IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

International Journal of Safety and Security Engineering
Vol. 11, No. 6, December, 2021, pp. 641-651
Journal homepage: http://iieta.org/journals/ijsse

IoT Modes of Operations with Different Security Key Management Techniques: A Survey

Samah Mohamed[1], Ayman M. Hassan[1], Heba K. Aslan[2,3*]

[1] Faculty of Engineering, Banha University, Banha 13512, Egypt
[2] Informatics Dept., Electronics Research Institute, Cairo 12622, Egypt
[3] Centre of Informatics Science, Faculty of Information Technology and Computer Science, Nile University, Giza 12588, Egypt

Corresponding Author Email: haslan@nu.edu.eg

https://doi.org/10.18280/ijsse.110604

Received: 11 October 2021
Accepted: 2 December 2021

Keywords:
IoT security, key management, key generation, key distribution, modes of operation, map link, IoT challenges, security goals

ABSTRACT

The internet of things (IoT) has provided a promising opportunity to build powerful systems and applications. Security is the main concern in IoT applications due to the privacy of exchanged data using limited resources of IoT devices (sensors/actuators). In this paper, we present a classification of IoT modes of operation based on the distribution of IoT devices, connectivity to the internet, and the typical field of application. It has been found that the majority of IoT services can be classified into one of four IoT modes: gateway, device to device, collaborative, and centralized. The management of either public or symmetric keys is essential for providing security. In the present paper, we survey different key management protocols concerning IoT, which we further allocate in a map table. The map table is a link between modes of operation and the associated security key management elements. The main target of this mapping table is to help designers select the optimum security technique that provides the best balance between the required security level and IoT system mode constraints.

1. INTRODUCTION

The evolution of traditional networks starts with connections between computers until the internet of things (IoT) emerges. IoT is a new kind of world interconnection between highly heterogeneous devices and appliances. The main advantages of this high technology are its high integration with social media, automated monitoring and ability to help make decisions in a cooperative way [1]. Figure 1 shows the data life cycle for any IoT architecture. This life cycle consists of five sequence stages, which are described as follows:

- The first stage represents the collection stage where the external data from different data sensors are collected.
- In the second stage, collected data is stored in the IoT cloud after being transmitted through smooth digitized communication with low power consumption methodologies, such as the following:
  - Short range communication (< 1m) such as Radio Frequency Identification (RFID) and Near Field Communication (NFC);
  - Medium range communication (1m - 10km) such as Bluetooth, Zigbee, Wi-Fi, Narrow Band (NB-IoT) Long Term Evolution Machine (LTE-M), and 5G;
  - Long Range Wireless Communication (> 10km) such as Low Power Wide Area Network (LPWAN), Very Small Aperture Terminal (VSAT), Low Power Wide Personal Area Network (6LowPAN), and IPv6.
- In the third stage, stored data is processed/analysed in the IoT cloud for knowledge extraction.
- The fourth stage involves sharing the processed data according to the user application.
- The fifth and final stage involves filtering all data, choosing only the needed data, and discarding the rest.

Given that this life cycle involves collecting a huge amount of data through sensing devices; this raises the need for a massive computation. In IoT, analysis, storage and processing of these collected data is difficult due to resources limitation. Alternatively, cloud computing is useful for handling massive data and realizing time reliability, scalability, flexibility, and security, as shown in Figure 2.

Figure 2 shows the principal of fog computing [2, 3], which is displayed close to the ground to save the need for expensive communication. Its main purpose is facilitating the optimal performance of IoT in the cloud by applying data that is stored, processed, filtered, and analysed on the edge of the network before being transferring to the cloud.

IoT security is the technology area concerned with safeguarding connected devices and networks on the internet. It plays a central role with no margin for error or shortage of supply. Moreover, security is important for delivering high-quality services with efficient costs, management, and monitoring. We describe the main security goals as follows [4, 5]: privacy, access control, authentication, authorization, integrity, and nonrepudiation. These security goals can be realized by different techniques of security elements, such as encryption and key management with authentication and verification certificates.

To achieve the above-mentioned security goals, users must use encryption algorithms. For proper application of these encryption algorithms, the key used in encrypting either public or symmetric keys must be distributed in a secure manner. This paper focuses on the classification of IoT modes of operation with the following two main security key management elements: key generation and key distribution.

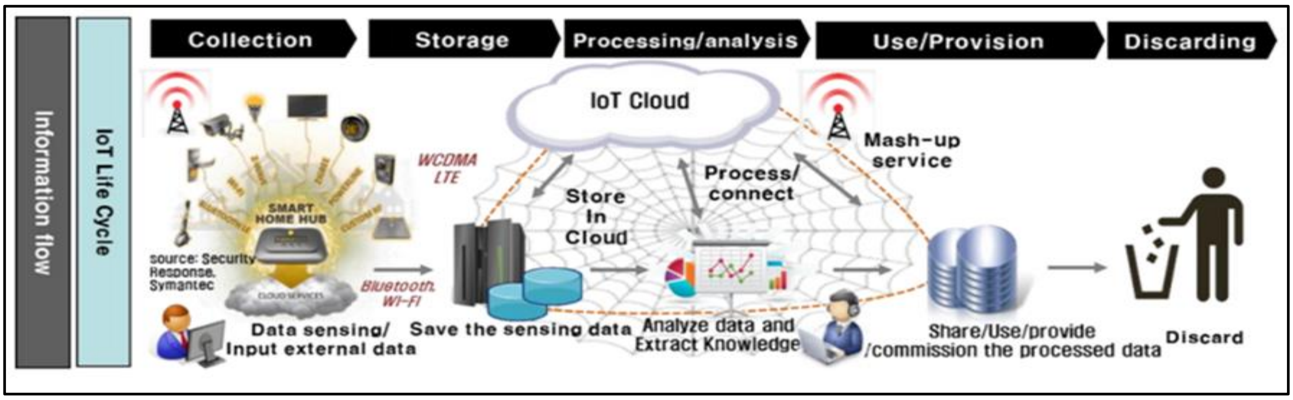


Figure 1. IoT architectures with the data life cycle

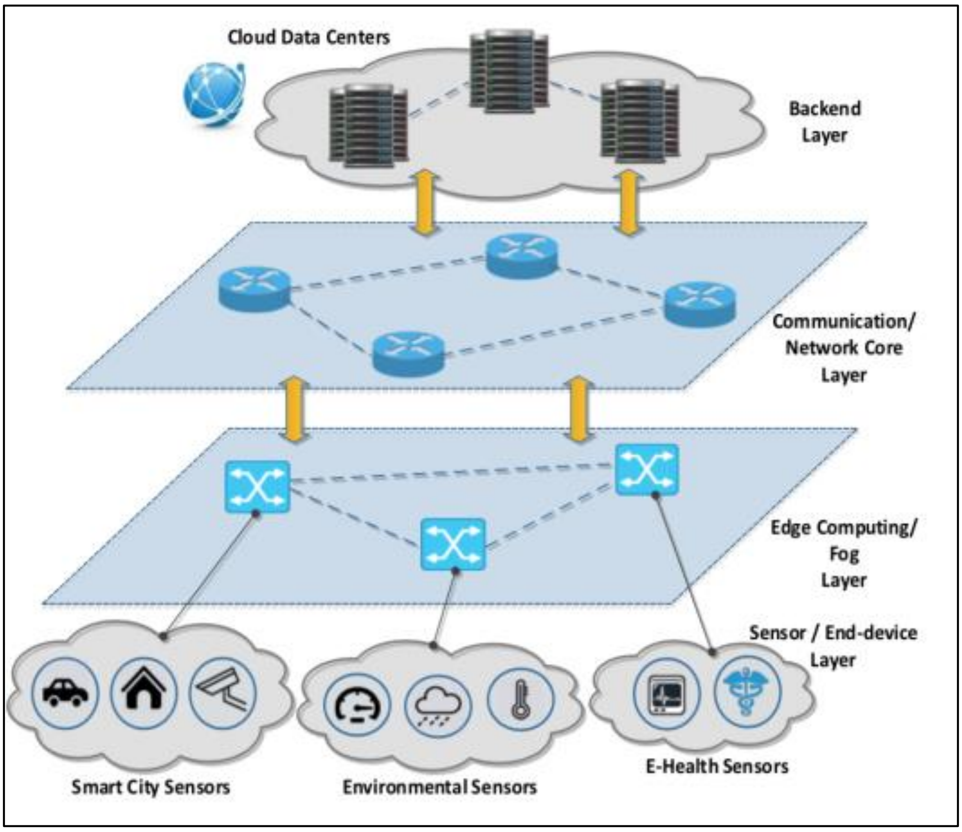


Figure 2. An introduction to fog computing [6]

The literature has proposed different solutions that solve the problem of key generation and distribution for IoT systems. This paper provides a mapping between these solutions and each IoT mode of operation. This map can help designers select the optimal security techniques that provide the best balance between the required security level and IoT system mode constraints. We summarize the contributions of the paper as follows:

-We classify various IoT architectures into four main modes of operation.

-We survey different key generation and distribution protocols.

-We map the results of the survey onto different IoT modes of operation.

The article is organized as follows: Section 2 presents a literature review concerning both key generation and distribution. Section 3 illustrates IoT modes of operation with a matrix map link between the modes and the main security elements. Finally, Section 4 provides a conclusion to the work.

2. LITERATURE REVIEW

This section illustrates the main security components of key management, key generation and distribution, for realizing the overall IoT security goals.

2.1 Key management

Key management refers to the management of cryptographic keys in a cryptosystem, as shown in Figure 3. The figure shows three types of cryptographic techniques, which are summarized below [7]:

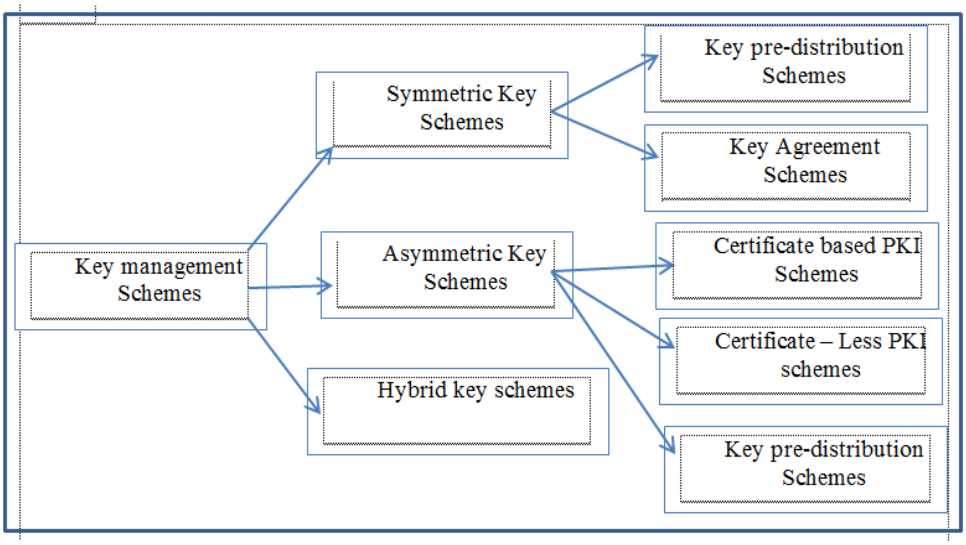

Key management Schemes
Symmetric Key Schemes
Asymmetric Key Schemes
Hybrid key schemes
Key pre-distribution Schemes
Key Agreement Schemes
Certificate based PKI Schemes
Certificate – Less PKI schemes
Key pre-distribution Schemes

Figure 3. Key management classification [7]

-A symmetric technique is a single shared secret key between the sender and the receiver. In this category, the key management is classified into pre-distributed technique and key agreement one. In the pre-distributed technique, the secret key is distributed before any system deployment by a Trusted Third Party (TTP). In contrast, the key agreement technique involves parties agreeing on a common secret key to be used after deployment.

-An asymmetric technique refers to when a pair of keys is used for data exchange between the sender and receiver. Figure 3 shows that the asymmetric key management scheme is classified into the certificate category, certified-less category, and key pre-distribution. The difference between them is the existence of a third party in certificate-based schemes to certify the authenticity of the user's public key. In the certificate-less category, however, the users certify their own public keys without the existence of TTP.

-A hybrid technique refers to a combination of symmetric and asymmetric key management schemes.

Key management also involves strategies for addressing major key functions such as generation, exchange, storage, distribution, and refreshing [8, 9]. Some examples are cryptographic protocol design, key servers, user procedures, and other relevant protocols. Successful key management is still considered a fundamental tool for ensuring the security of IoT devices which are characterized by limited resources. Therefore, there is a need for new schemes of key generation and distribution [10]. The following subsections present different key generation and distribution techniques.

2.1.1 Key generation

Key generation entails moving from a world where there is no key to a world where there is one. A "key" here refers to a value with a certain length for specific cryptographic algorithms (e.g., an Advanced Encryption Standard (AES) key is a sequence of 128, 192, or 256 bits, while an Rivest-Shamir-Adleman (RSA) key is 1024 bits or greater). Since keys have values that are unpredictable to third parties, key generation necessarily involves using source data that is unknown to other people. In general, this "source data" refers to random values obtained from a suitable source [11].

Broadly speaking, key generation can be classified into two types. The first type is authenticated key generation, where one of the communication parties can verify the identity of the other device. The second is an unauthenticated key generation, which is simply used to generate a pair of keys without authenticating each other [12].

Key generation protocols are classified according to cryptographic techniques, applications, the number of trusted members, and passes [3]. The remainder of this section proivde a survey of articles that have addressed different key generation techniques under IoT systems.

Tsai et al. [13] discuss key generation to enhance security in machine to machine (M2M) communication by providing an automatic key update mechanism for IoT devices. This scheme is based on Constrained Application Protocol (CoAP), which research has identified as a suitable solution for IoT applications. Their experiment results show that 1.03% latency overhead is added for better security performance. The key update flow consists of using Pseudo-Random Number Generators (PRNG) for the generation of the 32 bits LSB, which is then used for the generation of the other 32 bit MSB.

Furtak [12] presents a cryptographic Key Generating and Renewing system (KGR) which is applicable for clusters of IoT nodes or other systems. This system's working principle consists of the following two parts:

There is first a hardware component that uses a Trusted Platform Module (TPM) for key generation and the protection of stored and exchanged data.

The second part is a software component that utilises Message Queuing Telemetry Transport (MQTT) protocol to exchange data between nodes of the KGR system.

Malche et al. [14] present the Riddle and Code Secure Element (RCSE) to generate private and public key pairs. RCSE provides strong secured authentication, hardware-based cryptographic key storage and cryptographic countermeasures. It has a crypto accelerator, storage, supports HMAC & SHA256 Hash, AES-128 encryption, secure boot, and Firmware Authentication.

Wazid et al. [15] discuss Lightweight Device Authentication and Key Management Scheme for the Edge-based IoT environment (LDAKM-EIoT). In LDAKM-EIoT, the keys are managed based on efficient operations using exclusive OR (XOR) with one-way collision-resistant cryptographic hash functions.

Eldefrawy et al. [16] address authenticated key broadcasting through the Chinese Remainder Theorem (CRT) and the use of three nested hash functions. Each Industrial Internet of

Things (IIoT) node nj is loaded with the following three unique CRT modules: rj,a, rj,b, and rj,c. These modules are also primed to one another. Following this, nodes use a key distribution protocol with low computational cost operations (only hash and XOR operations) to be compatible with IIoT.

SKYGlow refers to the name of a secret key generation scheme. This scheme targets resource-constrained IoT platforms by applying the Discrete Cosine Transform (DCT) for the exchanged messages [10]. This reduces mismatches and increases the correlation between the generated secret bits. SKYGlow has a high performance in both indoor and outdoor scenarios at 2.4 GHz and 868 MHz, respectively. The results suggest that SKYGlow can create a secret of 128 bit keys of 0.9978 bits entropy with just 65 packet exchanges.

Thirumalai and Kar [17] provide a secure information exchange between cloud-to-IoT and IoT-to-IoT devices. This exchange is based on the authors' proposal of a new variant of RSA: the Memory Efficient Multi Key (MEMK) generation scheme. The principle of working for providing this memory efficiency is to reuse the RSA scheme with a Diophantine form of the nonlinear equation and does not use of multiplicative inverse function or Extended Euclidian algorithm.

Chugunkov et al. [18] use a modified function of feedback (OFB) in AES for Light Weight Pseudo-Random Number Generators (LW-PNRG). The main advantage of this function is to decrease the classic generator parameter. The basis of any LW-cryptosystem is symmetric algorithms. Symmetric algorithms are more efficient than asymmetric ones because they have a higher throughput. This advantage is essential for IoT devices with low computation power.

Liu et al. [19] use the tetrahedral oscillator for random key generation. The main advantage of this technique is being able to implement the design on 0.13um technology with 100Kb/s bit rate, which passes the NIST and diehard test.

Finally, we noticed from the abovementioned key generation techniques that they can be implemented either hardware such as being presented in articles [13, 14, 19] or standard common software schemes such as CoAP, CRT, DCT, MEMK, and pseudorandom number generator which are realised in articles [10, 12, 13, 16-18]. According to the system configuration, designers can choose the suitable method for key generation. The following subsection, moves to detail different key distribution protocols proposed for IoT.

2.1.2 Key distribution

In key distribution protocols, entities can be in the same or different security domains. In addition, the keys are either distributed by incorporating a trusted authority or calculated by the entities themselves [18]. We classify key distribution protocols into two classes:

•The first class is point to point key establishment between the entities without the use of a trusted authority. This method is applied in small systems. The key can either be symmetric, allowing the two parties to share the same key [20-23], or asymmetric, where each of the parties has a public key with an associated private key. A public key certificate for the intended recipient is used to realize data integrity and data origin authentication in a confidential way.

•The second class refers to key establishment where a TTP is used. The TTP is authorised to offer key management services such as generation, certification, distribution, and translation of keying material [18, 20-22]. Thus, the TTP is considered as Certification Authority (CA) where in a large system the key management is organised in a hierarchical way.

Figure 4 displays different forms of key establishment and initiation, as described below:

•The first form is the Key Distribution Center (KDC), whose main task is to generate and distribute keys upon entity requests. These keys can be distributed directly to entities, as shown in Figure 4(i), or sent back to the initiator to then forward them to the receiver entity, as shown in Figure 4(ii).

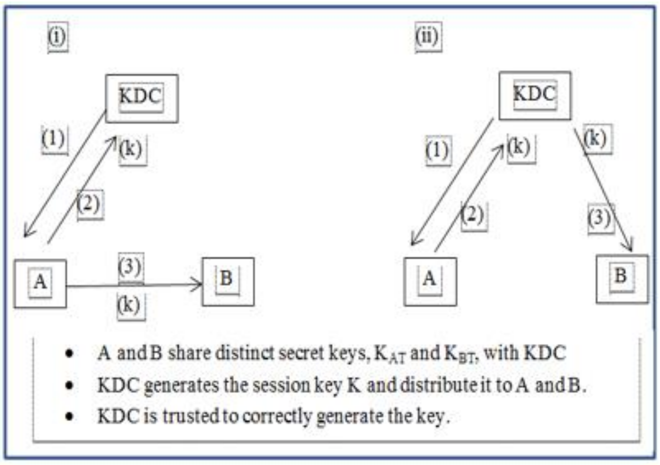
•The second form is the Key Translation Center (KTC), which receives the generated key from the entity. It then handles this key in a similar manner to the KDC. However, in the case of an asymmetric key, each entity contacts its authority to receive an appropriate public certificate [18, 20-22].

Previous studies have proposed several protocols for key distribution in IoT.

Sun et al. [24] propose a solution for securing data collected from wearable devices. Their solution is based on the use of Signed Sliding Window Coding (SSWC) in biometric cryptography technology. The SSWC principle of work is based on the extraction of the common feature for sharing the generated M-bit key among devices worn on different body parts. This M-bit is generated from lightweight noise signals with high randomness and bit generation rates.

Concone et al. [25] discuss the Secure Mobile Crowd sensing Protocol (SMCP) for fog-based applications. This article presents a low-power mobile edge through the use of a lightweight encryption technique such as ECC and Extended Triple Diffie-Hellman key agreement. The latter is particularly suitable for low-power mobile devices. The authors concentrated on designing a secure fog with an edge layer consisting of cheap wearable devices with limited computational resources. Fog works as an interface for remote (Cloud) data centers in real time.

Dammak et al. [26] introduce a novel Decentralized Lightweight Group Key Management Architecture for Access Control (DLGKM-AC) in IoT environments. This scheme is based on a hierarchical architecture composed of one KDC and several Sub Key Distribution Centers (SKDCs). Its main advantages are as follows: reducing storage, lowering the computation and communication overheads in IoT dynamic environments, and lower the rekeying overhead on the KDC.



- A and B share distinct secret keys, K_{AT} and K_{BT}, with KDC
- KDC generates the session key K and distribute it to A and B.
- KDC is trusted to correctly generate the key.

Figure 4. Key distribution center

Using Key pre-distribution Schemes (KPIs) [27] is a type of key agreement for selecting keys called session key to provide secure communication. This scheme reduces the key distribution overhead on devices since the key distribution is performed before the positioning of IoT devices. The scheme provides better results in the three important factors, which are connectivity, scalability, and resiliency. KPS methods are categorised into three types of schemes: random, deterministic, and hybrid groups. While the random scheme is based on selecting key rings in a random way, the deterministic scheme is based on key rings having a specific determination. Finally, the hybrid scheme is a combination of both the deterministic and random schemes.

Shin et al. [28] address Distributed IP Mobility Management (DMM) in smart home networks as one of the centralized applications which suit 5G networks. This mobility management increases the vulnerability to various security threats. This raises the need to ensure route optimisation, provide secure direct device communications, and realize information privacy. This proposed protocol is composed of both Route Optimization Initialization (RO-INIT) and Handover phases (RO-HO). The main purpose of this protocol is to provide key exchange, authentication, and perfect forward secrecy and privacy protection. This protection is verified by two formal security analysis tools: BAN and Automatic Validation of Internet Security Protocols and Applications (AVISPA).

Rabiah et al. [29] realise security authentication in key exchange protocol through a wireless channel between IoT devices and gateway communication. This protocol is implemented based on each pair of devices having two unique keys: a master key and an initial session key. Moreover, there is a constantly changing session key every session. This session key is a symmetric key that provides authentication, key exchange, confidentiality, and message integrity by applying Hashed Message Authentication Code (HMAC) and Key Derivation Function (KDF).

Guo et al. [30] use an Access Polynomial Based Self – healing Group key Distribution (AP-SGKD) protocol to secure group communication and improve communication efficiency. The AP-SGKD protocol's main feature is the extraction of the session key from the current broadcast message instead of a request for an update message. This leads to a shorter recovery time and lower communication overhead. This protocol also satisfies all basic security properties with optimal storage requirements.

Leshem et al. [31] present the unique encryption key construction and distribution per conversation with frequent changes of keys at each IoT device. This scheme is based on ensuring the existence of a common key between any pair of IoT devices in a predefined probability analysis which is set by the designer of the system.

Eldefrawy et al. [16], present a low computational cost key distribution protocol (using only Hash and XOR) to be compatible with IIoT. This lightweight protocol handles node addition and revocation with fast re-keying, and it is applicable for single message exchanges only. It also provides forward/backward secrecy from capture and server impersonation attacks. This key distribution protocol is verified with a security validation tool called Scyther.

Moharana et al. [32] discuss the IoT node's security over virtual network using two steps. First, they generate the total unique keys using the Balanced Incomplete Block Design (BIBD) model, which is distributed from Cloud Service Provider (CSP) to the different user groups over a secure communication channel. The main user group tasks involve using a routing table to track all external and internal communication and examining the destination IP of the packets through its gateway. Another important task is using a pairwise key exchange Diffe-Hellman (DH) protocol between IoT nodes. Finally, the authors use only XOR and the shared secret key for the encryption and decryption of messages that are transmitted between nodes.

Granjal et al. [33] propose a solution for key distribution between the device and service provider without the existence of a TTP. This scheme is based on providing two identity security schemes: encryption and signature. These schemes are proved by the random oracle model. The main aim of this proposal is to reduce computational operations on the smart meter side.

In summary, we note that there are various key distribution techniques that researchers have implemented to realise data security. In order to have low computation power to fulfill the restricted IoT devices resources, the abovementioned techniques are based on lightweight techniques such as: HMAC and HKDF [29], Hash and XOR operations [16], IP identity-based authentication techniques [28, 29], and Diffie-Hellman with XOR [32].

3. IOT MODES OF OPERATIONS

The IoT has a great impact on life applications to enhance the quality of life and to boost the world's economy. Common IoT domains include smart homes, wearable devices, industrial internet, smart cities, agriculture, energy engagement, and healthcare. The main elements in these applications are sensing devices, gateways (GW), analytics, and storage servers. Despite the aforementioned domains all being categorised under the IoT title, they have considerably different architectures and modes of connectivity. For example, the smart metering application requires relatively small and timely uplink traffic from utility meters to the central server, whereas smart city applications require asynchronous transmission between vast amounts of devices and sensors that require higher data rates. Key management is highly affected by the attributes of the IoT service, including the topology, number of targeted users, and nature of the data. Sensing applications that exchange real-time environmental information are updated frequently and have a short lifetime. These applications thus do not require a high level of security, and the keys can be exchanged less frequently. In some cases, the keys can even be hardcoded at the IoT device itself. Utility metering, however, includes sensitive customer information that affects privacy and security, which calls for stronger security measures. In this case, periodic key generation and exchange should be used.

Based on this survey of presented schemes, we define four different modes of IoT operation. We define mode as the way in which data is exchanged between IoT devices and the cloud. This classification aims to help designers select the optimum security scheme for different applications.

3.1 Centralized mode

In a centralized network, a central server is connected to the various nodes. This central server receives requests from nodes and then assigns tasks to them. In this mode of operation, a Global Positioning System (GPS) can be used for phone location detection, which is one of the most important pieces of contextual information for smart applications [34].

The main features of the centralized mode are as follows:

- Easy to maintain, manage, secure, and control through a centralized platform;
- Reduced costs, as redundancies of storage and processing power are avoided;
- Single-point network failure.

The risks of data centralization are as follows:

- Information security, as information is often stored on centralized servers, which increases the potential for hacks and leaks;
- Strict hardware specifications, which may lead to inefficiencies with non-standard devices and nodes.

In article [28], the authors discuss applications for smart home systems which work as use cases for this mode. It is based on a secure route optimisation protocol for DMM applied on the smart home systems at 5G networks. This is done through the following steps: key exchange, mutual authentication, privacy protection, and Perfect Forward Secrecy (PFS). The scheme's main elements are Mobile Node (MN), Mobility GW (MGW), Context Mobility Database (CMD), and Home GW (HGW). The MGW and HGW should mutually authenticate each other while negotiating a master session key. From this master key, the sub-session keys are derived to protect the data traffic transmitted over a smart home network. The main two phases of the protocol are as follows: the RO-INIT and RO-HO phases.

3.2 Device to device mode

Device to Device (D2D), or M2M, refers to communication between two devices. It is considered a peer-to-peer application and communication. Keeping the frequency spectrum or same sharing frequency provides the systems with the following advantages: using the same frequency spectrum to improve overall throughput, spectrum utilization, and energy efficiency. It could be a cluster of houses connected under one IP in the same network to reduce computation and communication. The common network used in this mode is the Zigbee network with Bluetooth Low Energy (BLE) communication. Another adding feature is the direct connections between devices in emergency time like fire [35].

This mode can be used in applications that use technology such as 4G, LTE, and 5G, where the data in gigabytes can be transferred to minutes. In other words, this mode can be used in the most updated applications with high-end technologies and services such as web browsing, streaming, and social media, which all need a high data rate.

The most commonly used mode is Wireless Ad-Hoc-Network mode (WANET) where devices can access each other's resources directly through a basic point-to-point wireless connection. All functions related to routing, network operations, security, addressing, and key management are performed by a collection of device nodes without the need for any central servers. Shen et al. [36] provide a use case which is based on using the Diffie-Hellman key agreement protocol with secret key extracted from physical channel characteristics using the computational hardness of discrete algorithms which are based on randomness and uniqueness of wireless fading channel properties. However, the use of discrete algorithms for secret key extraction resulted in a lower key generation rate and higher communication overhead. The main application of this protocol is for integration with the existing Wi-Fi direct protocol.

The GW is an important key element in certain IoT modes of operation for facilitating general connections between sources and destinations (either device to device or device to cloud), as shown in Figure 5.

Table 1. IoT secure key management surveyed articles

Key Generation		**Key Distribution**	
Reference no./year	**Techniques used**	**Reference no./year**	**Techniques used**
[13]/2020	Pseudo Random Number Generator (PRNG)with Libcoap (Const Table 1 displays a summary of the above-mentioned surveyed security articles classified for key generation and key distribution. rained Application Protocol)	[24]/2021	Sliding Window Coding (SSWC) as a light weight noise group key generation based on common feature extraction for sharing the generated M-bit Key.
[12]/2020	Key Generating & Renewing System (KGR)	[25]/2020	Secure Mobile Crowd sensing Protocol (SMCP)
[14]/2020	Ridde and Code secure element	[26]/2020	Decentralized Lightweight Group Key Management Architecture (DLGKM-AC)
[15]/2019	Lightweight Device Authentication &Key Management mechanism for the Edge based IoT environment (LDAKM-EIoT)	[27]/2020	Key Pre-Distribution schemes (KPSs)
[16]/2018	Chinese Reminder Theorem (CRT)	[28]/2019	Distributed IP Mobility Management (DMM) and Route Optimization Initialization (RO-INIT) and Handover Phases (RO-HO)
[10]/2018	Discrete Cosine Transform (DCT)	[29]/2018	Pair of devices with unique keys (master & initial session key) provided at configuration time – Hashed Message Authentication Code (HMAC) based Key Derivation Function (HKDF)
[17]/2017	Memory Efficient Multi Key Generation Scheme (MEMK)	[30]/2018	Self-Healing (SH) group key distribution
[18]/2016	Use modify function of feedback in AES for pseudorandom number	[31]/2018	Probability of Key construction and distribution with frequent message changes
[19]/2016	Tetrahedral oscillator with large jitter	[16]/2018	Hashing and XORing
		[32]/2017	Balanced Incomplete Block Design (BIBD) – Diffie- Hellman (DH) and XORing
		[33]/2015	Based on identity based encryption and a signature from Security Manager (SM)

3.3 Sensor to GW mode

The IoT GW has evolved to perform many tasks, from data filtering to visualization to complex analytics. The IoT GW connection tasks are as follows:

- Facilitating communication with connected devices with the non-internet connection;
- Data pre-processing, aggregation, filtering, and optimization, followed by storing, buffering, streaming, visualization, and analytics;
- Managing user access by device configuration management;
- Managing network security features and system diagnostics.

The Open Automation Software (OAS) platform performs data aggregation and networking functions. It can operate both in the data source and the cloud. The OAS platform is considered a flexible solution for most IoT and IIoT implementations.

Figure 6 illustrates the GW architecture layers of the IoT. First, GW is equivalent in responsibility to certain layers such as monitoring, preprocessing, storage, and security. Moreover, GW acts as an interface block between smart devices and the cloud data center or server [21].

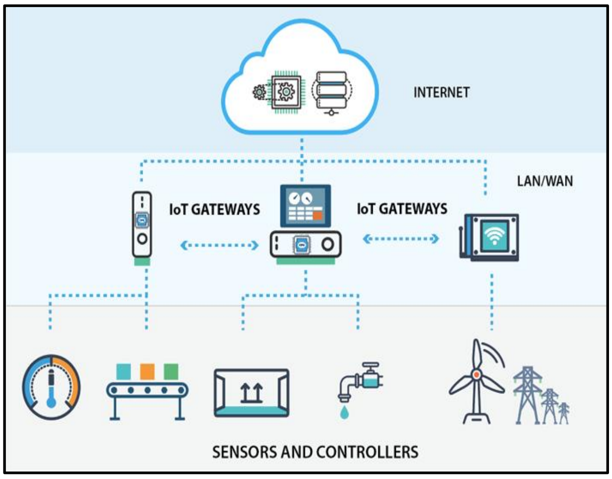


Figure 5. GW connection with sensors devices

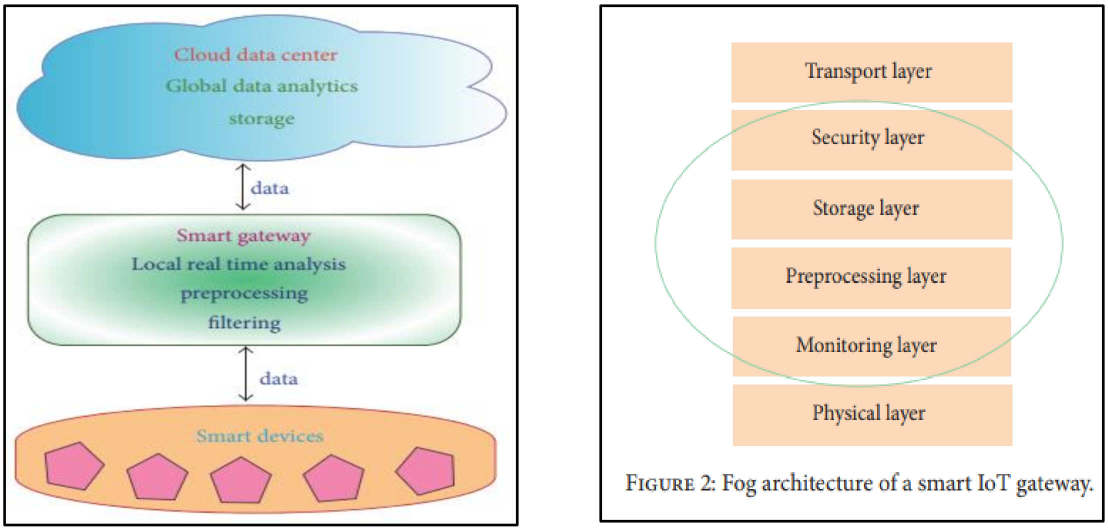


Figure 6. Smart GW for preprocessing

The common use case for this mode is Smart Metering (SM) applications where the power consumption of home appliances is sent from a group leader to SM in the following two cases [37]:

First, in the case of a small number of appliances, we do not need to group the appliances. Direct communication between leaders and devices to collect power consumption data via one-to-one is utilized.

Second, in the case of a large number of appliances, the whole home appliances work as a group where SP updates the group in a WBAN environment. The authors of article [37] realise security in SM group environments based on key exchange with group authentication in a dynamic way. Its main target is collecting data from sensors then transmitting it to SP which works as a gateway. SM group communication is better than one-to-one communication in relation to reducing communication overhead.

The scheme realises security by applying the following three phases [37]:

1. Node registration phase
2. Group authentication phase
3. Group session key distribution phase

In the node registration phase, the secret values are securely distributed to nodes using a threshold secret sharing scheme. This is performed by applying the following steps:

- Generating a single group master key by SM;
- Using this master key to generate a secret key for the participating meter nodes;
- Employing the secret key to distribute a session key.

In the group authentication phase, the SM checks all nodes by applying the following steps:

- Requesting authentication from meter nodes;
- Generating authentication reusable tokens using the secret values received at the registration phase by meter nodes;
- Confirming the nodes group by generating tokens from meter nodes to SM.

Finally, the group session key distribution phase is where the group leader encrypts each session key with a different secret key generated from SM for each meter node.

Figure 7 shows how the group keys are generated and distributed. While the GW controls group leaders, Meter Data Management System (MDMS) controls GWs and communicates with the subscriber authentication server with high computational efficiency. This scheme mainly aims to be applied to smart cities in the IoT, Cloud, Big data, and Mobile technology (ICBM) clusters.

Rabiah et al. [29] use case for the GW mode. This case is based on how each pair of devices has two unique keys: the master key (K_m) which is provided at the configuration time as long term key, and the latter is a symmetric session key(K_s) with its initial session key (K_{iks}) which is changed for each frame or message as short term key. This scheme is also based on HMAC, which is in turn based on HKDF. The scheme is applicable with the disconnection of IoT environments because keys are never exchanged over the network. In addition, no TTP is needed in this scheme. The main advantages of this scheme are less computation, less memory, and low energy usage. It maintains Perfect Forward Secrecy (PFS) property, as attackers need to know both the master and session keys to decrypt the message. Moreover, New (K_s) depends on the previous one for each session which is called (Cumulative).

3.4 Collaborative mode (smart device to mobile)

New crowd sensing applications are considered common use cases for the collaborative mode [6, 38]. This mode opens good progress in advanced IoT real applications. Examples of these real applications are the smart parking system and smart healthcare system. The collaborative model enhances the computing capabilities of the complex and huge data processing by combining processing capabilities and sensing based on the Mobile Cloud Computing (MCC) paradigm.

The collaborative mode of operation is an approach for overcoming the centralized mode drawback which leads to communication system delays [39, 40]. This latency drawback has a significant negative impact on special multimedia data, such as applications based on video and image acquisition devices. This mode of operation has been applied in various network layers and their attendant computing platforms. The main approach for applying this mode is reaching the optimisation use of computational resources of an IoT environment.

Concone et al. [25] illustrate one of the collaborative mode use cases called 'Secure Mobile Crowd sensing Protocol (SMCP) for fog-based applications". Th scheme is based on two types of lightweight encryption techniques. The first technique is Extended Triple Diffie-Hellman Key agreement (X3DHKA), which is applied in Android and iOS applications for low power mobile devices. The second technique is Elliptic Curve Cryptography (ECC). The main components of this scheme are as follows:

- The Edge Device's (ED's) main task is capturing raw data, such as done in wearable devices.
- Fog Devices (FD) with large computing power manage a different numbers of EDs. These devices works as real-time interfaces between EDs and remote cloud data centers.
- Cloud Data Center (CDC) supervises the new registration of the system with its edge and fog devices according to the corresponding applications.

3.5 Main factors of IoT connectivity

Data rate, coverage, and energy efficiency are the three key technical criteria required in IoT modes of operations. We provide the following descriptions of these parameters.

Data rate (on up/downlink): IoT application data rates range from a few hundred bits per second (bps) for metering to several Megabits per second (Mbps) for uplink video. As the complexity of IoT applications increases, we need higher data rates. Wi-Fi and cellular networks have high data rates with short-range bandwidth or complex waveforms and adaptive modulation.

Coverage: To connect devices, all IoT applications require strong coverage, while others just require coverage in specific interior locations such as: smart home applications mode and D2D mode; however, others demand significant coverage in distant areas such as in GW and collaborative modes. Coverage can be applied by various cellular technologies which can be classified into outdoor and indoor categories. Outdoor technologies, such as using 3G or 4G, are strong examples of wide area solutions. Indoor technologies, which are characterised by short-range communication, use Wi-Fi and Zigbee technologies.

Energy efficiency: The energy efficiency of a connection technology has a major impact on the lifetime of the maintenance cycle for IoT devices that rely on battery or energy harvesting. Moreover, it is influenced by the usage of the application's main parameters, such as topology, complexity, the connectivity technology's range, and the duration and frequency of message transmission.

For example, Zigbee is a short-range technology based on a mesh topology between devices over multiple hops. Through this approach, Zigbee may expand its coverage. However, it also has high battery consumption. This is because it is continually ready to transmit messages between devices at any time, similar to D2D mode. On the other hand, 2G is a technology that relies on a star topology. In star topology, energy consumption is not a limiting factor. Therefore, most of the intelligence and complexity tasks are executed at the base station. LPWA technology as NB-IoT, further decreases energy usage for longer battery life. This is done by simplifying the signaling protocol and lowering communication overhead to a minimum level.

Figure 8 demonstrates the above explanation of the main key factors of IoT device connectivity for system performance evaluation measures. These keys are presented by the three triangle sides which are: high data rate, low energy, and the wide area. Various ways of communication are represented inside the triangle which are: short-range, LPWA and finally cellular communication. As shown in Figure 8, the attributes of IoT services are closely coupled with the connectivity technology. For example, to achieve wide coverage, we should target LPWA (e.g. LORA) or cellular communication. However, if we target high speed applications, LPWA is generally not suitable due to its modest data rates. Furthermore, for applications where energy is a scarce resource, LPWA and short range communication should be targeted.

In addition, Mobility, positioning, latency, and scalability are other technical features that affect system applications.

In summary, we argue that all the above-mentioned IoT factors affect the connectivity modes of operation depending on the application. For SMs, for example, there is a need for high energy efficiency and data rates. In order to maintain the privacy, they require a dynamic change of transaction key. This could cause an accepted tolerance delay in SM. However, for the case of a vehicle application, there is a higher need for quick responses, mobility, and positioning than for energy efficiency.

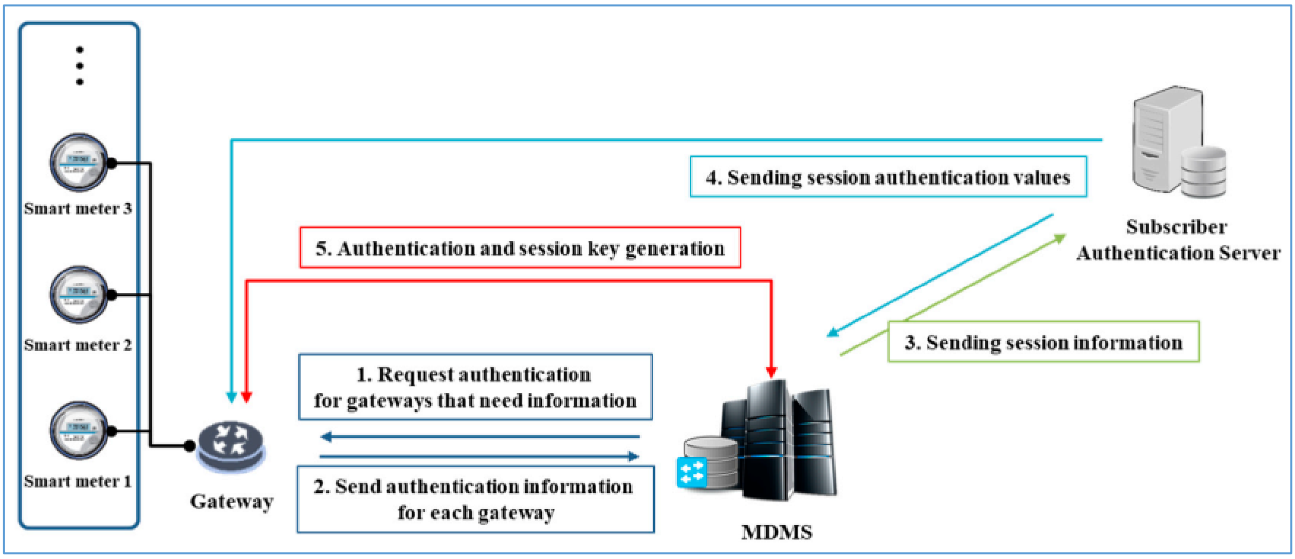

Smart meter 3
Smart meter 2
Smart meter 1
Gateway
1. Request authentication for gateways that need information
2. Send authentication information for each gateway
MDMS
3. Sending session information
Subscriber Authentication Server
4. Sending session authentication values
5. Authentication and session key generation

Figure 7. SM GW authentication phase with MDMS [32]

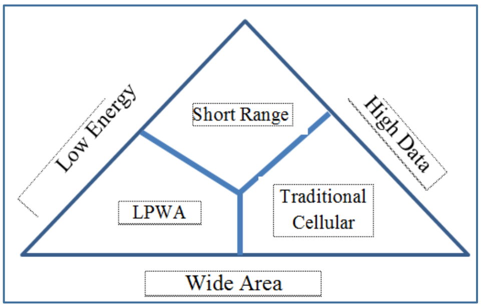


Figure 8. IoT connectivity key factors

3.6 Mapping between modes of operation & security elements

This section illustrates the link between IoT modes of operation and IoT security elements, as shown in Table 2. Regarding security elements, we address key generation and key distribution, which are considered the main building blocks for any security system. We place the articles surveyed in Section 2 in their proper location in the table for reaching the optimum security solution for each mode of operation, as shown in Table 2.

This table clearly shows that there are only a few security solutions proposed for the centralized mode of operation. The centralized mode of operation is the standard mode for non-IoT applications and has well-established schemes in the literature. However, for IoT applications with a high number of devices, such security solutions cause a high latency. In addition, the centralized mode represents a single point of failure at the central server. Most security solutions are proposed for D2D and GW modes of operation since they follow the majority of IoT solution architecture. We also note that the collaborative mode of operation is a promising candidate for future IoT systems since it can be connected to mobile devices, which leads to enhanced computational efficiency with low communication overhead.

Table 2. Matrix map between IoT modes of operations with security algorithms

		IoT Modes of Operation			
IoT Security Elements		Centralized	D2D	GW	Collaborative
	Key Generation		[13, 17, 36]	[10, 12, 14-19]	[24]
	Key Distribution	[28, 33]	[23, 27, 31, 41]	[16, 26, 28-30, 32, 37]	[24, 25, 28]

4. CONCLUSIONS AND FUTURE WORK

Security is considered the main development measuring tool for the IoT, as it is one of the major IoT challenges. The reliability and safety of IoT products depend on robust end-to-end approaches that protect consumers and their data. The main paper contributions are as follows. First, we classify the IoT connectivity mode of operations into four main modes of operations: GW, D2D, collaborative, and centralized. Second, we survey various IoT key management security articles focused on both key generation and distribution. Third, we map these articles in a map table. This mapping table consists of two dimensions: IoT modes of operation and security elements. This helps designers select the optimal security technique that provides the best balance between the required security level and each IoT mode's constraints. To our knowledge, no previous papers have discussed the aforementioned classification of IoT modes of operation. We believe that this effort will help to standardise the selection of key management techniques according to the limitations associated with each mode.

This survey should be followed by a quantitative assessment of key management techniques to choose the most suitable technique for each mode according to its limitations. For example, hard coded keys could be used in gateway mode, while it is not suitable for collaborative mode. Another example is the communication with key management center, which is feasible in the centralized and gateway modes but not the D2D and collaborative modes.

Although the collaborative mode is a highly promising IoT architecture, it has not been adequately covered in research. Therefore, studying and proposing new key management techniques that meet the challenges and limitations of this mode is a promising topic for future research.

REFERENCES

[1] Kumar, J.S., Patel, D.R. (2014). A survey on internet of things: Security and privacy issues. International Journal of Computer Applications, 90(11): 20-26. https://doi.org/10.5120/15764-4454

[2] Madakam, S., Ramaswamy, R., Tripathi, S. (2015). Internet of Things (IoT): A literature review. Journal of Computer and Communications, 3(5): 56616. http://dx.doi.org/10.4236/jcc.2015.35021

[3] Song, B., Kim, K. (2000). Comparison of existing key establishment protocols. KIISC (CISC 2000), 10(1): 677-690.

[4] Chris, R. (2017). The security implications of the Internet of Things. Journal of Cybersecurity Research, 2(1): 1-4. https://doi.org/10.19030/jcr.v2i1.9931

[5] Ali, I., Sabir, S., Ullah, Z. (2019). Internet of things security, device authentication and access control: A review. arXiv preprint arXiv:1901.07309.

[6] Chan, H., Perrig, A., Song, D. (2004). Key distribution techniques for sensor networks. Wireless Sensor Networks, pp. 277-303. https://doi.org/10.1007/978-1-4020-7884-2_13

[7] Manikandan, G., Sakthi, U. (2018). A comprehensive survey on various key management schemes in WSN. 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on, Palladam, India, pp. 378-383. https://doi.org/10.1109/I-SMAC.2018.8653656

[8] Chandramouli, R., Iorga, M., Chokhani, S. (2014). Cryptographic key management issues and challenges in cloud services. In Secure Cloud Computing, pp. 1-30. https://doi.org/10.1007/978-1-4614-9278-8_1

[9] Herbadji, A., Herbadji, D., Labiad, A. (2020). Information gathering and controlling over the internet by internet of things (IoT). International Journal of Safety and Security Engineering, 7(3): 49-54. https://doi.org/10.18280/rces.070301

[10] Margelis, G., Fafoutis, X., Oikonomou, G., Piechocki, R., Tryfonas, T., Thomas, P. (2019). Efficient DCT-based secret key generation for the Internet of Things. Ad Hoc Networks, 92: 101744. https://doi.org/10.1016/j.adhoc.2018.08.014

[11] Christof, P., Jan, P. (2010). Understanding Cryptography. Springer-Verlag, Berlin Heidelberg, 331-358. https://doi.org/10.1007/978-3-642-04101-3

[12] Furtak, J. (2020). Cryptographic keys generating and renewing system for IoT network nodes-A concept. Sensors, 20(17): 5012. https://doi.org/10.3390/s20175012

[13] Tsai, W.C., Tsai, T.H., Xiao, G.H., Wang, T.J., Lian, Y.R., Huang, S.H. (2020). An automatic key-update mechanism for M2M communication and IoT security enhancement. 2020 IEEE International Conference on Smart Internet of Things (SmartIoT), pp. 354-355. https://doi.org/10.1109/SmartIoT49966.2020.00067

[14] Malche, T., Maheshwary, P., Kumar, R. (2020). Secret key based sensor node security in the internet of things (IoT). 2020 5th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, pp. 464-469. https://doi.org/10.1109/ICCES48766.2020.9138078

[15] Wazid, M., Das, A.K., Shetty, S., JPC Rodrigues, J., Park, Y. (2019). LDAKM-EIoT: Lightweight device authentication and key management mechanism for edge-based IoT deployment. Sensors, 19(24): 5539. https://doi.org/10.3390/s19245539

[16] Eldefrawy, M.H., Pereira, N., Gidlund, M. (2018). Key distribution protocol for industrial Internet of Things without implicit certificates. IEEE Internet of Things Journal, 6(1): 906-917. https://doi.org/10.1109/JIOT.2018.2

[17] Thirumalai, C., Kar, H. (2017). Memory efficient multi key (memk) generation scheme for secure transportation of sensitive data over cloud and IoT devices. In 2017 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, pp. 1-6. https://doi.org/10.1109/IPACT.2017.8244948

[18] Chugunkov, I.V., Novikova, O.Y., Perevozchikov, V.A., Troitskiy, S.S. (2016). The development and researching of lightweight pseudorandom number generators. In 2016 IEEE NW Russia Young Researchers in Electrical and Electronic Engineering Conference (EIConRusNW), St. Petersburg, Russia, pp. 185-189. https://doi.org/10.1109/EIConRusNW.2016.7448150

[19] Liu, D., Liu, Z., Li, L., Zou, X. (2016). A low-cost low-power ring oscillator-based truly random number generator for encryption on smart cards. IEEE Transactions on Circuits and Systems II: Express Briefs, 63(6): 608-612. https://doi.org/10.1109/TCSII.2016.2530800

[20] Alohali, B.A., Kifayat, K., Shi, Q., Hurst, W. (2015). A survey on cryptography key management schemes for smart grid. Journal of Computer Sciences and Applications, Science and Education, 3(3A): 27-39. http://dx.doi.org/10.12691/jcsa-3-3A-4

[21] Saxena, N., Choi, B.J., Lu, R. (2015). Authentication and authorization scheme for various user roles and devices in smart grid. IEEE Transactions on Information Forensics and Security, 11(5): 907-921. https://doi.org/10.1109/TIFS.2015.2512525

[22] Uludag, S., Lui, K.S., Ren, W., Nahrstedt, K. (2015). Secure and scalable data collection with time minimization in the smart grid. IEEE Transactions on Smart Grid, 7(1): 43-54. https://doi.org/10.1109/TSG.2015.2404534

[23] ISO/IEC 11770-1. (1996). Information technology - security techniques - key management - Key Management-Part1: Framework.

[24] Sun, F., Zang, W., Huang, H., Farkhatdinov, I., Li, Y. (2020). Accelerometer-based key generation and distribution method for wearable IoT devices. IEEE Internet of Things Journal, 8(3): 1636-1650. https://doi.org/10.1109/JIOT.2020.3014646

[25] Concone, F., Re, G.L., Morana, M. (2020). SMCP: A Secure Mobile Crowdsensing Protocol for fog-based applications. Human-Centric Computing and Information Sciences, 10(1): 1-23. https://doi.org/10.1186/s13673-020-00232-y

[26] Dammak, M., Senouci, S.M., Messous, M.A., Elhdhili, M.H., Gransart, C. (2020). Decentralized lightweight group key management for dynamic access control in IoT

environments. IEEE Transactions on Network and Service Management, 17(3): 1742-1757. https://doi.org/10.1109/TNSM.2020.3002957

[27] Morshed Aski, A., Haj Seyyed Javadi, H., Shirdel, G.H. (2020). A full connectable and high scalable key pre-distribution scheme based on combinatorial designs for resource-constrained devices in IoT network. Wireless Personal Communications, 114: 2079-2103. https://doi.org/10.1007/s11277-020-07466-0

[28] Shin, D., Yun, K., Kim, J., Astillo, P.V., Kim, J.N., You, I. (2019). A security protocol for route optimization in DMM-based smart home IoT networks. IEEE Access, 7: 142531-142550. https://doi.org/10.1109/ACCESS.2019.2943929

[29] Rabiah, A.B., Ramakrishnan, K.K., Liri, E., Kar, K. (2018). A lightweight authentication and key exchange protocol for IoT. Workshop on Decentralized IoT Security and Standards (DISS), San Diego, CA, USA. https://dx.doi.org/10.14722/diss.2018.23004

[30] Guo, H., Zheng, Y., Li, X., Li, Z., Xia, C. (2018). Self-healing group key distribution protocol in wireless sensor networks for secure IoT communications. Future Generation Computer Systems, 89: 713-721. https://doi.org/10.1016/j.future.2018.07.009

[31] Leshem, G., David, E., Domb, M. (2018). Probability based keys sharing for IoT security. 2018 IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE), Eilat, Israel, pp. 1-5. https://doi.org/10.1109/ICSEE.2018.8645999

[32] Moharana, S.R., Jha, V.K., Satpathy, A., Addya, S.K., Turuk, A.K., Majhi, B. (2017). Secure key-distribution in IoT cloud networks. 2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS), Chennai, India, pp. 197-202. https://doi.org/10.1109/SSPS.2017.8071591

[33] Granjal, J., Monteiro, E., Silva, J.S. (2015). Security for the internet of things: A survey of existing protocols and open research issues. IEEE Communications Surveys & Tutorials, 17(3): 1294-1312. https://doi.org/10.1109/COMST.2015.2388550

[34] Sethi, P., Sarangi, S.R. (2017). Internet of things: architectures, protocols, and applications. Journal of Electrical and Computer Engineering, 2017: 9324035. https://doi.org/10.1155/2017/9324035

[35] Prasad, R., Rohokale, V. (2020). Internet of Things (IoT) and machine to machine (M2M) communication. In Cyber Security: The Lifeline of Information and Communication Technology, pp. 125-141. https://doi.org/10.1007/978-3-030-31703-4_9

[36] Shen, W., Hong, W., Cao, X., Yin, B., Shila, D.M., Cheng, Y. (2014). Secure key establishment for device-to-device communications. 2014 IEEE Global Communications Conference, Austin, TX, USA, pp. 336-340. https://doi.org/10.1109/GLOCOM.2014.7036830

[37] Lee, D.H., Lee, I.Y. (2018). Dynamic group authentication and key exchange scheme based on threshold secret sharing for IoT smart metering environments. Sensors, 18(10): 3534. https://doi.org/10.3390/s18103534

[38] Belmahdi, R., Mechta, D., Harous, S. (2021). A survey on various methods and algorithms of scheduling in fog computing. Ingénierie des Systèmes d'Information, 26(2): 211-224. https://doi.org/10.18280/isi.260208.

[39] McGinthy, J., Michaels, A. (2018). Session key derivation for low power IoT devices. 2018 IEEE 4th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing, (HPSC) and IEEE International Conference on Intelligent Data and Security (IDS), pp. 194-203. https://doi.org/10.1109/BDS/HPSC/IDS18.2018.00050

[40] Thirumalai, C., Shanmugam, S. (2017). Multi key distribution scheme by diophantine form for secure IoT communications. 2017 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, pp. 1-5. https://doi.org/10.1109/IPACT.2017.8245059

[41] Diaz-Sanchez, D., Marín-Lopez, A., Mendoza, F.A., Cabarcos, P.A., Sherratt, R.S. (2019). TLS/PKI challenges and certificate pinning techniques for IoT and M2M secure communications. IEEE Communications Surveys & Tutorials, 21(4): 3502-3531. https://doi.org/10.1109/COMST.2019.2914453