptography. The data of arbitrary size results in a fixed-sized hash or message digest. It is an irreversible process and cannot be broken easily by attackers. The message authentication codes (MACs), digital signatures, etc. are some of the authentication schemes making use of secure hash cryptographic techniques. Section 3 ahead discusses major aspects of how authentication, authorization is done and from which “trust” is developed among lifeless computational devices.

IoT devices & their mutual-authentication, authorization, trust metric

Numerous attacks are coming into recognition due to technological advancements. Attackers are also becoming capable of using software and hardware-aided exploits to breach IoT security. Earlier, the edge devices were so resource constrained that their mechanism was not supportive of efficient access control. Access control is also referred to as authorization. Authorization decides which device is allowed to access what resources and under pre-defined restrictions. It also decides whether to revoke certain access to a certain device or even completely deny the access [70]. Authentication is the process of verifying the identity of an entity and its trustworthiness [71]. A device is initially authenticated and then authorized with certain access privileges later. The Foundation of authentication relies upon the “Trust” among devices [72]. In human analogy, “Trust” is developed over due course of time based on certain emotional factors, however, in the case of lifeless machines and hardware sensing devices, Trust is developed based on something they can possess (keys) or know (login credentials) that are verifiable by the networking authorities and the device itself.

A major point of consideration is that a device is authenticated first and if the process of authentication is successful, then the device requests access to network resources by demanding authorization of its request. Figure 3 shows how two IoT network devices/nodes communicate with each other using secure keys and an encryption-decryption mechanism [73]. For mutual authentication using secure common symmetric keys, one of the IoT edge “Device-1” generates random numbers and transmits them to the other IoT edge “Device-2”. The device-2 then encrypts these numbers using the activated authentication key issued by the administrator. The authentication key is available for use only if the device has a valid ID. After obtaining the key, device-2 encrypts the random numbers and securely sends them back to the IoT edge device-1. After receiving the ciphertext, Device-1 decrypts the cipher using its activated key, which

is also obtained only if device-1 has a valid ID assigned. If the message decrypts successfully showing the same random numbers that were sent earlier, it confirms that the two authentication keys used on both the device-1 and device-2 sides were symmetric and the devices are mutually authenticated to each other. After successful authentication, the devices might be wanting to gain access permissions and operate over accessible data. Figure 4 shows the complete process of authorization below:

During authorization, the device requests the owner of the data to grant permission for access. If the grant is received, the IoT edge device sends an authorization request to the authorization server and on approval, receives a token. This token is sent to the resource pool manager that manages the shared resources. The resource manager holds the token and in return share the requested resources by the IoT network device. After releasing the resources successfully, the resource manager recollects these resources and terminates the token, and waits for the fresh token request.

Numerous modern-era web domains utilize a more secured version of hypertext transfer protocol (HTTP) which mostly runs over secure socket layers or TLS- an extensively used protocol that utilizes public key crypto-techniques for secure communication channel establishment [74]. To develop trust among digital devices, digital certificates are also used that have the public key of the server. The CA-certificate issuing authority signs these digital certificates which are to be issued to the IoT edge devices for their authentic identity’s sake [75]. Certain concrete policies and a set of attributed roles that are to be strictly followed by the interacting entities in a communicating network are defined in a PKI-public key infrastructure [76]. It is responsible for managing such trust developing digital certificates. There are login credential ways of authenticating humans that use a terminal interaction node to access the IoT network. Two-factor authentication is based on “what the user knows”- login credentials and “what the user has”- in possession (the hardware support in the form of a smart card or mobile phone device or a digital gadget) [77]. Three-factor authentication is an even more advanced method of authenticating the human users using login credentials, hardware device in possession, and “what the User Is!”—Biometric identity of the user [78]. In peer-to-peer or machine-to-machine communication, crypto-systems are used in integration with the smart IoT devices to boost security and enable stronger versions of keys for concrete authentication, which is lesser prone to attacks. Strong keys generated by crypto-systems are used to authenticate the network devices and those devices that are successfully authenticated are given access privileges based on the degree of permissions allowed for different level devices. One of the most common and major problems in integrating the complex crypto-systems with the IoT is the challenge in managing the cryptographically generated keys [79]. Trustful relationships

Fig. 3 Mutual authentication among two IoT edge devices using secure authentication Keys

Fig. 4 Authorization of an IoT edge device requesting permissions to access shared resources

among IoT devices can be done in various ways like contacting a root authority to validate the existence of a device with a certain ID number and check the proof of the legitimacy of the real or fake node in the network. If the identity is approved by the centralized administrator, then the communication and exchange of information can be done accordingly. In rooted central authority without any decentralization of power, secure socket layer-based public key infrastructure is used with a centralized certificate issuing (CA) in the communication, it just monitors the authenticity of the devices among them and manages the authentication keys of the IoT network.

A different 3rd party authority responsible for IoT device (sensors, gateways, user nodes) authentication is based on Kerberos [80]. Wireless sensor networks are also a major

part of IoT networks that transfer the computational load on the heavily configured base stations via which the commanded controlling of resource-constrained edge devices is possible. In some cases, the base station itself is configured in such a way to manage, distribute and revoke session keys so that lesser load will be on the battery-powered devices and higher performance can be achieved. For example, SNEP-sensor-network-encryption protocol utilizes the master authentication key mechanism where every individual edge device shares their respective private keys with the centralized base station [81]. A drawback of a centralized server is that it suffers from single-point failure. It means the failure of one system leads to the failure of a complete network. A single point of failure can be avoided if the decentralization of the central authority and its power is made among the decen-

tralized nodes [82]. This boosts the working of each device in an autonomous fashion building a greater level of trust rather than simple use of third-party intermediates with a lesser level of trust. In cyber-physical devices, trust is a metric that represents an approval on whether to share information or not with the device being authenticated. Blockchain authentication is quite different from centralized third part authority where the decentralized ledger can be used to perform authentication among peers using crypto-link verification [83]. Blockchain and IoT integration with secure authentication key applications is discussed in Sect. 5 of this article. Bitcoin is a digital cryptocurrency based on blockchain which was first created to replace cashless currency for use in day-to-day expenditures. Blockchain-based authentication and key management services are almost impossible to compromise by a single malicious block participant to manipulate transactions [84]. In general, security key management and distribution based on decentralized mechanisms are more fault-tolerant, resistant to attacks, and adaptive to changes than centralized mechanisms. The computational load for authentication and authorization is distributed over IoT devices, gateways, and user nodes, which leads to overall better flexibility and scalability of the network [85]. IoT devices need to be authenticated quickly taking into consideration the rapidly draining battery of the devices. The streams of data being generated in IoT network is rising rapidly and need for real-time computations and processing for cyber-physical systems is growing exponentially [86]. Fog computing was introduced by “Cisco Systems”. Fog computing helps the internet of things to extend the services of cloud computing and miniaturize it to the levels, up to which resource-constrained IoT edge devices can bear the computational load [87]. Network nodes extending the cloud services to the edge devices and providing a layer of abstraction to the users to hide complex technical aspects. These network nodes also need to be authenticated for the legitimacy of devices and building “Trust” among the peers. Edge computing is efficient in terms of filtered data collection at the higher levels of nodes such as gateways, where the pre-processed data from the edge devices is received [88]. An edge device authenticates the node to which the processed raw data is being sent. Data is itself accessible to different nodes to a restricted level based on authorization permissions assigned [89]. Session keys and intermediate authentication keys are responsible for generating a “trust” factor among peers interacting with one another. Everything discussed so far on authentication and authorization can be done only when two or more devices mutually agree to trust each other before exchanging information. KMS-Key management systems manage the various secure keys that are computed using a random key generator based on various parameters discussed in Sect. 4.

Internet of things is a vast research field with lots of scope for the integration of efficient but more secure heterogeneous

technologies. Technologies to be picked before implementing and deploying the IoT network for data collection depend on the demands of the problem statement and the requisites for data type/formats. This heterogeneous infrastructure of the IoT network helps in deciding which KMS-key management system is to be used for generating, managing, issuing, revoking, and discarding a particular key for an edge device. Two broadly implemented versions of IoT infrastructures are:

- *Local and Centralized IoT edge enabled Infrastructure* Vulnerable to a single point of failure and lesser fault-tolerant.
- *Global and Decentralized IoT edge enabled Infrastructure* Better load balancing and fault tolerance with no single point failure due to decentralization.

<Local, Centralized> [90, 91] The spatial/geographical positioning of sensor nodes has its own merits over the single point non-distributed cloud computing model, in terms of network security and data privacy. This type of infrastructure stores the authentication key and session key generation parameters locally on the devices. In this, there is no back and forth of security generating parameters from edge level devices to the higher cloud extending fog nodes. Only special authentication keys for gateway-to-gateway communication are generated for cases where information is to be sent from one cluster to another. Here, every cluster head represents its group of nodes and substantially provides guarantees on behalf of IoT edge devices, that these devices are also authentic if the cluster has been properly authenticated because every IoT edge device in that particular cluster is already authenticated by the cluster head during their initial registration phases. Local and centralized IoT infrastructure suffers from a single point of failure and thus the authorization procedure is exhibited by dividing the session keys to their respective devices while registering them. The login and credential information along with the authorization access policies are stored as locally available registered entities in the storage media. These keys are valid for specific purposes and time duration only. Caching of such cryptographic session keys makes it efficient for reconnecting to the same session rather than rebooting the complete session.

<Global, Decentralized> [92, 93] This type of IoT infrastructure stores data globally in a decentralized manner. The decentralized trust which is spread all over the network provides better levels of security by making the IoT network more resistant to attacks and more fault-tolerant. Decentralized authentication by global availability of session key pools and their formal verification by administrator enables the limiting impact of IoT attacks as a limiting amount of information is revealed if a node somehow gets compromised. Using internal firewalls boosts the network security

strength and protects it from flooding attacks, sinkhole, and blackhole attacks. It is feasible to do such integration of firewalls because the device location-based local authorization is not dependent on any third party, which enables proper machine-2-machine or peer-2-peer authentication. When the authentication keys providing system is not available due to some reason(s), say, due to any network threat or a DOS attack, the overall impact of using decentralized infrastructure is to limit the local authentication and authorization services. For even more security enhancements and session key generator node protection, a resilient backup of “Trust” and information about key generation, revocation, discarding of obsolete keys is stored for the pre-registered components of an IoT network.

Authentication prevails authorization and based on successful key encryption/decryption process, the devices share sensitive information. However, there are a lot of flaws in key management systems. Programmers develop lightweight protocols by shrinking down the complexity of the parameters based on which the respective session key is generated for as many sessions that are required to be established among network components. Section 4 ahead illustrates how minor flaws and vulnerabilities can cause the whole IoT network to be compromised and also illustrates the need for proper mutual authentication. IoT security enhancements are required in both aspects, Physical securities to avoid insider attacks and Cybersecurity for internal IoT architecture on which all components operate.

IoT edge level session keys: vulnerabilities, attacks and preventive solutions

Session keys are an important part of authenticating an edge device. Devices registering in an IoT network request for their respective SIDs (sensor IDs). SAIH-System administrator and initiation handler then listen to the single or multiple concurrent requests for registration as numerous devices might be interested to join the network at a particular point of time [35, 73]. SAIH issues the temporal ID, edge devices private key XN, and a random nonce RSR, which are further used to communicate with the gateway node (HGWNN) using their respective private keys. Each device establishes a secure communication session using their private keys to interact with the gateway node and get their IDs verified and approved by the cluster gateway. Finally, the gateway node sends the acknowledgments to the respective IoT nodes (edge devices) and also a command to delete their respective random nonces RSR. This way these devices are supported during their registration phase in an IoT network. It is to be noted that A1, A2, A3, up to AN in vector <SIDN, AN> are generated using XOR ($\oplus$) parameter. The random nonce for sensor registration, RSR, is XORed with private keys XN to generate AN

respectively. XOR parameter is difficult to reverse but has a flaw in its functioning. There are certain vulnerabilities in this mechanism. Suppose, $XN \oplus RSR = AN$, then again XORing, $AN \oplus RSR = XN$ (It yields the private key which is exploited). XOR parameter is used because it is lightweight and easy to implement on resource-constrained devices but hard to reverse. If the random nonce RSR is somehow available to the attacker (sensor capture attack), then every private key is reversible as shown below in Fig. 5.

Preventing all sessions authentication key hijacking by the attacker

IoT edge devices are physically exploitable and can be tampered with using hardware mechanical tools to breach the internal security parameters stored in the device’s memory or cache. If the edge device is captured and the attacker has physical tools to breach the hardware security, all the contents of the memory are exploitable and information vector <SID, XN, AN> might be revealed. In case of an asymmetric session key being used, if any of the private keys of the user node, gateway, or sensor node is compromised, then the whole session key for every device in that session is compromised [94]. The problem arises when the capturing of a single node enables the attacker to control the other network nodes capturing which opens the scope for various other attacks such as BOTNET attacks [95], node impersonation, and identity theft attacks. One session key vulnerability here holds the complete network vulnerable to the verge of collapsing. In this case, such an attack can be avoided if separate salts/nonces (random numbers) are used to generate separate individual keys for each sensor node. Here the HGWN bears the computational load of managing the increase in the number of keys per device connectivity. This prevents all the sessions from getting hi-jacked by the attacker, however, that one particular session which was attacked and accessed by the attacker by capturing the sensor node remains infected and requires a reboot [96]. Rebooting the device should be done in such a way that the forward and backward secrecy is maintained while re-registering the sensor node once again into the network with a fresh new key allotted for mutual authentication. Restarting the session or completely discarding the infected sensor node from the network can be done as partial solutions to the problem of session key compromising attacks.

Key vulnerability attacks in 3-factor smart card-based authentication [97–99]

Consider the following timeline Fig. 6, where suitable parameters for Three-factor smart card-based authentication exchange of information takes place among the user and the system administrator.

Fig. 5 Sensor node capture attack and private key extraction during IoT Edge Nodes registration

Fig. 6 A three-factor smart card -based authentication exchange of information taking place among the user and the system administrator of an IoT Network

In the above timeline exchange of information, a new user tries to create a fresh UID (username for itself). Only if the user id is fresh, a private key is issued to the user otherwise the request is rejected by the administrator system. Alpha and Beta parameters are security complex values generated to confuse the attacker as it is a multi-hashed value of user-id, the private key of gateway node, and the private key issued to the user only if the UID generating request was fresh. User ID issued to the newer user is stored along with the login credentials created into the database system. Miscellaneous values are stored in the smart card of the user. These credentials are

sent digitally through a secure communication channel and signed digitally as well [100]. A user on the terminal then performs certain formalities for the final authentication of the smart card credentials. Finally, the vulnerable intermediate key values and complex < Alpha, Beta > values are deleted permanently so that there are no traces left for the attacker to re-generate the security parameters. For more details on the parameters used in the figure in detail refer to [101, 102]. This complete process of smart card registration and authentication along with username and password creation has certain flaws as well. A smart card is a hardware entity vulnerable

to threats of stealth and misplacing. Attackers can steal the smart card to exploit the information stored on it to impersonate the user and access data on the user's behalf. This way user impersonation, gateway impersonation, and sensor node impersonation attacks can be executed with ease.

Avoiding gateway impersonation attack due to stolen smart card and exploited keys [19, 103]

Authentication keys of any device connecting to the IoT Network are limited to a size supported by the computational power of their respective hardware. Message passing is done in the form of multi-hashed values with various small parameters that were used to generate the final complex hash value. For instance, taking into consideration of Fig. 3.2. A simple message from the user node to the HGWN node is,

$$M = h(\text{UserID} || \text{GatewayID} || \text{Private Key User Node}(X_N) || \text{HGWN Key} || \text{TS}).$$

Here, Timestamp (TS) ensures forward and backward secrecy of the data. In the case of symmetric-key cryptosystems, there are very few dissimilar parameters. It means like RSR is a common parameter, but having heterogeneous parameters involved in the computation of messages based on hashes, to share necessary parameters required for generating the session keys from one node to another makes it difficult for the attacker to simultaneously compute all the hidden small parameters before the session expires. A session key is active only when there is an active session and the attacker is supposed to compute all such small varying parameters within the time window before the session terminates. It is not possible if the session key is composed of different parameters such as UserID including the private keys of gateway node, sensor node, and user nodes. The basic form of SECSK- secret session key is as shown below:

$$\text{Secret Session Key}_{(\text{Sensor}, \text{HGWN}, \text{User})} = h(\text{SID}_{\text{Sensor}} || \text{UID}_{\text{User}} || X_{\text{HGWN}} || X_N || X_{\text{USER}}).$$

It is not feasible for the attacker to exploit all these private parameters located at different device's memory at different distances because the secret session key is protected by the system administrator-initiator and handler very carefully in this solution.

Preventing IoT edge node impersonation attack due to stolen smart cards and exploited keys [104]

Attacker continuously attempts to overhear the network traffic and capture any vulnerable non-encrypted data packet during transmission. To impersonate as a sensor node, S_j after compromising the messages transmitted during the message

and parameter passing phase, the attacker has to compute Message "M" to obtain private keys and impersonate as a sensor node. It is very similar to the analysis of the gateway impersonation attack analysed earlier. So, Adversary is incapable of obtaining private keys of different devices due to changes in random nonces at the beginning of IoT edge device registration phases. An Attacker cannot perform the Sensor Node impersonation attack unless until it has obtained every device's private key from which the bigger secret session key was generated.

There exist numerous authentication and key generation, distribution, management frameworks from some of which are discussed ahead. Table 1 gives a comparison among some significant frameworks.

Security perspective and proofs with limitations

In 2011, Islam and Biswas [105] proposed an elliptic curve secure ID-based mutual authentication mechanism that used signature-based verifiable parameters. It was found that despite being lightweight and support towards session key generation, this authentication scheme suffered from MITM-man in the middle attack and brute force key guessing. Zhang [106], utilized a biometric dynamic scheme that also used an integrated ECG cryptosystem for peer-to-peer mutual authentication. This framework offered lightweight authentication with a smaller key size which was widely applicable in body area sensors. Due to insecure ways of transmitting crypto-parameters, this mechanism is vulnerable to session-key-hi-jacking and prone to node impersonation. Li [107] proposed a trust metric-based authentication framework that supports forward transmission security and provides an abstraction to the user terminal nodes. However, this approach suffers from key-reconstruction and passive attacks. Shi [108] introduced a dynamic lightweight peer-to-peer and gateway-to-gateway mutual authentication model. Shi's model is resilient towards brute force attacks but it is found to be vulnerable to MITM, session hi-jacking, session re-establishment, connection re-direction attacks. In 2015, Qin [109] also proposed their version of authentication protocol which uses a signature-based certificates scheme for IoT edge device authentication. This approach is resilient towards key escrows and provides both forward and backward secrecy. Qin's scheme is vulnerable to Type-1 attacks and does not provide proper mutual authentication. The working mechanism is certificated and providing simultaneous data integrity verification is a challenging task yet to be achieved. A non-cloneable function-based lightweight authentication protocol with resistance towards key-replication and metadata cloning was derived in 2017 by Xie [110]. The identities of IoT edge devices are successfully maintained authentication key replication is an issue. Identity theft attacks are also very much effective on this scheme. These traditional authen-

Table 1 Various significant traditional authentication mechanisms: features, attack analysis and limitations

Author and year	Authentication mechanism	Superiority of scheme	Features fulfilled	Attack analysis (excluding exceptions)	Structural enhancements and shortcomings
Biswas (2011)	Elliptic curve based secure ID based mutual authentication	A signature based verifiable ID scheme	Lightweight, stealth support for user, secure session key construction	Man in the Middle, lower brute force attacks and key reversal	Resilient and applicable for client–server based architecture
Zhang (2012)	ECG-Crypto-authentication based mutual authentication	Biometric and dynamic security scheme	Lightweight mutual authentication, Low key size, supports tiny devices and body area sensors	User anonymity, session key exploitation, session key hi-jacking, recoverability, static IoT attacks	Supports biometric authentication and can be extended upto all three factors of authentication
Li (2013)	Trust metric based key authentication and mgt	Communication medium based scheme	Re-booting of keys, Forward transmission security and abstraction	Packet in-transit breach passive attacks, key-guessing, key-reconstruction	Flexible with heavier cryptosystems
Shi (2015)	Motion/Movement aided communication channel based secret key generation and authentication	Communication medium oriented scheme	Lightweight session keys for peer to peer mutual authentication of nodes and gateways	Session key-Hijacking, passive snooping of communication tunnel, Man-in-the-Middle, Session re-establishment and re-direction	It is Resilient towards key escrow exploits and is scalable upto 674 nodes
Qin (2015)	Abstractive remote authentication which is revocable, secure and scalable	Signature based zero certi scheme	Resilient towards key escrow and exploits, Provides forward privacy and data transmission security using secret key sessions	It is vulnerable to Type-1 attacks and contradicts the claim for successful mutual authentication	Mechanism works without any digital certificates and data integrity verification is a challenge
Xie (2017)	Non-Clonable functions based lightweight mutual authentication	Physically non-replicable IoT Scheme	Resistant to identity theft, Lightweight authentication with revocation support	Key replication, Meta-data cloning of system functions, identity theft and Key re-construction	Non-linkability, identity secrecy is maintained but confidentiality is not ensured properly
Wang (2018)	Physically non-replicable function based crypto-free authentication	Physically non-replicable function based scheme	Revocability of keys, lightweight mutual authentication	Resilient towards node impersonation attacks but vulnerable towards other insider physical attacks that involve sensor tampering	Cluster head guides the sharing of secret keys among peer nodes in a network
Amin and Biswas (2015)	Three factor based authentication	login credentials, smart card and biometric based scheme	3-way strong authentication with lightweight transmittable key generating parameters	Prone to Node impersonation, sensor capture attack, sensor and Gateway forging attacks	Multiple key based attacks are possible despite the usage of three factor authentication

Table 1 continued

Author and year	Authentication mechanism	Superiority of scheme	Features fulfilled	Attack analysis (excluding exceptions)	Structural enhancements and shortcomings
Kasyoka (2020)	Non-pairing and No digital certificate based authentication	Signature schemes based on architecture that does not exploits digital certificates	Mutual authentication, resistant to key Escrow and supports forward secrecy	Brute-force, key reconstruction, re-keying attacks	Pairing free authentication
Tan (2020)	Cloud computing aided PUF based authentication for multiple node hops	Physically non-replicable function based scheme	Lightweight, supports anonymous identity allowing non-linking among disjoint nodes	It suffers from escrow attacks and gateway node key forgery attacks, node impersonation attacks	A typical authentication mechanism that reflects the shortcoming of PUF in network security
Leandro, Westphall (2021)	Multi factor mutual authentication	Adjustable variable response time, challenge-response function, and nonce based scheme	Utilizes adjustable tolerance time rate (ttr) to detect Man-in-the-middle attacks	Avoids both man-in-the-middle and impersonation attacks	This schemes implementation adds elliptic curve cryptography as a cipher option

tication schemes are benefactor in certain circumstances and lossy during the others in terms of both efficiency and IoT security. In 2018, Wang [111], proposed a physically non-replicable function-based non-cryptographic authentication scheme that supports revocability of the session keys during the real-time application of IoT network. This method is resilient towards node impersonation attacks but it is vulnerable to insider privilege escalation attacks and due to a non-decentralized system of authentication keys, these keys are stored on a single server making it vulnerable to single-point failures and lesser efficient in terms of processing and data transfer. Amin and Biswas [112] proposed a three-factor-based authentication approach that makes the benefit in security using all of the three, login credentials, smart card authentication, and biometrics of the user. The 3-way strong factors for authentication provide high-level security but the session key exchange mechanism suffers from few vulnerabilities that enable the attacker to breach security using sensor node capture attack, discussed earlier. If somehow an attacker breaches into the network and captures a sensor node, it can lead to multiple different attacks such as node impersonation, gateway forging, etc. In Amin and Biswas's scheme, multiple keys-based IoT edge attacks are possible although it uses three factors including the biometric identity of the user itself. In the year 2020, Kasyoka [113] proposed a non-pairing and certificates approach that utilizes digital signatures to provide mutual authentication and resistance to escrow attacks. It also supports forward and backward secrecy. However, this approach is vulnerable to Type-1, brute force, and rekeying attacks. Kasyoka's approach is pairing free which allows multiple devices to interact with each other and does not limits the number of devices exchanging information at a time. Tan [114] gave a PUF based multi-hop authentication scheme that exploits a non-replicable function-based approach. Its features included lightweight support for IoT edge device anonymity and allow non-linking among the disjointed nodes of an IoT network. It is a primitive mechanism that brings the limitations of PUF in IoT network security, which is irreplaceable [171]. Tan's scheme is vulnerable to gateway node forgery attacks and escrow attacks. In the year 2021, Leandro and Westphall [115] introduced a multi-factor peer authentication approach that utilizes random TTR's and nonces/salts for secure parameter passing in wireless sensor networks. TTR stands for tolerance time rate in an IoT network. Leandro and Westphall's approach are effective in avoiding man-in-the-middle attacks and identity impersonation attacks. In this, ECC-elliptic curve cryptography is a cipher option if required for higher end-to-end security. Section 5 describes the inter-relation between hybrid technology-oriented authentication mechanisms based on both IoT and Blockchain for high-end security and data security.

Internet of things and blockchain: basics, IoT keys and authentication

In this era of transformative technology, the Internet of things is expanding way beyond a simple sensor gateway mechanism sharing a stream of information on a restricted geographical space. Artificial intelligence and blockchain are such a trending transformative technology that is being utilized on both heavier computational networks and lighter versions for lightweight networks for high-end security. In this section, we discuss how the blockchain technology works, integration of IoT with the blockchain for enhancement of security, the role of session keys, attack vulnerabilities, scenarios during mutual authentication among IoT nodes communicating with other nodes (P2P), devices interacting with blockchain and devices indirectly exchanging information with a proxy server as an intermediate third party. Integration of AI with IoT is discussed in Sect. 6.

Basic overview of blockchain

Blockchain is a decentralized secure storage paradigm that stores the data after proper verification and authentication, it was founded in 2008 by Satoshi Nakamoto [116]. Data constitutes smart contracts, keys, files, transaction details, etc. It is a decentralized consensus mechanism based on secure algorithms which are used to access and store the data. Peer-to-peer data can be sent with proper mutual authentication or agreement based on keys. Blockchain uses smart contracts for higher-level layer application logic.

Blockchain architecture and security

Blockchain architecture is a very complex bundle of technologies, however, common components of a blockchain include nodes, transactions, blocks connected via cryptolinks, chain, miners, and most important of all, the consensus algorithm on which the blockchain operates. Any fresh record request or transaction request requires a new block to be constructed. The core components are:

- **Network node user** which has a copy of its complete blockchain snapshot.
- **Block** storage capable data structure that stores a subset of the transactions from a superset of transactions decentralized all over the nodes in the blockchain network.
- **Chains** It is a connective chain that can be expanded as series of blocks connected with crypto-links in a particular order.
- **Miners** These are significant functioning nodes that have power assigned to them to verify the blocks and take decisions whether to add certain blocks to the chain or not.

- **Consensus algorithm** it includes rules and functioning directions that contribute towards how the blockchain execution is carried out.

After fresh block construction, each individual is verified and the new block is signed using the digital signatures to give legitimacy to the data record. Every block contains some data, a hash of the block itself generated using a hashing algorithm, and also the hash from the preceding block. The data is kept in each of the blocks and the type of data depends on the purpose for which the blockchain was deployed. For example, a blockchain of ethereum provides the facility of smart contracts to the users, whereas blockchain of Bitcoin supports digitized crypto-currency that is aimed to be used as a digital currency rather than paper cash, so ethereum blockchain stores smart contracts whereas bitcoin blockchain support transactional storage of records in a meaningful sequence. Each block is associated with its hash generated using a hash algorithm such as SHA-256 that is based on KECCAK [117] based sponge functions. Hashes enable the blockchain to identify any changes exhibited in the network. Each node in the network is verifiable by the consensus algorithm and all the nodes within the blockchain network follow the consensus rules [118, 119]. Blockchain is highly unhackable as for an attacker to completely obtain control over the transactions, an attacker is required to authenticate itself to every dynamic block in the chain and obtain legitimate hashes of each block. Even if the block hashes are obtained, authenticating to each block is impractical as the keys for interaction with each block varies, and obtaining all the keys is an impossible task without collapsing the complete blockchain transaction system. In the Internet of things, such blockchain mechanism is being used extensively nowadays as IoT edge devices are not so resource-constrained due to advancements in technology. Blockchain is also used with IoT to provide higher securities instead of implementing heavier cryptosystem-based verification and authentication systems [119]. Following Fig. 7, illustrates the transaction verification and approval procedure with the addition of newer blocks in the pre-existing blockchain. In a simple blockchain transaction, a requesting user initiates the transaction which is forwarded to the p2p network of nodes. Inter-blocks communication and information interchange happen with hash verification for data integrity. Some peers may interact with others as well during transaction verification. Before this interaction can take place, peers authenticate themselves to each other using their respective private–public key pair or a single symmetric key. After successful authentication, peers develop trust based on which the critical transactional parameters and information are shared securely over a communication channel. Once, the transaction is verified, it is to be stored in the form of a block and for such, a new block is created for the ledger. Later, the block is merged with the co-existing clocks within the

Fig. 7 Blockchain transaction processing and addition of newer blocks

blockchain after proper verification and hash check mechanism is completed successfully.

The inclusion of the new block is such that it is unalterable and permanent. Crypto-link makes the bonding of the new block with the blockchain secure and data integral. Finally, the acknowledgment is broadcasted to the whole network of nodes and the individual copies of blockchains at their respective nodes are updated to maintain data integrity. This is how a typical blockchain performs a simple transaction. The architecture of blockchain has a lot of features that are beneficial from an industrial transaction management perspective, with several inbuilt characteristics as:

- *Cryptosystem support* transaction and identity verification are trustworthy due to the strong crypto-systems integration compatibility of blockchain. Session Keys for secure communication can be generated using private keys of individual network components [120, 121].
- *Immutable Data records* any record updates done in blocks cannot be modified or deleted.
- *Provenance* every transaction is traceable up to its origin from where it was initiated.
- *Decentralization of Ledger* every blockchain node has access to the complete decentralized database records.
- *Anonymous identity* every blockchain member produces an address for communication rather than the identity, keeping the original identity of a user/node anonymous.
- *Transparency* blockchain network is highly resilient and resistant towards unwanted attack-based system modification. Transactions are transparent and it requires very

high computational powered systems to override the whole blockchain network itself.

Utilizing blockchain into IoT networks started when the edge devices started to grow in computational power. Older boards like Arduino and smaller chipsets based on 8086 were highly resource-constrained and in-operable in terms of heavier security solution implementation [122]. With advancements in technology, heavier security cryptosystems and session key generation became feasible without putting a considerable amount of load on the IoT edge devices [123].

Blockchain for IoT edge services: role of keys

In the aim to construct a blockchain utilizing an IoT network, heavy edge devices and gateways are deployed that can access data from the network as well as securely store information on the decentralized blockchain. A single point of the vulnerable device can be used to compromise the whole IoT network leading to disastrous circumstances [124]. An IoT network can have numerous edge devices connected to the gateway(s) up to 256 devices even in a small industrial analogous application. The large number of devices connecting to a network requires that many number session keys be generated at a particular point in time. Re-keying can be done only when one device frees a key and that key is available for assignment to another device. If the number of devices wanting to connect to an IoT network exceeds the number of keys that a particular gateway node can store, it limits the connectivity of such devices and also limits the scalability of

the network. To solve this problem, Blockchain is exploited where secure pooling of pre-generated and verified session keys can be stored, accessed by the gateway when needed, allotted to a device in need, and finally release the key when the session terminates [125]. For example, the connection of numerous heterogeneous devices can bring a lot of vulnerabilities to the network. A Raspberry Pi or Nvidia Jetson Nano [126] board using an edge device might not be as resource-constrained as an Arduino board using a sensor device. This means higher security algorithms can be deployed over Raspberry Pi and Nvidia devices rather than an Arduino device. This example illustrates that some edge devices may be more secure in an IoT network, however, the lesser secure devices can be exploited by the attacker to obtain session keys. Based on successive analyses of keys and their pattern, the attacker can execute session re-establishment attacks [127], exploiting forward and backward secrecy of the data [128]. Storing the keys securely in a blockchain is possible but comes with higher costs of implementation and making the communicational channels stronger in terms of evading eavesdropping and passive attacks is a challenge. It is a challenge because predicting what hardware specification device will try to connect to the network in the future and what vulnerabilities it brings to the network is tough. Smaller security features like CAPTCHA's and OTPs [129] can boost the security of the network to certain limits. Even if authentication is achieved successfully, access grants are required to acquire shared resources and data services.

Access control and key authentication management

Access control is responsible for the final decisions of whether an access request is to be approved or not in a network. It is to provide a runtime environment with a set of procedures to detect, categorize and inhibit all functions in the system and perform such complex calculations with abstraction to the users by providing a plain simple user interface. Basic keys for secure session establishment are used to authenticate peers meanwhile complicated crypto-keys are used to authenticate the inter-blocks exchange of information in a decentralized ledger system. Authentication is done to detect whether the request coming from a peer is legitimate or not. Access requests from the attackers are filtered to some extent using key-based authentication systems [130]. Authentication can be done using one, two, three factors of authentication depending upon the need of security level. In traditional ways of IoT deployment, which lack artificial intelligence or blockchain, identity is authenticated with a login credential pair of userID and passwords for each particular device in the network. Protocols that utilize single signing-on can accelerate the authentication procedure due to their simple design, it also provides a third-party system that compromises the confidentiality and privacy of data

being exchanged in a simple design authentication mechanism. To solve this problem of third-party intervention, a newer design was introduced in the article [131]. This particular framework supported the user to only authenticate itself to the blockchain (such as Bitcoin [132] or Ethereum [133]). Once the user is authentic to the blockchain, it can use the access to obtain the session keys to securely connect to its peer if needed. Smart contract broadcasts the token and the blockchain address while authenticating the identity, and the edge enabled IoT network services will receive the combination vector containing the user's public key, internet protocol address, and the key which is used to authenticate the block and device mutually. Biometric information can be collected, stored, and verified by blockchain to solve the impersonated attackers' problem in the access authentication scenarios [134].

Confidentiality and reliability of data

IoT has wide applicability in various fields like smart gadgets, healthcare monitoring devices, agriculture, geographical landscape surveying, military drones, and war fields analyzing smart devices, finances, etc. [168]. Health care and medical facilities are getting sensor-based upgrades that boost the data collecting and patient monitoring capabilities of hospital management. It helps in improving the methods for quicker responses towards patients in need of urgent care-taking. The private health considerate data recorded by these sensors is to be stored safely while maintaining integrity, confidentiality, and privacy of the person associated with the data. Rui [135] introduced a mechanism that uses decentralized storage and enhances the byzantine scheme for fault tolerance with the consensus protocol to securely store the information. Xue [136] on the other hand provided a design that deals with the dynamic crowdsourcing architecture. This architecture is established keeping blockchain as its foundation and ensures data privacy.

Blockchain for data sharing in IoT network [137]

Data is the basis for higher accuracies in predictions when analyzed by artificial intelligence algorithms. Larger the amount of data, better the learning of the algorithms to yield promising experimental results. This data is collected by various heterogeneous devices in different storage formats. Data could be temperature or humidity reading in agricultural fields or a patient's respiratory rate and oxygen consumption level in healthcare areas. All such heterogeneous forms of data cannot be stored in single chaos of information. Blockchains blocks are variant in size and connected with a crypto-link. Processed relatable information can be stored in one block and linked with another block that might contain a different type of information [169,170].

Such a storage mechanism is more categorized and easier to manage. Blockchain utilization is only meaningful when the data size is larger. Implementing blockchain for smaller data generating IoT networks is meaningless as it will be highly inefficient in terms of computation utilization and energy constraints. Isolation of mixed heterogeneous data in non-blockchain authentication mechanisms will be higher and time-consuming due to repetitive data. Therefore, sharing data from the IoT services in the database can assign the resource properly and reduce the avoidable cost. Due to numerous data formats and heterogeneous devices, there is a lack of trust among the peers of a network. This lack of trust can be overcome only by a proper authentication mechanism. Introducing a way to securely store keys or data (blockchain) and establish secure communication among peers is required for an IoT network to work in safe mode. Blockchain provides only a better option of sharing the data securely. Accessing that stored data wirelessly requires the use of strong authentic keys for establishing a secure communication session among peers (P2P network). There exist two scenarios of blockchain and IoT edge devices interaction as discussed below:

- IoT edge things inside the Blockchain network cluster
- IoT edge things outside the Blockchain network cluster

IoT edge things inside the blockchain network cluster

In this organization of the network, IoT edge devices behave as normal nodes in the cluster of the blockchain network. IoT devices are treated as a lightweight node which does not require storing any duplicate copy of the complete distributed ledger. These lightweight nodes also do not take part in the mining process. Maximum distributed implementation is observed in this kind of structure. Key management systems play their part of duties for authenticating each node with the other during inter-node communication. The most important “key” for a device is the key that is used by the device to authenticate itself to the blockchain, this key verification decides whether the device is allotted a secure ID and allowed to be a part of the IoT-Blockchain cluster, or not. This key can also be decentralized if higher-end security is required. In traditional IoT Networks, a session key was generated taking the private keys of all the interacting components. For example, in most of the IoT networks, a sensor-node/IoT-things, gateway, and the user interact with each other only after the session key is generated. Numerous session key generation mechanisms utilize the private keys of the sensor, gateway, and user as Sensorkey, HGWNkey, or FGWNkey and Userkey respectively.

Here HGWN denotes the home gateway node for an in-cluster communicating edge device and FGWN represents the foreign gateway node for an edge device of one cluster,

communicating with another gateway of a different cluster. Figure 8 shows the inside edge device-to-blockchain interaction components:

Overall performance of the network is not so much affected even if few IoT nodes are dysfunctional or corrupted. The pre-configured high-performance blocks are the strong points of the blockchain network. Work that requires heavy computation is assigned as a block instead of a single node. Therefore, the dysfunction of the IoT edge devices will not be affecting the overall network performance.

IoT edge things outside the blockchain network cluster

In this organization of cluster nodes, there is a need for a proxy server system that acts as the intermediate between the communication among the IoT nodes and the blockchain. The blocks of blockchain are interconnected with cryptolinks that are different from those outside the blockchain components. For communication establishment, the sensor nodes are required to authenticate themselves to the proxy server which is a centralized component and interacts with the blockchain. Figure 9 shows below the outside cluster IoT node to blockchain interaction.

A request from the edge layer of the IoT network to the blockchain can only be transferred if the proxy server is itself authenticated to the blockchain. This agreement can only be achieved if there exist two different session keys that are difficult to replicate or steal. One session key is also sufficient but bears a greater risk of being compromised. The advantage of this cluster organization comes from its flexible configuration. As the IoT edge devices are indirectly linked to the blockchain network cluster, this centralized proxy server can act as a network request re-director. It is simpler to control requests from the IoT nodes and supplies efficient performance by reducing the data transfer latency. In comparison with the structure of IoT edge devices inside a blockchain network, this structure allows the IoT network system to exploit the procedures of blockchain. When the nodes are distributed, it is important to secure the communication channel with the security keys. To manage the security keys a block-chain based key management scheme [138] can be employed which facilitates resource authentication. The scheme is based on designated prover proof-of-work (DPPOW) that is used to ensure resource authentication in the fog environment. The secure key management can be observed through hash-chains that use advanced blockchain technology allowing the generated key pairs to verify themselves for the IoT devices. The system uses Ethereum to develop a blockchain network for speedy validation of the keys. The block-based key management experience its application in a named data network where the authenticity is obtained by storing the public key hashes into permissioned blockchain for each network

Table 2 Blockchain based key management systems for IoT network components and their authentication, KMS features and attacks identified

Author	Authentication	Confidentiality	Availability	Integrity	Feature	Attack identified	Application environment
Chen et al. 2021 [141]	✓	×	×	×	Block-chain based security and Data recoverability	Plain text attacks, non-repudiation attacks	Fog IoT
Panda et al. 2021 [142]	✓	✓	×	✓	quick validation and increased scalability	Spoofing, data tampering, DOS attack, impersonation attack,	Distributed IoT
Lou et al. 2018 [143]	✓	×	✓	✓	Reduces verification rounds and higher verification efficiency	Fake content identification	Named data networks
Lei et al. 2017 [144]	✓	✓	×	✓	Decentralised blockchain network simplifying heterogeneous key management for dynamic collection	Black box	VANET
Ma et al. 2020 [140]	✓	✓	×	×	Efficient and automatic key management through smart contracts	DoS, Public key Tampering,	IoT
Ma et al. 2019 [145]	✓	✓	×	×	Excellent scalability, privacy principles for hierarchical access control	Fork after withholding (FAW) attack, selfish mining	IoT
Tian et al. 2020 [146]	✓	×	×	×	Secure cluster formation and node movement replacing third party with stake block chain for key management	Pseudo Base station attack, interception attack	DWSN
Wang et al. 2019 [147]	✓	×	×	×	Reduced communication cost for key updates and revocation. Low computation for conditional anonymity	stolen verifier attacks, replay attack, impersonation attack	Smart edge grid computing network
Jia et al. 2021 [148]	✓	✓	✓	×	Reduced time for data storage. Minimised space of each sensor	Offline dictionary attacks	Intelligent building energy management system
Chen et al. 2021 [149]	✓	✓	×	✓	Easy management of devices through device manager	Malware	IoT

Fig. 8 IoT edge devices/things inside the Blockchain Network cluster

Fig. 9 IoT edge things outside the Blockchain Network cluster

node. The verification job is done by the gateway which benefits the router with less communication and key management overhead. In a heterogeneous network, like vehicular communication systems (VCSs), the security managers use block-chain-based key management systems to capture vehicle departure information, encapsulating keys in the block for transfer and applying rekeying to vehicles within the same security network [139]. The framework allows for secure dynamic transactions reducing key transfer during vehicles handover. The public key is bonded with its identity using blockchain technology. The decentralized key management mechanism (DB-KMM) [140] allows for periodic revocation and key updates to prevent attacks caused due to insecure keys. All these authentication mechanisms and security key management systems have certain vulnerabilities towards the attacks. Table 2 presents the blockchain-based key management system for IoT network components and their proper authentication delivery comparison.

A smart contract-based mechanism is applied for the detection of adversaries in the network. In a fog computing environment, the privacy of users is violated due to the dependency on third-party for key generation access. Therefore, distributed blockchain architecture based on blockchain technology satisfies the decentralization and high scalability requirements incorporating privacy principles for cross-domain access [145]. The Industrial IoT (IIoT) environment considers the dependency for key management on untrusted

third parties where security and reliability are often compromised. In such a scenario, stake blockchain-based secure key management (BC-EKM) is constructed for a hybrid sensor network. The stake blockchain replaces the functions of untrusted base stations in dynamic wireless sensor networks by acting as a trust machine for key management. The implementation of cluster formation and secure node movement algorithms depict improved trustworthiness in IIoT. The requirement of high latency and real-time services in smart grid systems demands secure communication, which the existing protocols fail to provide. The blockchain-based mutual authentication and key agreement provide conditional anonymity and security assurance in edge computing-based smart grid systems [147]. Even, intelligent business systems face many challenges due to the limited sharing of underlying information when the WSNs are deployed in these systems. To overcome the lacking flexibility in the system, a blockchain-based dynamic key management strategy is applied. The key management cycles were managed through the generation of sub-secret keys into sub-data clocks and distributing the keys for data block acquisition followed by secret key reconstruction. The performance of the work was marked by efficient storage, low energy consumption, and high network connectivity [148]. The blockchain architecture is popular for its distributed nature which is considered better than centralized systems, which face compromises due to single point failure. The protocol proposed in [149] uses a device

manager to connect different IoT devices to blockchain-based networks and maintain the security and privacy of the devices through authenticated group key management. Section 6 ahead focuses on Artificial intelligence-based integrational operation of IoT edge networks and compares smart key management and authentication frameworks giving side-by-side features, vulnerabilities, and limitations of each.

Artificial intelligence and IoT: architecture, attack analysis, and learning security

With the gradual growth of smart network applications and complex network theory advancements along with the development of hybrid sensors controlled by the latest micro-processing boards [150], the Internet of things can generate a huge amount of data. Every sensing or data collecting node itself sends streams of binary data at lightning-fast speeds. Healthcare and agricultural fields are also giving huge attention to IoT sensing networks that provide numerous facilities to these fields. Growing applications in health monitoring, smart gadgets, and connectivity demands of users to the internet boost the IoT global value and recognition. The huge amount of data generated by the larger scaled networks is stored on large storage capacity media, private servers, and even cloud like Amazon's AWS [151]. Figure 10 describes the co-operative working of AI and IoT.

This data stored contains lots of unprocessed data that is raw and could yield ground-breaking results from a prediction and forecasting point of view. However, IoT edge devices, gateway nodes, proxy servers, and even heavy computation-powered servers are not able to process such a huge amount of data. Artificial Intelligence then comes to the rescue where lots of hybrid algorithms based on decision trees and statistical approaches are available to consume that data and provide a learning capability to the real-time operating IoT network. This network performs over feedback approach where the data standard deviations are coupled with the average values of the sensing nodes to correct the deviations and also produce more accurate results. The inclusion of Artificial Intelligence in the IoT network not only enhances the data processing and prediction capacity but also provides support and justifies the cost of database infrastructure. Figure 10 shows how data are collected from edge layer devices and brought back to the storage hub where larger data can be operated upon by high-level technologies like Hadoop, cloud computing, and finally, artificial intelligence. In the architecture shown in Fig. 10, numerous heterogeneous devices are collecting data every millisecond (basic time iteration for a sensing device to capture data and repeat its action). These captured raw data are sent to heavy computation capacity fog nodes for further processing. Data is pre-processed and stored in an organized manner in the data lake. Hadoop can

be incorporated if managing data otherwise becomes hectic. The smart ensemble algorithms then feed upon this huge data lake of pre-processed data to learn about the type of data and sensor reading deviations. The more the algorithms learn and provide feedback, the better the corrections are made in the data, and correspondingly more accurate results are obtained from such complicated assembly of heterogeneous technologies. Feedback learning is for data corrections and refinement purposes. The researchers have proposed numerous models that incorporate AI and ML-based models for data classification and analysis. The basic idea of employing ML and AI techniques is to make an automated system that learns through experiences [152–154]. In Fig. 11, an AI-based model is depicted that is an iterative procedure in different phases of collecting data from different devices and training the data using classifiers (Fig. 11).

Various phases of Fig. 11 are discussed below that provide the working of ensemble algorithms in the process of learning and feedback over training and testing data respectively:

1. *Data Collection* The interconnected devices communicate through the network hence generating data that needs to be collected on large storage devices or cloud-based storage to maintain integrity. The heterogeneous formats of the data are generated amongst which the user data needs to be identified for further processing.
2. *Data Pre-processing* The collected raw data is often inconsistent and incomplete to understand. The pre-processing phase converts the data into a readable format based on certain behavior or trends. Some of the data might be missing related to our interest, therefore encoding is also applied to get fill the missing values and apply to encode the fields that bear a different behavior.
3. *Sampling* When large datasets are considered IoT environment, the study of patterns and trends have to be made on a subset of the dataset to select data points for manipulation and representation of data under examination. There are various techniques used for the optimized sampling of data. In AI the dataset is divided into training and testing set so that the model learns from the training data, and for testing applied on testing data.
4. *AI-based classification* The model is designed based on the dataset, interest of the analysts, and expected outcomes from the data trends. The existence of different AI-based models eases the task of analyzing the data and finding suitable patterns from it. The different algorithms react and are trained according to the classification problem. For instance, if the problem is to predict the target variable to achieve a specified accuracy, then supervised learning algorithms can be used regression, decision tree, random forest, KNN, logistic regression. Similarly, if we do not have any prediction or outcome variable to follow, we use unsupervised learning algorithms (Apriori algo-


```
graph TD; A[IoT Edge Devices collecting Data] --> B[Data Collection]; B --> C[Data Pre-Processing]; C --> D[Sampling]; D --> E[Training Data]; D --> F[Testing Data]; E --> G[Learning AI Classifiers]; G --> H[Produce Final Model]; H --> I[Evaluating Model]; F --> I; I --> J[Feedback for Data Refinement during Pre-Processing]; J --> C;
```

The flowchart illustrates the IoT Edge AI Model Training and Evaluation Process. It begins with IoT Edge Devices collecting Data, which leads to Data Collection. This is followed by Data Pre-Processing, which then leads to Sampling. Sampling involves Splitting data into Training and Testing sets. The Training Data is used for Learning AI Classifiers, which includes Decision Trees, Random Forest, XGBoost, Logistic Regressor, KNN, and Naive Bayes etc. The Learning AI Classifiers produce a Final Model, which is then evaluated using the Testing Data. The Evaluation Model provides Feedback for Data Refinement during Pre-Processing, which is fed back into the Data Pre-Processing stage.

value. The predictive value is saved for evaluation of the model based on testing data samples.

6. *Evaluating model* The model is evaluated based on testing data and then making a match of the output value to the predicted value. The performance of the model is evaluated based on different parameters as accuracy, precision, F1-score, and recall. The parameters are calculated based on the false positive rate, false negative, true positive, and true negative rates. Every parameter has its

significance, for instance, the higher the accuracy of the model, the better is the model.

Lots of researchers have done significant work in the field of smart IoT networks that utilize AI and feedback learning for performance optimization, attack prediction, and deploying smart key management systems. Table 3 highlights the authentication and key management frameworks that exploit artificial intelligence algorithms to obtain smart edge device connectivity and better response learning towards network attacks.

Punithavati [156] recently proposed an IoT security framework that is based upon cancelable biometric systems (CBS). It is based on a support vector mechanism and applicability is cloud computing dependent. This approach is vulnerable to insider and malware attacks. In 2018, Beza-wada [157] introduced a behavioral finger-printing protection and authentication scheme that was successful in providing availability of the data but is found weak against IP spoofing and IP masquerading attacks. This framework fails to provide confidentiality and data integrity despite using algorithms based on KNN, decision trees like Adaboost, XGBoost, and Random Forest. Chakraborty [158], gave a challenge-response protocol that works on algorithms like LogitBoost, Random forest, SMO-SVM, bagging, and Kstar. This framework is vulnerable to machine learning-powered rigorous attacks that target node's response based on congestion like BOTNET attacks, Malware attacks, etc. Restuccia [159] in the year 2018 proposed a polymorphic security-enhanced mechanism for authentication and data sharing based on secret keys. It uses SDN-software defined networking environment and reinforcement learning. This security mechanism is vulnerable to eavesdropping, DOS attacks, IP spoofing, falsification, obfuscation, and jamming attacks. Meidan [164] used whitelist device identification using a random forest classifier on a typical IoT network. Malware and DDoS attacks are vulnerabilities and can be exploited to compromise data integrity and availability to its respective users. Sridhar [161] in 2017 used a lattice-based cryptographic protection protocol that featured unsupervised learning for WSN-wireless sensor networks in IoT. Its shortcomings are observed during DDoS attacks and Quantum algorithm attacks. In this IoT security framework, confidentiality and availability of the data are preserved, however, the data integrity can be compromised by the attacker. Karuppiiah, Punithavati in 2019 [163] brought a lightweight version of cancelable biometric authentication protocol. It was created for smart IoT networks but its performance was discouraging because it used random forest classifiers and yet the security was compromised in terms of confidentiality, availability, and data integrity. A kernel-level machine learning-based intelligent system was proposed by Fang [160] in 2019. It is designed for 5G wireless IoT networks and uses a

Q-learning algorithm. It is vulnerable to interception and spoofing attacks. In recent 2021, Shakeel [162] derived a cross-validation mechanism for device authentication and secret key handler based on the logistic regressor algorithm. Shakeel's approach is highly vulnerable to data scavenging attacks in IoT. A privacy enhanced framework for cloud-IoT integrated security was proposed by Quan and Wang [172]. Lian and Yin, Pham derived that Although AI-empowered schemes bring security, these changes require time to adapt and learn from previous growth mechanisms and there are certain unavoidable security breaches and vulnerabilities that relate with the smart bugs, exploits and viruses. Federated learning is a solution that only requires CSs to upload local models instead of detailed data [173] 174]. Section 7 concludes this article by highlighting possible future directions for IoT security in terms of authentication and key management systems. It is obvious that there are lots of shortcomings in IoT based data networks due to the heterogeneity of devices but there is no denial in the fact that the advantages and simplicity among complex task, which is achieved due to the IoT technology are worth the network risks but require lots of improvement in cases where highly sensitive data is being shared.

Future scope of improvement and conclusion

Internet of things still has tremendous growth potential in the future despite already being so much popular technology in the cyberworld. This article systematically surveyed two of the most trending technologies, blockchain and AI, their involvement with the IoT universe, and how they improve a lot of issues in IoT while introducing certain newer challenges. It is already discussed in a separate section how newer challenges are to be addressed if we involve complex technologies with IoT to enhance performance. Like in the case of blockchain, the data handling, privacy, and level of toughness for authentication breaches rise because blockchain can store vast data very securely and is very difficult to breach security. However, less secure components of such technological integration like, sensors, actuators are vulnerable to get exploited and leak some major security parameters through which an attacker can reach higher-level nodes and eventually proceed to conquer the whole network. Devising newer cryptographic ways to generate keys in a lightweight manner that is secure for both the resource-constrained edge devices and the high-performance upper-level nodes is a major challenge in IoT [discussed in Sect. 4]. Users of IoT need to understand that cost optimization of IoT implementation and physical deployment depends on the components with varying specifications. An efficient device with a higher performance value should be the choice for picking sensing nodes [165]. In wire-

Table 3 IoT Network and Artificial Intelligence based authentication and Key Management System

Author	Protection scheme	Confidentiality	Availability	Integrity	Algorithm used	Attack identified	Application Environment
Punithavathi et al. [156]	Cancelable Biometric System (CBS)	✓	×	✓	Support Vector based CBS	Insider attacks, Malware attacks	Cloud based IoT
Bezawada et al. [157]	Behavioural Fingerprinting	×	✓	×	KNN, Decision Tree	IP Spoofing and IP Masquerading	IoT
Chakraborty et al. [158]	Challenge-response protocol	×	✓	×	NaiveBays, LogitBoost, RandomForest, SMO-SVM, Bagging, HoeffdingTree, KStar, and LogisticModelTree	Machine learning (ML)-empowered attack	IoT
Restuccia et al. [159]	Security-by-Design (Polymorphic software)	×	×	✓	Reinforcement Learning	Eavesdropping, denial-of-service (DoS), spoofing, message falsification/injection, and jamming	Software-Defined Networking
Fang et al. [160]	Kernel machine-learning-based intelligent scheme	×	✓	×	Q-Learning algorithm	Interception and Spoofing attacks	5G wireless networks
Sridhar et al. [161]	Lattice based cryptography	✓	✓	×	Unsupervised Learning	Distributed Denial of Service Attacks, eavesdropping and Quantum algorithm attacks	WSN, IoT
Shakeel et al. [162]	Cross validation	✓	✓	✓	Logistic Regression	Data Scavenging Attacks	IoT
Karupiah and Punithavathi et al. [163]	lightweight cancelable biometric authentication	×	×	×	Random Forest	User impersonation attacks, insider attacks, and user-specific key theft attacks	Smart IoT
Meidan et al. [164]	White list device identification	✓	×	×	Random Forest	Malware, DDoS attack	IoT

less sensor networks, the nodes are unevenly deployed and sensors themselves are heterogeneous in terms of the cyber-physical specifications. Some may argue that the advantage of IoT is its capability to operate among a variety of hardware specification nodes, but the problem of security to ensure data privacy and confidentiality is to be considered as well. As we surveyed various schemes, protocols, and frameworks in IoT security using keys, these keys must be shared among wireless devices security. Section 4 also discussed how such weak keys based on common parameters can be exploited to execute botnet and node impersonation attacks. Due to rising technologies in both the dark and white web cyberspace, It is a challenge to cope with newer IoT attacks identified on daily basis [166]. There will always be a constant need for improvements in security mechanisms as every security mechanism comes with its shortcomings. That is why there is a constant need for maintenance and bugs removal when security technology is deployed for public use. In the case of IoT edge security, it is a challenge to load balance the security toughness of such authentication key [discussed in Sects. 2, 3, and 4] on the low computational edge layer and higher computational fog layer. The fog layer extends cloud computing services to the lower layers of the IoT architecture.

Blockchain-based authentication architectures of IoT networks highly fault-tolerant and secure in terms of data confidentiality. However, the interaction of smaller devices with the blockchain directly is very heavy on battery power. Blockchain enthusiasts should collaborate with battery power experts [167] to increase the power capacity of devices while addressing the overheating issues of silicon chips due to consistent use. Intermediate authentication keys between edge nodes and blockchain can be pre-generated using AI. These generated keys can be stored in the same blockchain by adding additional separate blocks each containing, authentication keys, crypto-keys, data separately but still connected with crypto-links. Pre-generated session keys and other crypto keys save the amount of time a device waits for the administrator to exchange parameters and then generate such keys. A mechanism that quickly authenticates a connection requesting device and allots a pre-generated session key helps in establishing faster secure connections. Time-saving execution and improved performance may not be significant in the case of one single session established using the pre-generated session key. However, in the case of hundreds and thousands of IoT devices connecting and disconnecting to an IoT network yields significant latencies and delays the overall performing capacity of the network. A smart-artificial intelligence-based algorithm that cognitively predicts the maximum and a minimum number of keys likely to be required during a certain number of devices connecting to an IoT network, can reduce the waiting time of devices. Researchers should notice that AI-driven decision

making and feedback learning enables some consciousness to the network and by reducing the waiting time, it saves the battery power of edge devices. Theoretically, it is obvious that the larger the battery reserve, the longer the device executes its tasks in an IoT network. It ensures longer device runs and labour work from nodes before they run out of battery power. The need for lightweight security protocols always prevails where the power of the devices is concerned. Reducing key size (optimal) and randomizing the intermediate parameters that are used to generate the session and authentication keys can probabilistically increase the chances of not getting compromised. Reduced key size puts a lesser toll on edge nodes and can make a balance among heterogeneous devices. These challenges describe how certain IoT issues can lead to newer issues and make it obvious that there is still great scope for improvement in IoT technology. In this article, we reviewed various authentication schemes, blockchain-based IoT authentication frameworks, Artificial Intelligence-driven smart decisive learning authentication models. We provided the comparative analysis of various presently available works significant in IoT security from a traditional, blockchain AI perspective. We conclude this article hoping that future authentication mechanisms and key management systems will be able to address these challenges. This article is directed towards young researchers that want to explore and pursue the IoT research area, from a security perspective.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Badamasi YA (2014) The working principle of an Arduino. In: 2014 11th international conference on electronics, computer and computation (ICECCO). IEEE, pp 1–4

2. Zhou K, Yuan Y (2020) A smart ammunition library management system based on raspberry pie. *Procedia Comput Sci* 166:165–169
3. Abdelhafez HA, Halawa H, Pattabiraman K, Ripeanu M (2021) Snowflakes at the Edge: A Study of Variability among NVIDIA Jetson AGX Xavier Boards. In: *Proceedings of the 4th International Workshop on Edge Systems, Analytics and Networking*, pp 1–6
4. Sehgal A, Perelman V, Kuryla S, Schonwalder J (2012) Management of resource constrained devices in the internet of things. *IEEE Commun Mag* 50(12):144–149
5. García L, Parra L, Jimenez JM, Lloret J, Lorenz P (2020) IoT-based smart irrigation systems: an overview on the recent trends on sensors and IoT systems for irrigation in precision agriculture. *Sensors* 20(4):1042
6. Selvaraj S, Sundaravaradhan S (2020) Challenges and opportunities in IoT healthcare systems: a systematic review. *SN Appl Sci* 2(1):1–8
7. Seregni M, Sassanelli C, Cerri D, Zanetti C, Terzi S (2016) The impact of IoT technologies on product-oriented PSS: The “home delivery” service case. In: *2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)*. IEEE, pp 1–5
8. Avoine G, Canard S, Ferreira L (2019) IoT-friendly AKE: forward secrecy and session resumption meet symmetric-key cryptography. *European Symposium on Research in Computer Security*. Springer, Cham, pp 463–483
9. Xiao G, Guo J, Da Xu L, Gong Z (2014) User interoperability with heterogeneous IoT devices through transformation. *IEEE Trans Industr Inf* 10(2):1486–1496
10. Gupta A, Christie R, Manjula PR (2017) Scalability in internet of things: features, techniques and research challenges. *Int J Comput Intell Res* 13(7):1617–1627
11. Srivastava G, Parizi RM, Dehghantanha A, Choo KKR (2019) Data sharing and privacy for patient iot devices using blockchain. *International Conference on Smart City and Informatization*. Springer, Singapore, pp 334–348
12. Shaw RN, Walde P, Ghosh A (2020) IOT based MPPT for performance improvement of solar PV arrays operating under partial shade dispersion. In: *2020 IEEE 9th Power India International Conference (PIICON)*. IEEE, pp 1–4
13. Song F, Zhu M, Zhou Y, You I, Zhang H (2019) Smart collaborative tracking for ubiquitous power IoT in edge-cloud interplay domain. *IEEE Int Things J* 7(7):6046–6055
14. Dey M, Nazari A, Zajic A, Prvulovic M (2018) Emprof: Memory profiling via em-emanation in iot and hand-held devices. In: *2018 51st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, pp 881–893
15. Shwartz O, Mathov Y, Bohadana M, Elovici Y, Oren Y (2018) Reverse engineering IoT devices: effective techniques and methods. *IEEE Int Things J* 5(6):4965–4976
16. Perera C, Zaslavsky A, Liu CH, Compton M, Christen P, Georgakopoulos D (2013) Sensor search techniques for sensing as a service architecture for the internet of things. *IEEE Sens J* 14(2):406–420
17. Tsague HD, Twala B (2018) Practical techniques for securing the Internet of Things (IoT) against side channel attacks. *Internet of things and big data analytics toward next-generation intelligence*. Springer, Cham, pp 439–481
18. Deogirikar J, Vidhate A (2017) Security attacks in IoT: A survey. In: *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE, pp 32–37
19. Das AK, Bera B, Wazid M, Jamal SS, Park Y (2021) On the security of a secure and lightweight authentication scheme for next generation IoT infrastructure. *IEEE Access* 9:71856–71867
20. Yadav P, Feraudo A, Arief B, Shahandashti SF, Vassilakis VG (2020) Position paper: A systematic framework for categorising IoT device fingerprinting mechanisms. In: *Proceedings of the 2nd International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things*, pp 62–68
21. Ni T, Gu X, Wang H, Li Y (2013) Real-time detection of application-layer DDoS attack using time series analysis. *J Control Sci Eng*
22. Schellenberg F, Gnad DR, Moradi A, Tahoori MB (2018) An inside job: Remote power analysis attacks on FPGAs. In: *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, pp 1111–1116
23. Joye M, Tunstall M (eds) (2012) *Fault analysis in cryptography* (Vol. 147). Springer, Heidelberg
24. Yoshikawa M, Nozaki Y, Asahi K (2016) Electromagnetic analysis attack for a lightweight block cipher TWINE. In: *2016 IEEE/ACIS International Conference on Wireless Information Technology and Systems (ICWITS) and Applied Computational Electromagnetics (ACES)*. IEEE, pp 1–2
25. Tushir B, Sehgal H, Nair R, Dezfouli B, Liu Y (2021) The Impact of DoS Attacks on Resource-constrained IoT Devices: A Study on the Mirai Attack. *arXiv preprint arXiv:2104.09041*
26. Rizvi S, Kurtz A, Pfeffer J, Rizvi M (2018) Securing the internet of things (IoT): A security taxonomy for IoT. In: *2018 17th IEEE International Conference On Trust, Security And Privacy in Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, pp 163–168
27. Gallais A, Hedli TH, Loscri V, Mitton N (2019) Denial-of-sleep attacks against IoT networks. In: *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, pp 1025–1030
28. Ahemd MM, Shah MA, Wahid A (2017) IoT security: A layered approach for attacks & defenses. In: *2017 international conference on Communication Technologies (ComTech)*. IEEE, pp 104–110
29. Meijer C, Verdult R (2015) Ciphertext-only cryptanalysis on hardened Mifare classic cards. In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp 18–30
30. Banik S, Barooti K, Durak FB, Vaudenay S (2020) Cryptanalysis of LowMC instances using single plaintext/ciphertext pair. *IACR Trans Symmetric Cryptol* 130–146
31. Kendhe AK, Agrawal H (2013) A survey report on various cryptanalysis techniques. *Int J Soft Comput Eng (IJSC)* 3(2):287–293
32. Mallik A (2019) Man-in-the-middle-attack: understanding in simple words. *Cyberspace Jurnal Pendidikan Teknologi Informasi* 2(2):109–134
33. Rughoobur P, Nagowah L (2017) A lightweight replay attack detection framework for battery depended IoT devices designed for healthcare. In: *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions)(ICTUS)*. IEEE, pp 811–817
34. Rajan A, Jithish J, Sankaran S (2017) Sybil attack in IOT: Modelling and defenses. In: *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, pp 2323–2327
35. Amin R, Islam SH, Biswas GP, Khan MK, Leng L, Kumar N (2016) Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks. *Comput Netw* 101:42–62
36. Alani MM (2018) IoT lotto: Utilizing IoT devices in brute-force attacks. In: *Proceedings of the 6th International Conference on Information Technology: IoT and Smart City*, pp 140–144
37. Koliak C, Kambourakis G, Stavrou A, Voas J (2017) DDoS in the IoT: Mirai and other botnets. *Computer* 50(7):80–84
38. Haji SH, Ameen SY (2021) Attack and anomaly detection in iot networks using machine learning techniques: A review. *Asian J Res Comput Sci* 30–46

39. Sahmi I, Mazri T, Hmina N (2018) Security study of different threats in Internet of Things. In: *The Proceedings of the Third International Conference on Smart City Applications*. Springer, Cham, pp 785–791
40. Bhasin S, Danger JL, Guilley S, Ngo XT, Sauvage L (2013) Hardware Trojan horses in cryptographic IP cores. In: *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography*. IEEE, pp 15–29
41. Elmalaki S, Ho BJ, Alzantot M, Shoukry Y, Srivastava M (2019) Spycon: Adaptation based spyware in human-in-the-loop iot. In: *2019 IEEE Security and Privacy Workshops (SPW)*. IEEE, pp 163–168
42. Arul E, Punidha A (2020) Adware Attack Detection on IoT Devices Using Deep Logistic Regression SVM (DL-SVM-IoT). In: *Congress on Intelligent Systems*. Springer, Singapore, pp 167–176
43. Huseynov H, Kourai K, Saadawi T, Igbe O (2020) Virtual Machine Introspection for Anomaly-Based Keylogger Detection. In: *2020 IEEE 21st International Conference on High Performance Switching and Routing (HPSR)*. IEEE, pp 1–6
44. Magness JM (2020) SLiver: Simulation-Based Logic Bomb Identification/Verification for Unmanned Aerial Vehicles
45. Luo Z, Zhao S, Lu Z, Sagduyu YE, Xu J (2020) Adversarial machine learning based partial-model attack in IoT. In: *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*, pp 13–18
46. English KV, Obaidat I, Sridhar M (2019) Exploiting memory corruption vulnerabilities in connman for iot devices. In: *2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, pp 247–255
47. Nirmal K, Janet B, Kumar R (2020) Analyzing and eliminating phishing threats in IoT, network and other Web applications using iterative intersection. *Peer-to-Peer Networking and Applications*, 1–13
48. Airehrour D, Gutierrez J, Ray SK (2016) A lightweight trust design for IoT routing. In: *2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*. IEEE, pp 552–557
49. Stiawan D, Idris M, Malik RF, Nurmaini S, Alsharif N, Budiarto R (2019) Investigating brute force attack patterns in IoT network. *J Electr Comput Eng*
50. Damghani H, Damghani L, Hosseinian H, Sharifi R (2019) Classification of attacks on IoT. In: *4th International Conference on Combinatorics, Cryptography, Computer Science and Computation*
51. Kwon S, Park S, Cho H, Park Y, Kim D, Yim K (2021) Towards 5G-based IoT security analysis against Vo5G eavesdropping. *Computing* 103(3):425–447
52. Wallgren L, Raza S, Voigt T (2013) Routing attacks and countermeasures in the RPL-based internet of things. *Int J Distrib Sensor Netw* 9(8):794326
53. Pundir S, Wazid M, Singh DP, Das AK, Rodrigues JPC, Park Y (2020) Designing efficient sinkhole attack detection mechanism in edge-based IoT deployment. *Sensors* 20(5):1300
54. Chugh K, Aboubaker L, Loo J (2012) Case study of a black hole attack on LoWPAN-RPL. In: *Proc. of the Sixth International Conference on Emerging Security Information, Systems and Technologies (SECURWARE)*, Rome, Italy (August 2012), pp 157–162
55. Hariri A, Giannelos N, Arief B (2019). Selective Forwarding Attack on IoT Home Security Kits. In: *Cyber-ICPS/SECPRE/SPOSE/ADIoT@ ESORICS*, pp 360–373
56. Rajashree S, Soman KS, Shah PG (2018) Security with IP address assignment and spoofing for smart IOT devices. In: *2018 international conference on advances in computing, communications and informatics (ICACCI)*. IEEE, pp 1914–1918
57. Siddiqui ST, Alam S, Ahmad R, Shuaib M (2020) Security threats, attacks, and possible countermeasures in internet of things. *Advances in data and information sciences*. Springer, Singapore, pp 35–46
58. Aman MN, Chua KC, Sikdar B (2016) Position paper: Physical unclonable functions for iot security. In: *Proceedings of the 2nd ACM international workshop on IoT privacy, trust, and security*, pp 10–13
59. Bettayeb M, Nasir Q, Talib MA (2019) Firmware update attacks and security for IoT devices: Survey. In: *Proceedings of the Arab-WIC 6th Annual International Conference Research Track*, pp 1–6
60. Hosseinzadeh S, Rauti S, Hyrynsalmi S, Leppänen V (2015) Security in the internet of things through obfuscation and diversification. In: *2015 International Conference on Computing, Communication and Security (ICCCS)*. IEEE, pp 1–5
61. Burange AW, Misalkar HD (2015) Review of Internet of Things in development of smart cities with data management & privacy. In: *2015 International Conference on Advances in Computer Engineering and Applications*. IEEE, pp 189–195
62. Zegzhda P, Zegzhda D, Kalinin M, Pechenkin A, Minin A, Lavrova D (2016) Safe integration of SIEM systems with Internet of Things: Data aggregation, integrity control, and bioinspired safe routing. In: *Proceedings of the 9th International Conference on Security of Information and Networks*, pp 81–87
63. Doshi R, Aphorpe N, Feamster N (2018) Machine learning ddos detection for consumer internet of things devices. In: *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, pp 29–35
64. Ahmed A, Latif R, Latif S, Abbas H, Khan FA (2018) Malicious insiders attack in IoT based multi-cloud e-healthcare environment: a systematic literature review. *Multimed Tools Appl* 77(17):21947–21965
65. Uwagbole SO, Buchanan WJ, Fan L (2017) An applied pattern-driven corpus to predictive analytics in mitigating SQL injection attack. In: *2017 Seventh International Conference on Emerging Security Technologies (EST)*. IEEE, pp 12–17
66. Vojković G, Milenković M, Katulić T (2019) IoT and Smart Home Data Breach Risks from the Perspective of Croatian Data Protection and Information Security Law. In: *Proceedings of the ENTRENOVA-ENTERprise REsearch INNOVation Conference (Online)*, Vol. 5, No. 1, pp 253–263
67. Jung J, Kim J, Choi Y, Won D (2016) An anonymous user authentication and key agreement scheme based on a symmetric cryptosystem in wireless sensor networks. *Sensors* 16(8):1299
68. Wang S, Wang J, Yu Z (2018) Privacy-preserving authentication in wireless IoT: applications, approaches, and challenges. *IEEE Wirel Commun* 25(6):60–67
69. Amos R, Georgiou M, Kiayias A, Zhandry M (2020) One-shot signatures and applications to hybrid quantum/classical authentication. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp 255–268
70. Pinno OJA, Grégio ARA, De Bona LC (2020) ControlChain: a new stage on the IoT access control authorization. *Concurr Comput Pract Exp* 32(12):e5238
71. Park N, Kang N (2016) Mutual authentication scheme in secure internet of things technology for comfortable lifestyle. *Sensors* 16(1):20
72. Najib W, Sulistyo S (2019) Survey on trust calculation methods in Internet of Things. *Procedia Comput Sci* 161:1300–1307
73. Banerjee S, Odelu V, Das AK, Srinivas J, Kumar N, Chattopadhyay S, Choo KKR (2019) A provably secure and lightweight anonymous user authenticated session key exchange scheme

- for internet of things deployment. *IEEE Internet Things J* 6(5):8739–8752
74. Liu A, Alqazzaz A, Ming H, Dharmalingam B (2019) Iotverif: Automatic verification of SSL/TLS certificate for IoT applications. *IEEE Access*
 75. Panwar M, Kumar A (2015) Security for IoT: An effective DTLS with public certificates. In: 2015 International Conference on Advances in Computer Engineering and Applications. *IEEE*, pp 163–166
 76. Oniga B, Farr SH, Munteanu A, Dadarlat V (2018) Iot infrastructure secured by tls level authentication and pki identity system. In: 2018 Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4). *IEEE*, pp 78–83
 77. Jalbani KB, Jalbani AH, Soomro SS (2020) IoT Security: To Secure IoT Devices with Two-Factor Authentication by Using a Secure Protocol. In: Industrial Internet of Things and Cyber-Physical Systems: Transforming the Conventional to Digital. *IGI Global*, pp. 98–118
 78. Far HAN, Bayat M, Das AK, Fotouhi M, Pournaghi SM, Doostari MA (2021) LAPAS: lightweight anonymous privacy-preserving three-factor authentication scheme for WSN-based IIoT. *Wirel Netw* 27(2):1389–1412
 79. Wazid M, Das AK, Shetty S, Rodrigues JPC, Park Y (2019) LDAKM-ElIoT: lightweight device authentication and key management mechanism for edge-based IoT deployment. *Sensors* 19(24):5539
 80. Jena SK, Tripathy BK, Gupta P, Das S (2019) A Kerberos based secure communication system in smart (internet of things) environment. *J Comput Theor Nanosci* 16(5–6):2381–2388
 81. Barkovskis N, Ozols K, Elsts A (2021) Survey of low-power wireless network technologies for the internet of things. *Autom Control Comput Sci* 55(2):177–194
 82. Durand A, Gremaud P, Pasquier J (2017) Decentralized web of trust and authentication for the internet of things. In: Proceedings of the Seventh International Conference on the Internet of Things, pp 1–2
 83. Latif S, Idrees Z, Ahmad J, Zheng L, Zou Z (2021) A blockchain-based architecture for secure and trustworthy operations in the industrial Internet of Things. *J Ind Inf Integr* 21:100190
 84. Hong S (2020) P2P networking based internet of things (IoT) sensor node authentication by Blockchain. *Peer-to-Peer Netw Appl* 13(2):579–589
 85. Kouicem DE, Bouabdallah A, Lakhlef H (2018) Internet of things security: a top-down survey. *Comput Netw* 141:199–221
 86. Suárez-Albela M, Fraga-Lamas P, Castedo L, Fernández-Caramés TM (2019) Clock frequency impact on the performance of high-security cryptographic cipher suites for energy-efficient resource-constrained IoT devices. *Sensors* 19(1):15
 87. Samann FEF, Zeebaree SR, Askar S (2021) IoT provisioning QoS based on cloud and fog computing. *J Appl Sci Technol Trends* 2(01):29–40
 88. Kim H, Kang E, Broman D, Lee EA (2020) Resilient authentication and authorization for the Internet of Things (IoT) using edge computing. *ACM Trans Internet Things* 1(1):1–27
 89. Sha K, Yang TA, Wei W, Davari S (2020) A survey of edge computing-based designs for IoT security. *Digit Commun Netw* 6(2):195–202
 90. Rao BB, Wao AA (2021) Design a novel approach for token based authentication in IOT networkS. *Ilkogretim Online* 20(4)
 91. Ejaz M, Kumar T, Ylianttila M, Harjula E (2020) Performance and efficiency optimization of multi-layer IoT edge architecture. In: 2020 2nd 6G Wireless Summit (6G SUMMIT). *IEEE*, pp 1–5
 92. Mubarakali A (2021) An efficient authentication scheme using blockchain technology for wireless sensor networks. *Wirel Pers Commun* 1–15
 93. Muzammal SM, Murugesan RK (2019) A study on secured authentication and authorization in Internet of Things: Potential of blockchain technology. In: International Conference on Advances in Cyber Security. Springer, Singapore, pp 18–32
 94. Alshahrani M, Traore I (2019) Secure mutual authentication and automated access control for IoT smart home using cumulative keyed-hash chain. *J Inf Secur Appl* 45:156–175
 95. Ali I, Ahmed AIA, Almogren A, Raza MA, Shah SA, Khan A, Gani A (2020) Systematic literature review on IoT-based botnet attack. *IEEE Access*
 96. Yamaguchi S, Tanaka H, Ahmadon MAB (2020) Modelling and evaluation of mitigation methods against IoT malware Mirai with agent-oriented Petri net PN2. *Int J Internet Things Cyber-Assurance* 1(3–4):195–213
 97. Shamshad S, Mahmood K, Kumari S (2020) Comments on “A Multi-factor User Authentication and Key Agreement Protocol Based on Bilinear Pairing for the Internet of Things”. *Wirel Pers Commun*, 1–4
 98. Wu F, Li X, Xu L, Vijayakumar P, Kumar N (2020) A novel three-factor authentication protocol for wireless sensor networks with IoT notion. *IEEE Syst J* 15(1):1120–1129
 99. Lee J, Yu S, Park K, Park Y, Park Y (2019) Secure three-factor authentication protocol for multi-gateway IoT environments. *Sensors* 19(10):2358
 100. Zhu X, Badr Y, Pacheco J, Hariri S (2017) Autonomic identity framework for the internet of things. In: 2017 International Conference on Cloud and Autonomic Computing (ICCAC). *IEEE*, pp 69–79
 101. Sharma G, Kalra S (2018) A lightweight multi-factor secure smart card based remote user authentication scheme for cloud-IoT applications. *J Inf Secur Appl* 42:95–106
 102. Wu F, Xu L, Kumari S, Li X, Shen J, Choo KKR, Das AK (2017) An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment. *J Netw Comput Appl* 89:72–85
 103. Cui J, Zhang Z, Li H, Sui R (2018) An improved user authentication protocol for IoT. In: 2018 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC). *IEEE*, pp 59–593
 104. Li X, Niu J, Kumari S, Wu F, Sangaiah AK, Choo KKR (2018) A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments. *J Netw Comput Appl* 103:194–204
 105. Islam SH, Biswas GP (2011) Improved remote login scheme based on ECC. In: 2011 International Conference on Recent Trends in Information Technology (ICRTIT). *IEEE*, pp 1221–1226
 106. Zhang Z, Wang H, Vasilakos AV, Fang H (2012) ECG-cryptography and authentication in body area networks. *IEEE Trans Inf Technol Biomed* 16(6):1070–1078
 107. Xia H, Jia Z, Li X, Ju L, Sha EHM (2013) Trust prediction and trust-based source routing in mobile ad hoc networks. *Ad Hoc Netw* 11(7):2096–2114
 108. Shone N, Dobbins C, Hurst W, Shi Q (2015) Digital memories based mobile user authentication for IoT. In: 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing. *IEEE*, pp 1796–1802
 109. Xiong H, Qin Z (2015) Revocable and scalable certificateless remote authentication protocol with anonymity for wireless body area networks. *IEEE Trans Inf Forensics Secur* 10(7):1442–1455
 110. Xie L, Wang W, Shi X, Qin T (2017) Lightweight mutual authentication among sensors in body area networks through physical unclonable functions. In: 2017 IEEE International Conference on Communications (ICC). *IEEE*, pp 1–6

111. Sun W, Cai Z, Li Y, Liu F, Fang S, Wang G (2018) Security and privacy in the medical internet of things: a review. *Secur Commun Netw*
112. Amin R, Biswas GP (2015) A secure three-factor user authentication and key agreement protocol for tmis with user anonymity. *J Med Syst* 39(8):1–19
113. Kasyoka P, Kimwele M, Angolo SM (2020) Cryptanalysis of a pairing-free certificateless signcryption scheme. *ICT Express*
114. Tan X, Zhang J, Zhang Y, Qin Z, Ding Y, Wang X (2020) A PUF-based and cloud-assisted lightweight authentication for multi-hop body area network. *Tsinghua Sci Technol* 26(1):36–47
115. Loffi L, Westphall CM, Grüttner LD, Westphall CB (2021) Mutual authentication with multi-factor in IoT-Fog-Cloud environment. *J Netw Comput Appl* 176:102932
116. Nakamoto S (2008) Bitcoin: A peer-to-peer electronic cash system. *Decent Bus Rev*
117. Bertoni G, Daemen J, Peeters M, Van Assche G (2009) Keccak sponge function family main document. Submission to NIST (Round 2) 3(30):320–337
118. Mingxiao D, Xiaofeng M, Zhe Z, Xiangwei W, Qijun C (2017) A review on consensus algorithm of blockchain. In: 2017 IEEE international conference on systems, man, and cybernetics(SMC). IEEE, pp 2567–2572
119. Bach LM, Mihaljevic B, Zagar M (2018) Comparative analysis of blockchain consensus algorithms. In: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). IEEE, pp 1545–1550
120. Huh S, Cho S, Kim S (2017) Managing IoT devices using blockchain platform. In: 2017 19th international conference on advanced communication technology (ICACT). IEEE, pp 464–467
121. Zhang Y, He D, Choo KKR (2018) BaDS: Blockchain-based architecture for data sharing with ABS and CP-ABE in IoT. *Wirel Commun Mobile Comput*
122. Kodama F, Shibata T (2017) Beyond fusion towards IoT by way of open innovation: an investigation based on the Japanese machine tool industry 1975–2015. *J Open Innov Technol Mark Complex* 3(4):23
123. Zhang WZ, Elgendy IA, Hammad M, Iliyasu AM, Du X, Guizani M, Abd El-Latif AA (2020) Secure and optimized load balancing for multitier IoT and edge-cloud computing systems. *IEEE Internet Things J* 8(10):8119–8132
124. Park SH, Park JK (2016) IoT industry & security technology trends. *Int J Adv Smart Conver* 5(3):27–31
125. Ribeiro V, Holanda R, Ramos A, Rodrigues JJ (2020) Enhancing key management in LoRaWAN with permissioned blockchain. *Sensors* 20(11):3068
126. Cass S (2020) Nvidia makes it easy to embed AI: the Jetson nano packs a lot of machine-learning power into DIY projects-[Hands on]. *IEEE Spectr* 57(7):14–16
127. Tyson G, Sastry N, Rimac I, Cuevas R, Mauthe A (2012) A survey of mobility in information-centric networks: Challenges and research directions. In: Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications, pp 1–6
128. Gupta S, Parne BL, Chaudhari NS (2021) ISAG: IoT-enabled and Secrecy Aware Group-based handover scheme for e-health services in M2M communication network. *Fut Gen Comput Syst*
129. Shivraj VL, Rajan MA, Singh M, Balamuralidhar P (2015) One time password authentication scheme based on elliptic curves for Internet of Things (IoT). In: 2015 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW). IEEE, pp 1–6
130. Fernández-Caramés TM, Fraga-Lamas P (2018) A review on the use of blockchain for the internet of things. *Ieee Access* 6:32979–33001
131. Fiorentino G, Occhipinti C, Corsi A, Moro E, Davies J, Duke A (2020) Blockchain: Enabling Trust on the Internet of Things. *The Internet of Things: From Data to Insight* 141–157
132. Schilling L, Uhlig H (2019) Some simple bitcoin economics. *J Monet Econ* 106:16–26
133. Wood G (2014) Ethereum: a secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper* 151(2014):1–32
134. Fu MH (2020) Integrated Technologies of Blockchain and Biometrics Based on Wireless Sensor Network for Library Management. *Inf Technol Libr* 39(3)
135. Rui-jun YANG, Jin-bo HU, Yan CHENG (2019) Design of data sharing module based on medical block chain. In: 2019 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS). IEEE, pp 149–152
136. Chen Y, Meng L, Zhou H, Xue G (2021) A blockchain-based medical data sharing mechanism with attribute-based access control and privacy protection. *Wirel Commun Mobile Comput*
137. Shafagh H, Burkhalter L, Hithnawi A, Duquenois S (2017) Towards blockchain-based auditable storage and sharing of iot data. In: Proceedings of the 2017 on cloud computing security workshop, pp 45–50
138. Gagneja K, Kiefer R (2020) IoT devices with non-interactive key management protocol. *Sixth Int Conf Mobile Secure Serv (MobiSecServ)* 2020:1–6. <https://doi.org/10.1109/MobiSecServ48690.2020.9042942>
139. Dedeoglu V, Dorri A, Jurdak R, Michelin RA, Lunardi RC, Kanhere SS, Zorzo AF (2020) A journey in applying blockchain for cyberphysical systems. In: 2020 International Conference on COMMunication Systems & NETWORKS (COMSNETS). IEEE, pp 383–390
140. Ma Z, Zhang J, Guo Y, Liu Y, Liu X, He W (2020) An efficient decentralized key management mechanism for VANET with blockchain. *IEEE Trans Veh Technol* 69(6):5836–5849
141. Chen T, Zhang L, Choo KKR, Zhang R, Meng X (2021) Blockchain Based Key Management Scheme in Fog-enabled IoT Systems. *IEEE Internet Things J*
142. Panda SS, Jena D, Mohanta BK, Ramasubbareddy S, Daneshmand M, Gandomi AH (2021) Authentication and Key Management in Distributed IoT using Blockchain Technology. *IEEE Internet Things J*
143. Lou J, Zhang Q, Qi Z, Lei K (2018) A blockchain-based key management scheme for named data networking. In: 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN). IEEE, pp 141–146
144. Lei A, Cruickshank H, Cao Y, Asuquo P, Ogah CPA, Sun Z (2017) Blockchain-based dynamic key management for heterogeneous intelligent transportation systems. *IEEE Internet Things J* 4(6):1832–1843
145. Ma M, Shi G, Li F (2019) Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the IoT scenario. *IEEE Access* 7:34045–34059
146. Tian Y, Wang Z, Xiong J, Ma J (2020) A blockchain-based secure key management scheme with trustworthiness in DWSNs. *IEEE Trans Industr Inf* 16(9):6193–6202
147. Wang J, Wu L, Choo KKR, He D (2019) Blockchain-based anonymous authentication with key management for smart grid edge computing infrastructure. *IEEE Trans Ind Inf* 16(3):1984–1992
148. Jia C, Ding H, Zhang C, Zhang X (2021) Design of a dynamic key management plan for intelligent building energy management system based on wireless sensor network and blockchain technology. *Alex Eng J* 60(1):337–346
149. Chen CM, Deng X, Gan W, Chen J, Islam SH (2021) A secure blockchain-based group key agreement protocol for IoT. *J Supercomput* 1–23

150. Uddin MI, Alamgir MS, Rahman MM, Bhuiyan MS, Moral MA (2021) AI Traffic Control System Based on Deepstream and IoT Using NVIDIA Jetson Nano. In: 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST). IEEE, pp 115–119
151. Bermudez I, Traverso S, Mellia M, Munafo M (2013) Exploring the cloud from passive measurements: The Amazon AWS case. In: 2013 Proceedings IEEE INFOCOM. IEEE, pp 230–234
152. Mahdavejad MS, Rezvan M, Barekatin M, Adibi P, Barnaghi P, Sheth AP (2018) Machine learning for internet of things data analysis: a survey. *Digit Commun Netw* 4(3):161–175
153. Singh S, Sharma PK, Yoon B, Shojafar M, Cho GH, Ra IH (2020) Convergence of blockchain and artificial intelligence in IoT network for the sustainable smart city. *Sustain Cities Soc* 63:102364
154. Trakadas P, Simoens P, Gkonis P, Sarakis L, Angelopoulos A, Ramallo-González AP, Karkazis P (2020) An artificial intelligence-based collaboration approach in industrial iot manufacturing: key concepts, architectural extensions and potential applications. *Sensors* 20(19):5480
155. Došilović FK, Brčić M, Hlupić N (2018) Explainable artificial intelligence: A survey. In: 2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO). IEEE, pp 0210–0215
156. Punithavathi P, Subbiah G (2017) Can cancellable biometrics preserve privacy? *Biom Technol Today* 2017(7):8–11
157. Bezawada B, Bachani M, Peterson J, Shirazi H, Ray I, Ray I (2018) Behavioral fingerprinting of iot devices. In: Proceedings of the 2018 Workshop on Attacks and Solutions in Hardware Security, pp 41–50
158. Chakraborty N, Li J, Mondal S, Luo C, Wang H, Alazab M, Pan Y (2020) On Designing a Lesser Obtrusive Authentication Protocol to Prevent Machine Learning based Threats in Internet of Things. *IEEE Internet Things J*
159. Restuccia F, D'Oro S, Melodia T (2018) Securing the internet of things in the age of machine learning and software-defined networking. *IEEE Internet Things J* 5(6):4829–4842
160. Fang H, Wang X, Tomasin S (2019) Machine learning for intelligent authentication in 5G and beyond wireless networks. *IEEE Wirel Commun* 26(5):55–61
161. Sridhar S, Smys S (2017) Intelligent security framework for iot devices cryptography based end-to-end security architecture. In: 2017 International Conference on Inventive Systems and Control (ICISC). IEEE, pp 1–5
162. Shakeel PM, Baskar S, Fouad H, Manogaran G, Saravanan V, Montenegro-Marin CE (2021) Internet of things forensic data analysis using machine learning to identify roots of data scavenging. *Futur Gen Comput Syst* 115:756–768
163. Punithavathi P, Geetha S, Karuppiyah M, Islam SH, Hassan MM, Choo KKR (2019) A lightweight machine learning-based authentication framework for smart IoT devices. *Inf Sci* 484:255–268
164. Meidan Y, Bohadana M, Shabtai A, Ochoa M, Tippenhauer NO, Guarnizo JD, Elovici Y (2017) Detection of unauthorized IoT devices using machine learning techniques. *arXiv preprint arXiv:1709.04647*, pp 255–268
165. Raza A, Ikram AA, Amin A, Ikram AJ (2016) A review of low cost and power efficient development boards for IoT applications. In: 2016 Future Technologies Conference (FTC). IEEE, pp 786–790
166. <https://www.fireeye.com/current-threats/annual-threat-report/mtrends.html>
167. Zhao K, Wang ZL, Yang Y (2016) Self-powered wireless smart sensor node enabled by an ultrastable, highly efficient, and superhydrophobic-surface-based triboelectric nanogenerator. *ACS Nano* 10(9):9044–9052
168. Song J, Zhong Q, Wang W, Su C, Tan Z, Liu Y (2021) FPDp: flexible privacy-preserving data publishing scheme for smart agriculture. *IEEE Sensors J* 21(16):17430–17438. <https://doi.org/10.1109/JSEN.2020.3017695>
169. Xiong Z, Zhang Y, Niyato D, Wang P, Han Z (2018) When mobile blockchain meets edge computing. *IEEE Commun Mag* 56(8):33–39
170. Liu M, Yu FR, Teng Y, Leung VC, Song M (2018) Distributed resource allocation in blockchain-based video streaming systems with mobile edge computing. *IEEE Trans Wirel Commun* 18(1):695–708
171. Wang W, Qiu C, Yin Z, Srivastava G, Gadekallu TR, Alsolami F, Su C (2021) Blockchain and PUF-based lightweight authentication protocol for wireless medical sensor networks. *IEEE Internet Things J*
172. Wang T, Quan Y, Shen XS, Gadekallu TR, Wang W, Dev K (2021) A privacy-enhanced retrieval technology for the cloud-assisted Internet of Things. *IEEE Trans Ind Inform*
173. Wang W, Fida MH, Lian Z, Yin Z, Pham QV, Gadekallu TR, Su C (2021) Secure-enhanced federated learning for ai-empowered electric vehicle energy prediction. *IEEE Consumer Electron Mag*
174. Lian Z, Wang W, Su C (2021) COFEL: Communication-Efficient and Optimized Federated Learning with Local Differential Privacy. In: ICC 2021-IEEE International Conference on Communications. IEEE, pp 1–6

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.