Lightweight Cryptography Algorithms for Internet of Things enabled Networks: An Overview

L.Mary Shamala¹, Dr.G.Zayaraz², Dr.K.Vivekanandan³, Dr.V.Vijayalakshmi⁴

¹Research Scholar, Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India.
²Professor, Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India.
³Professor, Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India.
⁴Associate Professor, Department of Electronics and Communication Engineering, Pondicherry Engineering College, Puducherry, India.

*mary@pec.edu, gzayaraz@pec.edu, k.vivekanandan@pec.edu, vvijizai@pec.edu

Abstract. Internet of Things is a worldwide set-up of interconnected entities that permits millions of devices to communicate with each other. Combined with reliable communication, ensuring security concerning confidentiality, integrity, and authenticity is a great challenge in IoT. Unsecured IoT devices open gateway for attacks. Unprotected and vulnerable devices, at times, allow easy entry for hackers, enabling them to have access to the shared network and personal, corporate assets. Conventional security measures are not suitable and cannot be applied to IoT technologies because of their minimum storage, low processing capacity, and limited computing power. Besides, scalability and heterogeneity issues arise when a variety of devices are interconnected. This paper presents the security threats and requirements of IoT cryptography, technology, and trends. The paper also discusses the challenges faced and the comparison of solutions already existing in IoT security.

1. Introduction

The Internet of Things(IoT) is a notable term in the evolution of the Internet that builds a network of small objects that can connect millions and millions of devices from various platforms via IPV6(6LoWPAN) protocol[1]. The two most essential exponents, RFID (Radio Frequency and Identification) and WSNs (Wireless Sensor Networks.) in IoT technology, have found a profound usage in several applications such as traffic control and environmental surveillance, home automation, and many more. CISCO’s current research estimates that by the year 2020, nearly 50 billion devices will be connected to the Internet [2]. It is certain that in the coming years’ technology will be used to connect devices rather than connecting people. This implies that every person will be surrounded by actuators, sensors, smart objects, RFID tags that change our interaction with the physical world around us. This evolution brings hard-hitting challenges in the security and privacy of IoT technologies.

Practically IoT devices have a limited number of gates for security, minimum storage, processing capacity, and limited power consumption. The design aim of any lightweight cipher is to increase security with reduced cost in terms of hardware, processing power, and computation. Conventional cryptography methods are not suitable for these resource-constrained devices to transmit data securely. One such advanced method is "Lightweight cryptography (LWC)," highly used for Internet Security Protocols to provide adequate security. Lightweight symmetric cryptographic algorithms are generally classified into Stream, Block, Hash, and Authenticated ciphers. Practically, block ciphers are more efficient and comfortable to implement and are considered workhorses than other ciphers. Block ciphers are mainly designed to deliver security, integrity, and authenticity. The lightweight block cipher design is the emerging, widely discussed topic among the researchers, academicians, and consultants. In this paper, the hardware and software implementation of recent lightweight block ciphers compared to other ciphers in the cryptographic field are summarized. The objectives of this paper are:

- Provide an overview of IoT and lightweight cryptography.
• Emphasize some of the security attacks that may occur on each layer of IoT layered architecture along with their causes.
• Present the design trends and security requirements of IoT cryptography.
• Provide a comparison of lightweight symmetric ciphers used to protect IoT.

The paper is systematically organized with Section 2, giving a summary of definitions, trends, and IoT elements. Section 3 discusses the hardware and software design challenges in IoT cryptography. Section 4 presents the lightweight cryptographic algorithms, its requirements, and design considerations. An overview of lightweight cryptography and ultra-lightweight cryptography is given in section 5. Section 6 discusses various symmetric block ciphers. Some of the future aspects of security and its challenges in IoT are presented in section 7. Section 8 concludes the paper.

2. Definitions, Trends and Elements
Kevin Ashton first framed the term Internet of Things in 1999 concerning the supply chain management [3]. IoT consists of global networking that unites uniquely identifiable physical objects with sensing, actuating, and communicating capabilities. The RFID group had defined the Internet of Things as “a world-wide network of interconnected objects which are uniquely addressed, based on standard communication protocols” [4]. The ITU explains IoT as, “having a connectivity for everyone connecting anything at any time, from any place” [5].

According to Cluster of European research projects [6], Internet of Things is defined as, “Things are active participants in business, information and social processes where they are enabled to interact and communicate data and information sensed about the environment, while reacting autonomously to the real/physical world events and influencing it by running processes that trigger actions and create services with or without direct human intervention”.

Forrester [7] explains IoT as “a smart environment that uses information and communications technologies to make the critical infrastructure components and services of a city’s administration, education, healthcare, public safety, real estate, transportation and utilities more aware, interactive and efficient.”

Atzori et al. [1] specified that the Internet of Things can be observed in three patterns:

i. Things-oriented (sensors) mainly focus on objects and identifying the standard way to integrate them.
ii. Internet-oriented (middleware), importance is given to standard networking and IP control exploiting to establish a well-organized connection between devices. It also simplifies the IP, making it possible to use on devices having limited capacity.
iii. Semantic-oriented (Knowledge) aims to harness semantic technologies, object description, and data management for the enhanced representation, storing and managing the large amount of data.

The past decade has witnessed a steady development in the Internet of Things (IoT) branch. Currently, there are 14.2 billion interconnected devices in use. Gartner, Inc. forecasts that by 2021 this number is further expected to hit 25 billion devices. Fig.1 illustrates the worldwide number of interconnected devices from 2015 to 2025.
The three IoT components [7] that enable consistent ubiquitous computing are:

i. Hardware - contains sensors, actuators, and embedded communication hardware.
ii. Middleware - contains on-demand storage and computing tools for data analytics.
iii. Presentation tools - provides easy understanding, visualization, and interpretations. These tools can be widely accessed on different platforms and can be designed for different applications.

3. Design Challenges in Lightweight Cryptography

Despite widespread research in both programming and equipment used for hardware and software implementation, the result shows that the correlation between them is cumbersome because of contrast in the executing platform [8].

3.1 Hardware implementation

The memory consumption, code size, and energy consumption are the primary metrics, considering that the hardware is earliest. The exact type of circuit only gives an exact measure of any lightweight cryptography. However, the simulation results are not the same with different tools. Hence, there is no standard comparison regarding the hardware implementation of different algorithms. Impending on to memory consumption, smaller blocks using smaller keys are preferable for lightweight primitives. However, “burning” keys into devices use read-only structures than read/write memory. It enables the key schedule to use only simple operations using the master keys. Energy efficiency is at the core of hardware implementation, along with latency, the time to do a given operation is also taken as one of the criteria in designing lightweight block ciphers.

3.2 Software implementation

Lightweight cryptography has implemented primitives like the RAM consumption, code size, and the throughput in bytes per cycle, which are essential metrics. The FELICS framework evaluates the performance of different lightweight algorithms using diverse metrics across different implementations. The FELICS results are shown in Table 1 for three micro-controllers: 8-bit AVR, 6-bit MSP, and 32-bit ARM-based on the 128-bit block in counter mode. Input as the block or stream ciphers and the output is the code size (bytes), RAM (bytes), and time taken (CPU cycles). The output extracted is made concise into one unit called Figure of Merit (FoM) that is used to rank block ciphers (the lower, the better).
4. Lightweight Cryptography Algorithms

Lightweight Cryptography (LWC) is a cryptography field that focuses on fast development and efficient cryptographic techniques for resource-constrained environments that can replace the traditional computationally expensive ones while achieving an adequate level of security [10]. Lightweight solutions are designed to be lighter concerning their key size, memory requirements, and execution time. This facilitates lesser resources for used compared to heavyweight solutions. There are no constrained requirements for any lightweight algorithm to fit in; however, the key size, block size, code measures, clock cycles, and much more are given higher importance. The goal of creating a lightweight algorithm design is to compromise various factors like low resource requirements, performance, and cryptographic strength of the algorithm [10].

4.1 Requirements of LWC

Based on the design challenges, the lightweight algorithms are intended to use smaller block sizes (32, 48, or 64 bits) than a conventional cipher, which has a larger block size (64 or 128 bits). LWC also uses smaller key sizes (less than 96 bits). The least key size, according to NIST, is 112 bits [12]. In ISO/IEC 29192[12], lightweight properties are detailed established on their target platforms. Firstly, Lightweight properties of hardware are evaluated by essential measures such as chip size and energy consumption. Secondly, RAM size with smaller code is favoured for lightweight applications in software implementations.

4.2 Design considerations of LWC

The following design considerations perceived in designing a lightweight cryptographic solution:

- A decrease in the algorithm’s main variables like block size, key length, and the algorithm’s internal state can cause security problems such as CBC erodes and brute-force key attacks.
- Lightweight algorithms should be built upon elements that are widely used and thoroughly analysed.
- Transformations with simplified layers like decreasing the ROM requirements.
- Use of productive components like data-dependent bit following, shift registers, low cost, and many more.
- Designing simpler key schedules that can derive subkeys rapidly forward or backward.
- Applying basic operations with a more significant number of rounds.
- Using operations according to the resources available on the target platform. that allow implementation trade-offs.

Lightweight cryptography is needed for IoT due to the following reasons:

- Efficient end-to-end communication: End nodes must be equipped with a symmetric key algorithm for achieving end-to-end security. For resource-constrained IoT devices, it is crucial
to have a cryptographic operation with less resource consumption. Implementing a lightweight symmetric key algorithm on end devices consumes less energy [13][14].

- Relevancy to resource-constrained devices [15][16][17]: The lightweight cryptographic primitives occupy smaller space than conventional primitives. Hence, lightweight cryptographic primitives would open the likelihood of more network connections even with resource-constrained devices.

5. IoT Cryptography

The concept of lightweight cryptography can be broadly classified based on the various levels of security they demand: IoT cryptography and ultra-lightweight cryptography [2]. The former category is associated only with low resource devices that comprise of lower security level. The latter deals with low power processors and modern cryptography primitives being expensive. As they are having greater connectivity, they are expected to provide high-level security. “An ultra-lightweight cryptography algorithm is one running on very cheap devices that are not connected to the internet, that which are easily replaced if necessary and have a limited lifetime.” Such algorithms can be used in devices like RFID tags, smart cards, rain tags, memory encryption, remote car keys, etc. [16].

IoT devices are connected to the Internet. An 80–bit key is not advisable for protecting these devices and connecting them to a global platform. This situation gives rise to IoT cryptography. Though IoT devices are computationally weak compared to a higher-end device, they perform multiple tasks. Therefore, a versatile primitive is needed for IoT. IoT devices require encryption and function like authenticating user communications, authenticating manufacturer updates, etc. Whereas, Ultra-lightweight devices demand only a single cryptographic operation [2]

Security is also one of the main differences between these two. An improved network connection is a reason for getting better results. For example, an IoT enabled device with is unsecured can be used to perform a denial of service (DoS). Hence there is a demand for a much higher security level. It is necessary to have a minimum of 128 bits rather than 80 bits secret keys in such situations. "An IoT cryptographic algorithm is one running on a low-power device connected to a global network such as the Internet" [2][18]. As IoT devices are networked, there must be ideally one primitive for all IoT. Since an attacker may physically access some of these devices, Side Channel Attack (SCA) countermeasures must simpler to carry out.

IoT devices are enabled to perform various tasks, and hence multipurpose microcontrollers will be preferred to perform cryptographic operations than electronic circuits. Hence software efficiency is of prime importance to IoT cryptography. Algorithms such as Chaskey, RECTANGLE, Lea, SPECK, Sparx, etc. Table 2 specifies the design requirements of IoT cryptography primitives have been designed for software efficiency.

| Design Requirement | Description |
|--------------------|-------------|
| Type               | Block cipher or sponge. IoT devices perform multiple tasks; a versatile primitive is needed. |
| Block size         | 96 bits is the minimum, higher sizes (>128 bits) must be preferred. |
| Key Size           | At least 128 bits since smaller keys may give easy access to opponents |
| Relevant Attacks   | Same as traditional ciphers; therefore a much conservative security model must be used |
| Target Platform    | Microcontrollers and Low end processors |
| SCA resilience     | It is important to implement SCA countermeasures since IoT devices can be physically attacked by the opponents in some cases. |
| Functionalities    | Cryptographic operations such as encryption, hashing, authentication and so on |
| Flexibility        | The algorithm must be decently efficient on a wide range of microcontrollers |
6. IoT and Lightweight Cryptography
Block ciphers mostly replace academic designers to build a lightweight symmetric algorithm. Based on structural anatomy, symmetric block ciphers are classified as Substitution Permutation Network (SPN) and Feistel network (FN) [17]. An SPN takes in blocks of plaintext and keys and has the advantage of the easy and simple-to-analyze structure. FN possesses the features of identical encryption and decryption using only a reversal of key schedule. Different algorithms have been used to address the issues of lightweight design constraints. Some of these trends have come up with the rise of block ciphers [2]. These are highlighted explicitly in two parts: non-linear operations and key schedule.

6.1 Non-Linear Operations
The cryptography algorithm possesses an essential feature of non-linearity. This property is available by S-Boxes or by using non-linear arithmetic operations. S-Box-based algorithms are then classified as two subcategories. The first category uses Look-Up Tables (LUT), whereas the second category implements the bit-slice. Primitives following the ARX design consider only modular additions for arithmetic operations [19] [20]. S-Boxes are used in LUT algorithms; these are then performed with look-up tables [22]. Similarly, S-Boxes are also used by bit slice-based algorithms, but the difference is for computing the S-Box layer, table look-ups are not needed. To compute the S-Box parallely, bitwise options such as AND and XOR can be performed on words [22]. ARX-based ciphers are considered the finest cipher for microcontrollers by Felics framework [23].

6.2 Key Schedule
The key schedule of lightweight algorithms must be as simple as possible. The popular methods for constructing simple key schedules are described below:

(i) Even-Mansour and “Selecting” Key Schedules
Lightweight algorithms use round constants in each round and selects bits of master keys in each round. Such key schedule is a form of Even-Mansour construction [24]. Different chunks of the master key can also be used as subkey during encryption. This process is called selection of key schedule as forms a sub key using bits of master key. The main benefit of such a key schedule is that less logic is required in computing round keys. Moreover, key state does not require updating which is considered to be expensive in hardware.

(ii) Round Function Based
A straightforward method for a simple key schedule is reusing important parts of the round function for the process of updating the key state. The entire round function or only parts of it can be used in this case [25]. Few recent lightweight algorithms use a tweak along with the key [27]. Variability is provided through tweak. Tweak is a public variable that enables the user to use advanced modes of operation.

6.3 Lightweight ciphers – A Brief review
A wide variety of block ciphers are proposed and implemented on different platforms viz. AES [27] Advance Encryption standard - a block cipher implemented on AVR, GPU etc., PRESENT [29] an ultra-lightweight block cipher implemented using different architectures: Serialized, Pipelined and Round-based, LEA [29] Link Encryption Algorithm- a block cipher implemented on Verilog, Simon [30] a block cipher implemented on ASIC, SPECK [31] implemented on ILC 350-PN controller adopting Matsui’s algorithm, DES [33] Data encryption standard - a block cipher and its versions DESX, DESXL implemented on Synopsis Design vision with minimum gate equivalents, TEA [33] Tiny encryption algorithm – a block cipher based on ARX design is the best design to achieve energy efficient server, HIGHT [34] a block cipher has higher throughput and less energy consumption, ECC [35] a block cipher was implemented on a 8-bit microcontroller. The existing survey shows that
there are many more algorithms apart from the aforementioned and each algorithm is implemented on different platforms and compared. From the study it is prominently evident that there is no settled outcome from any algorithm. Very few algorithms are proficient in providing security to IoT devices.

6.4 Performance Comparison of Lightweight Algorithms
Lightweight block ciphers' success rate is mandatory to optimize the trade-offs between cost, performance, and security. The design aim of any lightweight cipher is to increase security with reduced cost in terms of hardware, processing power, and computation. The parameters listed are taken into account for measuring the performance of any cipher. A comparison of prominent lightweight symmetric block ciphers [23] that are primarily optimized for software implementation is presented in this section. Table 3 lists the existing software-oriented lightweight block ciphers for securing IoT devices and their FELICS results for enciphering 128 bits of the message using counter mode.

Table 3. FELICS Results for encrypting 128 bits of message

| Cipher Name    | Block Size (bits) | Key Size (bits) | Number of Rounds | Code Size (bytes) | RAM (bytes) | Execution Time (cycles/block) | Figure of Merit (FoM) |
|----------------|-------------------|-----------------|------------------|-------------------|-------------|--------------------------------|----------------------|
| Chaskey        | 128               | 128             | 8                | 200               | 84          | 563                            | 4.3                  |
| Chaskey-LTS    | 128               | 128             | 12/16            | 232               | 84          | 643                            | 4.9                  |
| SPECK          | 64                | 128             | 27               | 278               | 60          | 965                            | 5.1                  |
| SIMON          | 64                | 128             | 44               | 420               | 80          | 1422                           | 7.3                  |
| LEA            | 128               | 128             | 24               | 336               | 120         | 1146                           | 7.9                  |
| RECTANGLE      | 64                | 128             | 25               | 440               | 76          | 2400                           | 8.6                  |
| SPARX          | 128               | 128             | 32               | 1480              | 112         | 2917                           | 12.7                 |
| AES            | 128               | 128             | 10               | 1352              | 124         | 3969                           | 14.1                 |
| HIGHT          | 64                | 128             | 12               | 396               | 76          | 1868                           | 15.1                 |
| Fantomas       | 128               | 128             | 32               | 670               | 100         | 5666                           | 15.6                 |
| Robin          | 128               | 128             | 16               | 2036              | 158         | 6962                           | 22.6                 |

Lightweight block ciphers are compared using the primary drivers, namely code size, memory, and execution time apart from its speed and simplicity. The comparisons of these ciphers in terms of code size, RAM consumption, execution time are shown in Fig. 2, Fig.3 and Fig.4, respectively. These quantities are obtained for 32-bit ARM microcontroller.

Fig. 2. Comparison of code size

Code size is determined by a set of instructions of the target processor. Though some four or five ciphers have smaller code size, the faster verification and smaller code size of Chaskey makes it highly suitable to go on with other parameters.
The code size metric focuses on the registers used in RAM and ROM for encryption of block ciphers. The quality of lightweight cipher is directly proportional to the amount of data processed during encryption. It is affected by the number of shifts and XOR operations. From the graph, it is proven that Chaskey has the least memory consumption 583, the worst being 6962, and the Robin cipher.

![Figure 3. Comparison of RAM consumption](image)

The power consumption directly depends on the execution time and computing speed. The number of computations defines the lightness of the ciphers. It is calculated as the ratio of the number of cycles per block. Fig 4 shows that shorter execution time compared to other ciphers. Chaskey uses ARX structure and a strong diffusion technique to reduce the data complexity, thereby reducing time complexity. From the comparison, it is evident that when code size is exceptionally critical, Chaskey, RC5-20, SIMON, SPECK, and even RECTANGLE seem reasonable choices. However, combining all the metrics, it is evident that Chaskey is the best performing.

![Figure 4. Comparison of execution Time](image)

7. Security Challenges and Future trends of Lightweight Cryptography

Optimizing the trade-offs between the design goals such as security, cost, and performance is a difficult task for every designer. The resource constraints and compromises of LWC mentioned as forth paved the way for designing a highly secure system. (i) constraints in resources influence cryptographers to design lightweight algorithms with relatively small block sizes and key length. More specifically, this allows an opponent to attack the lightweight algorithm using the matching ciphertext attack [38]. (ii) Lightweight cryptographic primitives target systems with relatively low or medium security requirements, the specifications of the algorithm to be used with which the desirable compromise can be found. (iii) Lightweight algorithms are designed to be small to allow the designers to use simpler confusion and diffusion transformations. Hence proper security margins are not designed in the internal structures of lightweight algorithms (iv) There are chances of side-channel attacks against hardware realizations of some lightweight algorithms which are not at all examined [37].
Along with the aforementioned constraints for providing security, these essential security services should be essentially maintained in IoT technology [38] to strengthen the user's trust:

- **Confidentiality** ensures that message at draft or in transit can only be accessible to sender or receiver.
- **Integrity** ensures that original content of the message cannot be modified by any intruder.
- **Authentication** ensures the validity of message by having sender identity be verified to the receiver.
- **Authorization** ensures that only IoT authorized users can have access to resources and connectivity among others.
- **Privacy** ensures the hiding of personal data and access to control the activities of this data.

IoT is known to offer many economic benefits; however, it faces many challenges. A few challenges that need to be addressed while accommodating the trillion of IoT devices are:

- **Most of the IoT devices do not have basic security requirements.** The potential to have standard encryption processes and protocols is reduced by the wide variety of IoT devices and hardware profiles, creating a barrier to visibility in security incidents. It leads to insufficient transparency concerning privacy and security.
- **IoT network security poses more challenges than traditional network security due to increased communication protocols, standards, and device capabilities; these all raise issues and increase complexity creating security blind spots.**
- **Most IoT authentication scenarios are machine-to-machine based without any human intervention, which is unlikely to the enterprise networks where authentication processes involve a human being entering a credential.**
- **Currently, IoT employs AES for confidentiality and integrity and ECC for integrity, authentication, and key exchange. AES is considered to be a lightweight block cipher; however, its large design space and inherent exposure to SCA makes it the best choice.**
- **Examining the IoT architecture further discusses the security violations in different layers of IoT.** The three essential IoT architecture layers [39] are as follows:
  - **Information is collected into the perception layer.** This is then classified into the perception node and the perception network. Here, the Perception node functions data acquiring and controlling, whereas the perception network layer provides control instructions.
  - **The network layer performs transmission of a network and provides security to information.** It delivers a ubiquitous access environment to the perception layer.
  - **The application layer specifies all applications that use the IoT or in which IoT has deployed.**

Table 4 summarizes the security attacks at the three layers of IoT architecture [41], along with their causes of a security violation. This shows that IoT is significantly in jeopardy to attacks; thereby, providing security in the IoT environment becomes a challenge due to the following reasons [1]: (a) It is easy to attack IoT components physically as mostly they are left unattended. (b) Wireless communication is used by IoT, which is exposed to several attacks. (c) Any device can connect to the network at any time, which may cause unauthorized entry into the network. (d) Finally, most of the IoT components have limited resources in terms of bandwidth and energy. Implementing complex security solutions in those components may inhibit the efficient working of devices.
Table 4. Summary of security attacks at IoT layers

| Target layer | Attack                      | Effect and vulnerability reason                                                                 | Security violation         |
|--------------|-----------------------------|---------------------------------------------------------------------------------------------------|----------------------------|
| Perception Layer | Eavesdropping             | It tries to steal information that is transmitted over a network. Takes advantage of unsecured transmission | Confidentiality             |
|              | Node Capture                | Gains full control over a key node, such as a gateway node                                       | Confidentiality, Authentication, Integrity |
|              | Fake Node and Malicious    | Stop transmitting real information                                                               | Availability, Integrity    |
|              | Replay Attack               | Intruder takes authentic information from the sender                                             | Authentication              |
|              | Timing Attack               | Weak computing capabilities of devices are used to extract secrets                                | Confidentiality             |
| Network Layer | Denial of Service (DoS) Attack | Prevent authentic users from accessing devices or other network resources                         | Availability, Authorization |
|              | Man-in-the-Middle (MtM) Attack | Intercept and alter the communication                                                              | Confidentiality, Integrity, Authentication |
|              | Storage Attack              | User’s information may be changed to incorrect details due to replication of information at storage | Integrity                  |
|              | Exploit Attack              | Gains control of the system and steals information stored on a network. Occurs due to security vulnerabilities in an application, system or hardware | Confidentiality             |
| Application layer | Cross Site Scripting       | Attacker changes the contents of the application                                                  | Integrity                  |
|              | Malicious Code Attack       | Cause undesired effects and damage to the system                                                  | Confidentiality, Integrity, Availability |
|              | The ability of dealing with  | Leads to network disturbance and data loss                                                         | Availability                |
|              | Mass Data                   |                                                                                                   |                            |

Security approaches: a review
The security level of any system is measured based on the diffusion level (the inversion of input to output), confusion level (ensuring the relationship between key, plain-text, and ciphertext), and linear and differential cryptanalysis (estimates the complexity of attacks). M. El-Haii et. al [40] examined the different lightweight ciphers using Raspberry Pi and Arduino. The result states that every algorithm is superior in its way such as Speck for Memory consumption, Rectangle for Speed performance, Present for security.

Yao et. al [42] Security challenges in IoT are addressed by the elliptic curve decisional Diffie hellman (ECDDH) problem to reduce communication cost and improve execution efficiency. Yang et al. [41] presented a discrete, secure information administration for health monitoring systems using the keyword retrieval method. Since various medical institutions monitor the patients, it is necessary to have a distributed access between health institutions making the electronic health records viable.

Singh et. al[43] presented a combined lightweight algorithm that assembles symmetry and asymmetry algorithms for a better space environment. The four variables, namely data size, battery power, memory space, and computation power, are checked for their threshold value, and a proper, either symmetric or asymmetric lightweight encryption is used. Still, it requires a clear explanation about the cipher structure, key size, block size, and security metrics.

Al salami et.al[44] al suggested a lightweight encryption algorithm for smart homes that provides confidentiality with a favorable level of efficiency and reduces overhead cost. Ernest[45] addressed the IoT constraints such as size, speed, simplicity, code size, execution time, and memory metrics with lightweight primitives and new technology. Biryukov et. al [46] carried out an intensive survey of lightweight cryptographic primitives and listed the pros and cons of IoT. Further, he discusses the need for specific algorithms for ultra-lightweight cryptography and IoT cryptography.

Research had tried to address many issues of IoT security; some difficulties in providing security still exist. Bringing out an efficient security system with minimum computation time and less power consumption in resource-constrained devices is still a challenge as there is always a tradeoff between
cost, security, and performance. The fast-growing digital technological device usage in society and protecting these devices’ connectivity to a global network also adds up as a challenge to this issue. Moreover, providing a lightweight parameter to a cipher is difficult because every algorithmic design tries to find a solution to a different set of requirements under one roof. Hence there is always a need for proper lightweight security and privacy mechanisms by which these issues can be solved.

8. Conclusion
Several limitations of IoT devices are discussed in this paper: reduced computation power, reduced communication capabilities, high latency, and reduced input energy. The objective of this paper is to put forth a clear vision of providing security to the network. The performance comparison of existing lightweight algorithms and some of the research challenges in the IoT environment that are to be focused on future research are discussed. The recently used encryption and decryption algorithms, otherwise called ciphers, are not applicable for all the applications due to its complexity. The design requirements and different elements, issues, and trends in IoT security articulated in this paper give a valued solid idea in designing a lightweight block cipher. Hence, to use IoT with all its technological advantages, an acceptable security system is needed, which takes the research innovative.

References
[1] Atzori, L., Iera, A. and Morabito, G., 2010. The internet of things: A survey. Computer networks, 54(15), pp.2787-2805.
[2] Evans, D., 2011. The internet of things: How the next evolution of the internet is changing everything. CISCO. Int. J. Internet, 3(2), pp.123-132.
[3] Ashton, K., 2009. That ‘internet of things’ thing. RFID journal, 22(7), pp.97-114.
[4] Jia, X., Feng, Q., Fan, T. and Lei, Q., 2012, April. RFID technology and its applications in Internet of Things (IoT). In 2012 2nd international conference on consumer electronics, communications and networks (CECNet) (pp. 1282-1285). IEEE.
[5] Borgia, E., 2014. The Internet of Things vision: Key features, applications and open issues. Computer Communications, 54, pp.1-31.
[6] Sundmaeker, H., Guillemin, P., Friess, P. and Woelfflé, S., 2010. Vision and challenges for realising the Internet of Things. Cluster of European Research Projects on the Internet of Things, European Commission, 3(3), pp.34-36.
[7] Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M., 2013. Internet of Things (IoT): A vision, architectural elements, and future directions. Future generation computer systems, 29(7), pp.1645-1660.
[8] Diehl, W., Farahmand, F., Yalla, P., Kaps, J.P. and Gaj, K., 2017, September. Comparison of hardware and software implementations of selected lightweight block ciphers. In 2017 27th International Conference on Field Programmable Logic and Applications (FPL) (pp. 1-4). IEEE.
[9] Katagi, M. and Moriai, S., 2008. Lightweight cryptography for the internet of things. Sony Corporation, 2008, pp.7-10.
[10] Mouha, N., 2015, July. The design space of lightweight cryptography.
[11] McKay, K., Bassham, L., Sönmez Turan, M. and Mouha, N., 2016. Report on lightweight cryptography (No. NIST Internal or Interagency Report (NISTIR) 8114 (Draft)). National Institute of Standards and Technology.
[12] ISO/IEC 29192-2:2012, Information technology — Security techniques — Lightweight cryptography — Part 2: Block ciphers. URL: https://www.iso.org/obp/ui#iso:std:iso-iec:29192-2:ed-2:v1:en
[13] Li, F., Zheng, Z. and Jin, C., 2016. Secure and efficient data transmission in the Internet of Things. Telecommunication Systems, 62(1), pp.111-122.
[14] Jing, Q., Vasilakos, A.V., Wan, J., Lu, J. and Qiu, D., 2014. Security of the Internet of Things: perspectives and challenges. Wireless Networks, 20(8), pp.2481-2501.
[15] Batra, I., Luthach, A.K. and Pathak, N., 2016, March. Research and analysis of lightweight cryptographic solutions for internet of things. In Proceedings of the second international
conference on information and communication technology for competitive strategies (pp. 1-5).

[16] Panasenko, S. and Smagin, S., 2011. Lightweight cryptography: Underlying principles and approaches. International Journal of Computer Theory and Engineering, 3(4), p.516.

[17] Kong, J.H., Ang, L.M. and Seng, K.P., 2015. A comprehensive survey of modern symmetric cryptographic solutions for resource constrained environments. Journal of Network and Computer Applications, 49, pp.15-50.

[18] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J., Seurin, Y. and Vikkelsoe, C., 2007, September. PRESENT: An ultra-lightweight block cipher. In International workshop on cryptographic hardware and embedded systems (pp. 450-466). Springer, Berlin, Heidelberg.

[19] Bossuet, L., Grand, M., Gaspar, L., Fischer, V. and Gogniat, G., 2013. Architectures of flexible symmetric key crypto engines—a survey: From hardware coprocessor to multi-crypto-processor system on chip. ACM Computing Surveys (CSUR), 45(4), pp.1-32.

[20] Dinu, D., Perrin, L., Udovenko, A., Velichkov, V., Großschädl, J. and Biryukov, A., 2016. Design Strategies for ARX with Provable Bounds: SPARX and LAX (Full Version). IACR Cryptol. ePrint Arch., 2016, p.984.

[21] Kumar, A. and Tejani, S., 2018, February. S-BOX Architecture. In International Conference on Futuristic Trends in Network and Communication Technologies (pp. 17-27). Springer, Singapore.

[22] Rebeiro, C., Selvakumar, D. and Devi, A.S.L., 2006, December. Bitslice implementation of AES. In International Conference on Cryptology and Network Security (pp. 203-212). Springer, Berlin, Heidelberg.

[23] Dinu, D., Biryukov, A., Großschädl, J., Khovratovich, D., Le Corre, Y. and Perrin, L., 2015, July. Felic–fair evaluation of lightweight cryptographic systems. In NIST Workshop on Lightweight Cryptography (Vol. 128).

[24] Even, S. and Mansour, Y., 1997. A construction of a cipher from a single pseudorandom permutation. Journal of cryptology, 10(3), pp.151-161.

[25] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B. and Wingers, L., 2013. The SIMON and SPECK Families of Lightweight Block Ciphers. IACR Cryptol. ePrint Arch., 2013, p.404.

[26] Liskov, M., Rivest, R.L. and Wagner, D., 2002, August. Tweakable block ciphers. In Annual International Cryptology Conference (pp. 31-46). Springer, Berlin, Heidelberg.

[27] Bos, J.W., Osvik, D.A. and Stefan, D., 2009. Fast Implementations of AES on Various Platforms. IACR Cryptol. ePrint Arch., 2009, p.501.

[28] Rolfos, C., Poschmann, A., Leander, G. and Paar, C., 2008, September. Ultra-lightweight implementations for smart devices–security for 1000 gate equivalents. In International Conference on Smart Card Research and Advanced Applications (pp. 89-103). Springer, Berlin, Heidelberg.

[29] Lee, D., Kim, D.C., Kwon, D. and Kim, H., 2014. Efficient hardware implementation of the lightweight block encryption algorithm LEA. Sensors, 14(1), pp.975-994.

[30] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B. and Wingers, L., 2015. SIMON and SPECK: Block Ciphers for the Internet of Things. IACR Cryptol. ePrint Arch., 2015, p.585.

[31] Duka, A.V. and Genge, B., 2017, April. Implementation of SIMON and SPECK lightweight block ciphers on programmable logic controllers. In 2017 5th International Symposium on Digital Forensic and Security (ISDFS) (pp. 1-6). IEEE.

[32] Leander, G., Paar, C., Poschmann, A. and Schramm, K., 2007, March. New lightweight DES variants. In International Workshop on Fast Software Encryption (pp. 196-210). Springer, Berlin, Heidelberg.

[33] Kataoka, H., Sawada, A., Duolikun, D., Enokido, T. and Takizawa, M., 2016, March. Energy-aware server selection algorithms in a scalable cluster. In 2016 IEEE 30th
International Conference on Advanced Information Networking and Applications (AINA) (pp. 565-572). IEEE.

[34] Mohd, B.J., Hayajneh, T., Khalaf, Z.A. and Ahmad Yousef, K.M., 2016. Modeling and optimization of the lightweight HIGHT block cipher design with FPGA implementation. Security and Communication Networks, 9(13), pp.2200-2216.

[35] Eisenbarth, T., Kumar, S., Paar, C., Poschmann, A. and Ultsch, L., 2007. A survey of lightweight-cryptography implementations. IEEE Design & Test of Computers, 24(6), pp.522-533.

[36] Panasenko, S. and Smagin, S., 2011. Lightweight cryptography: Underlying principles and approaches. International Journal of Computer Theory and Engineering, 3(4), p.516.

[37] Kumar, S.A., Vealey, T. and Srivastava, H., 2016, January. Security in internet of things: Challenges, solutions and future directions. In 2016 49th Hawaii International Conference on System Sciences (HICSS) (pp. 5772-5781). IEEE.

[38] Alaba, F.A., Othman, M., Hashem, I.A.T. and Alotaibi, F., 2017. Internet of Things security: A survey. Journal of Network and Computer Applications, 88, pp.10-28.

[39] Ray, P.P., 2018. A survey on Internet of Things architectures. Journal of King Saud University-Computer and Information Sciences, 30(3), pp.291-319.

[40] M. El-Haii, M. Chamoun, A. Fadlallah and A. Serhrouchni, "Analysis of Cryptographic Algorithms on IoT Hardware platforms," 2018 2nd Cyber Security in Networking Conference (CSNet), Paris, 2018, pp. 1-5, doi: 10.1109/CSNET.2018.8602942.

[41] Yang, Y., Zheng, X. and Tang, C., 2017. Lightweight distributed secure data management system for health internet of things. Journal of Network and Computer Applications, 89, pp.26-37.

[42] Yao, X., Chen, Z. and Tian, Y., 2015. A lightweight attribute-based encryption scheme for the Internet of Things. Future Generation Computer Systems, 49, pp.104-112.

[43] Singh, S., Sharma, P.K., Moon, S.Y. and Park, J.H., 2017. Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions. Journal of Ambient Intelligence and Humanized Computing, pp.1-18.

[44] Al Salami, S., Baek, J., Salah, K. and Damiani, E., 2016, August. Lightweight encryption for smart home. In 2016 11th International Conference on Availability, Reliability and Security (ARES) (pp. 382-388). IEEE.

[45] Singh, S., Sharma, P.K., Moon, S.Y. and Park, J.H., 2017. Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions. Journal of Ambient Intelligence and Humanized Computing, pp.1-18.

[46] Biryukov, A. and Perrin, L.P., 2017. State of the art in lightweight symmetric cryptography.