Security Optimization of Resource-Constrained Internet of Healthcare Things (IoHT) Devices Using Lightweight Cryptography

V Jayaprakash, Amit Kumar Tyagi

ABSTRACT. The term "Internet of Things" is becoming increasingly popular and promising, ushering in a new era of smarter connectivity across billions of gadgets. In the foreseeable future, IoT's potential is boundless. The healthcare industry, often known as IoHT, is the most demanding application of IoT. Sensors, RFID, and smart tags are used to start any IoT system, but these applications lack the necessary resources such as power, memory, and speed. The key requirement is secure information transformation because it contains sensitive patient information that might be extremely dangerous if it falls into the hands of an unauthorized person. Encryption approaches that have been used in the past are ineffective. Lightweight cryptography is the most viable solution for protection of data at the physical layer. This paper emphasizes on some lightweight symmetric block ciphers that are widely applied in the health sector such as CLEFIA, KATAN and SIMON. The software performance such as key size, block size, number of rounds, execution time, encryption time, and memory occupation of these algorithms have all been well analyzed using Python across various platforms. Furthermore, we focus on the issues and need for IoHT device security, and we give solutions that can be used as a source of information for the health care industry to implement smart and secure procedures in order to improve patient happiness. In comparison to the other two algorithms, the results demonstrate that KATAN is more memory efficient and SIMON is the fastest.

KEY WORDS*: Security, IoHT, Sensors, RFID, Resource constraints, Lightweight symmetric block cipher, CLEFIA, KATAN, SIMON, Python, memory occupation, execution time, key size.*

1 INTRODUCTION

The Internet of Things has become the most widely used term in the world today. It is a technical concept that entails practical devices such as sensors and actuators that are used to collect real-time data, convey that data over the internet, and store that data on cloud-based platforms with or without human participation [1-3]. In 1999, Kevin Ashton coined the term "Internet of Things" to promote the usage of radio frequency-based identification (RFID), which involves a variety of embedded devices. With the advent of home automation, industrial energy meters, wearable and self-health care devices in 2011, the tremendous expansion of IoT-based devices began [4]. Health care is an important sector that is one of the major contributors to the total number of IoT enabled devices in the world. The invention of IoHT enables patients to self-assess their body conditions and also simultaneously upload these data to the hospital's server so that doctors can keep track of patients' health condition and call for checkups and visits only when required which ultimately helps in saving money as well as time [5-6]. However, the massive outbreak of this technology has led to many issues and challenges regarding the security of patient's data.

Data protection is required at three layers in any IoHT device: physical/design, communication, and computation. [7] They are further divided into resource-rich (phones, tablets, laptops) and resource-constrained (sensors, RFID) devices. Devices with limited resources are frequently utilized to handle real-time applications that demand precise data processing. Furthermore, they are constrained in terms of power consumption, memory, and processing rates [8-9]. The focus of this research is on the implementation of algorithms for device security in the latter group.

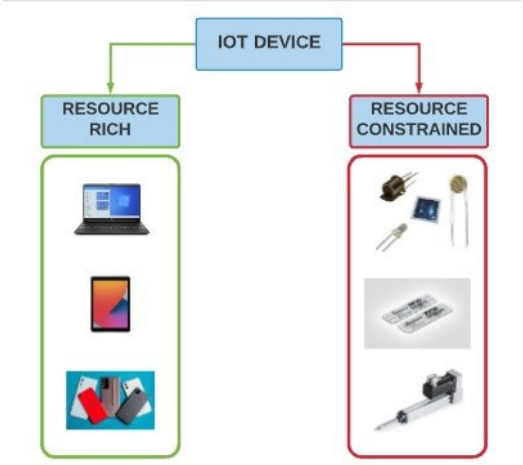


Fig 1: Categorization of IoT devices

In most of the countries, the authentic information provided by the healthcare data should be confined through “Health Information and Portability Accountability Association (HIPAA)” [10]. Efficient and safe implementation of these healthcare systems can be achieved by using optimized and robust security systems [11]. Cryptography is the widely applied technique to secure the data and prevent the leakage of information. An IoHT device begins at the implementation of physical layer using sensors, RFID tags, actuators etc. to acquire the information regarding patient’s health. Typical encryption algorithms like AES, DES, RSA cannot be applied to these embedded devices as they are more suitable for devices with high computation powers. Lightweight cryptographic (LWC) techniques are utilized in such fields.

As the term suggests they are capable of operating at lower power, smaller memory and better computation speeds [12]. The most commonly used LWC methods in the field of healthcare are PRESENT, CLEFIA, PICOLO, KATAN, SPECK and SIMON. These ciphers are most widely used in IoHT devices as they are more advantageous in both software as well as hardware application. The function of any cryptography is to convert any plain text to corresponding cipher text using secret keys, logical shifting operations and permutation levels and then convert the cipher text back into plain text. Block ciphers are given more importance in cryptographic fields as they are more efficient and versatile in choosing the information including the key size [13-14].

ORGANIZATION OF THE WORK:

The paper is organized as follows: Section II give a literature survey regarding related researches followed by motivation in Section III. Section IV discusses the difficulties of developing a secure IoT system, while Section V offers a solution based on the use of various lightweight block cypher algorithms. The working principle and algorithm of the used block cypher cryptography techniques are explained in Section VI. The simulation results and analyses are presented in Section VII. Section VIII gives an overview of the research's future scope, followed by a conclusion in section IX.

2 RELATED WORKS

Bassam Aboushosha [15], proposed a symmetric block cipher technique called SLIM based on 32-bit block size Feistal structure which uses 4 S-boxes to perform nonlinear operation on 16-bit word and is highly efficient against linear and differential attacks. The technique proved to be effective in wireless sensor networks where the transmission width is only few bytes. Further, XinXin Fan [16], discusses a lightweight stream cipher WG-8 originated from the Welch-Gong family of cryptography. Some of the

block ciphers such as TEA Wheeler and XTEA has also been proposed. Implementation of this cipher in low power microcontrollers proved that they are highly efficient and consume very less power. Then, an ultra-lightweight block cipher called QTL has been suggested by Li et al. [17] which is slight variation of the FN and is capable of operating at faster speeds compared to other standard internal encryption structures. QTL follows the same encryption and decryption process and is proved to occupy a smaller area and highly cost effective. Further, Biswas [18] surveyed a verity of security mechanism such as KATAN, TWINE, AES and LED which are some standard mechanisms adopted for data confidentiality. He proposed a technique using chaotic maps and genetic operations which uses points on elliptical curves to find the communicating nodes. Moreover this, SecureData, a method developed for IoT based human services that collects data by preserving the privacy of the users was analyzed by Hai Tao [19].

The developed method was tested on FPGA equipment using the KATAN technique of encryption. At the cloud level, a circulated database method was adopted in order to preserve patient's privacy. The obtained results proved to be authoritative and authentic. A lightweight blockchain architecture for healthcare database management was proposed by Leila Ismail and Huned Materwala [20]. The network participants are divided into demographic clusters by maintaining one copy of ledger. Forking is avoided by using a using a Head Blockchain Manager to handle transactions. The proposed method outperforms traditional Bitcoin network in terms of network traffic generated and computation speed. Further, a novel ultra-lightweight cryptographic technique named Hummingbird was presented by Daniel Engels, Xinxin Fan [21] provides security with minimal block size and is efficient against linear and differential attacks. The analysis was performed on 8-bit Atmel and 16-bit Texas instruments microcontrollers. The simulation showed to achieve 4.7 times faster throughput compared to PRESENT simulated in similar platforms. Then, Chiu C. Tan,, Haodong Wang, Sheng Zhong and Qun Li [22] developed a lightweight identity based cryptography for body sensor networks that manages security, privacy and accessibility for health care monitoring and tested it on commercially available sensors. Simulation results showed that the proposed method performs faster computation than other sensor platforms but suffered from slow query performance compared to other ciphers.

Abdul Rehman Raza, Khawir Mahmood, Muhammad Faisal Amjad, Haider Abbas [23] implemented 64, 80 and 128 bits of LED block cipher across various programming languages such as C++, Java, Python. Software efficiency and throughput was studied using 32- and 64-bit platforms using Windows and Linux operating systems. They have highlighted and studied the impact in the choice of programming language and platform on the performance of the algorithm. Results show that the choice of platform and language can affect the efficiency of an algorithm with a factor as high as 400. Finally, Norah Alassaf 1 & Adnan Gutub [24] have proposed further improvements in SIMON cipher that can be used to preserve medical data in an IoT setup. The work is compared with AES algorithm in terms of memory consumption and execution time. The proposed technique offers high security while maintaining a trade of between cost and performance as well as ROM and RAM memory consumption.

3 MOTIVATION

Health care is one of the fastest sectors to adapt to the changes made in IoT based systems. ""MarketsAndMarkets" predicts that IoHT will be worth US$ 163.2B, commercial report claims a spending of $117B, and McKinsey estimates an economic impact of more than US$ 170B" [25]. development of e-health systems such as electrocardiography, electroencephalogram, diabetes can be cost saving and help patients suffering with chronic diseases reduce the number of hospitals visits [26]. Also, the outbreak of covid-19 pandemic has created a fear in minds of people and refraining them visiting hospitals which could potentially cause them to suffer from the virus. This has enabled the IoHT sector to grow exponentially and will continue to bloom for the next few years. People are now looking for safer and less expensive ways to maintain and monitor their health. Due to the increased number of users, it has become an attractive sector for hackers.

This raises the need to develop IoT based systems with enhanced security that enables safe transfer and computation of patients' data. Security can be achieved by various methods like cryptography, block

chain technology, machine learning techniques like supervised, unsupervised and reinforced learning etc. [27]. This paper focuses on simple and lightweight block ciphers that are implemented to protect the data at the sensing/physical layer of any Iot based system.

4 CONCERNS AND CHALLENGES IN IMPLEMENTATION OF CRYPTOGRAPHIC TECHNIQUES TO RESOURCE CONSTRAINED IoHT DEVICES

From physical sensors to computer servers, any IoT network incorporates a wide variety of platforms. This opens the door to a slew of new concerns for users, including privacy, security, compatibility, scalability, and interoperability [28]. IoT devices are a particularly appealing target for hackers because they interact directly with the actual environment to collect sensitive data [29]. These devices can potentially be physically damaged in addition to being tapped to gather the sensitive data provided. As a result, cyber security is required, which is regarded as a key problem in the implementation of authentication, data security, availability, privacy, and accessibility [30]. The method adopted for securing the sensitive data completely depends on the environment. The proposed method must be suitable and highly secure to the applied layer of an IoT device but should be designed in such a manner that it does not affect any of its regular activities. Conventional PC cryptographic techniques do not fit into this category as these devices are highly resource constrained.
The cryptographic technique used to preserve this information must be designed by keeping in mind the limitation of the device. The major challenges include (see Fig. 2): [31]

- Low computation power
- Lower energy
- Reduction in availability of space due to smaller size
- Reduction in memory space (ROM and RAM)
- Lower power
- Faster execution time

Fig 2: Challenges in implementation of cryptography

5 SOLUTIONS TO ENHANCE SECURITY IN PHYSICAL LAYER OF IoHT DEVICES

The main characteristics to be taken into consideration while choosing the right cryptographic techniques are cost, performance and security level. Performance can further be divided into subsections such as energy and power consumption, latency, computation speed, memory occupation and different attack models such as linear and differential attacks, side channel attacks and gault injection attacks [32]. Most of the above-mentioned issues are resolved using LWC techniques with a simple key and fewer rounds, but block cypher security is achieved by employing a rigorous internal structure such as FN, hybrid, SPN, GRX, and others to make it impervious to attacks [1]. Cryptographic techniques are categorized as symmetric and asymmetric based on the number of keys. A symmetric cryptographic technique uses the same key for encryption as well as decryption whereas asymmetric technique has two private public key pairs [33]. The encryption and decryption processes of symmetric block cyphers are continuous. In a symmetric block cypher, obtaining the plain text in the reverse procedure is difficult. As a result, they outperform asymmetric cyphers since the usage of two separate keys slows down the computing process [34].

6 LIGHTWEIGHT CRYPTOGRAPHIC TECHNIQUES

The general architecture, encryption mechanism and the size of the plaintext and key corresponding to the LWC algorithms namely CLEFIA, KATAN and SIMON are discussed below.

6.1 CLEFIA

CLEFIA is a 128-bit symmetric block cipher which is developed based on general Feistel network (GFN) and can be implemented by using key ciphers of 128-bits, 192 bits and 256 bits. A GFN based technique is an extrapolated version of FN. The text is encrypted by spitting the word into few sub-blocks and applying FN mechanism and further performing cyclic rotation of bits based on the number of sub-blocks [35]. It is considered as one of the best alternatives for AES, a standard encryption method adopted by the U.S government to protect sensitive data [36-37].

The main techniques involved are branching and number of rounds of encryption. The data processing part of CLEFIA contains 4-whitening keys, 2n number of round keys (n is number of rounds) which are 32 bits wide (see Fig 3). A key scheduling process takes place in order to produce intermediate keys which are usually updated in every two rounds which in turn is expanded to derive at 2n round keys and 4 whitening keys [38-39]. The two s boxes are useful in order to face algebraic and byte ordering saturation attacks [40].

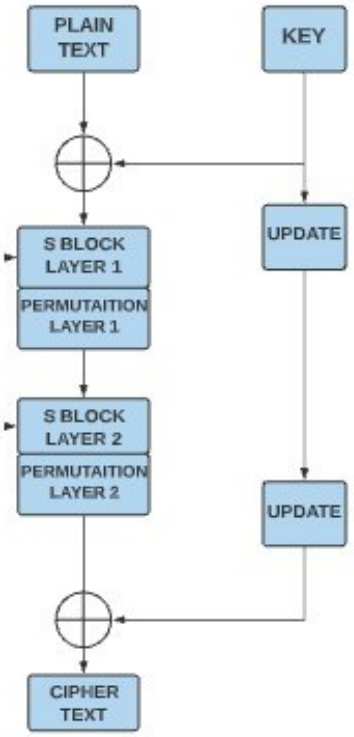


Fig 3: CLEFIA flowchart [35]

Table 1: Building block of CLEFIA algorithm

BLOCK SIZE (BITS)	KEY SIZE (BITS)	NO OF BRANCHES (D)	NO FO ROUNDS (N)
128	128	4	18
128	192	8	22
128	256	8	26

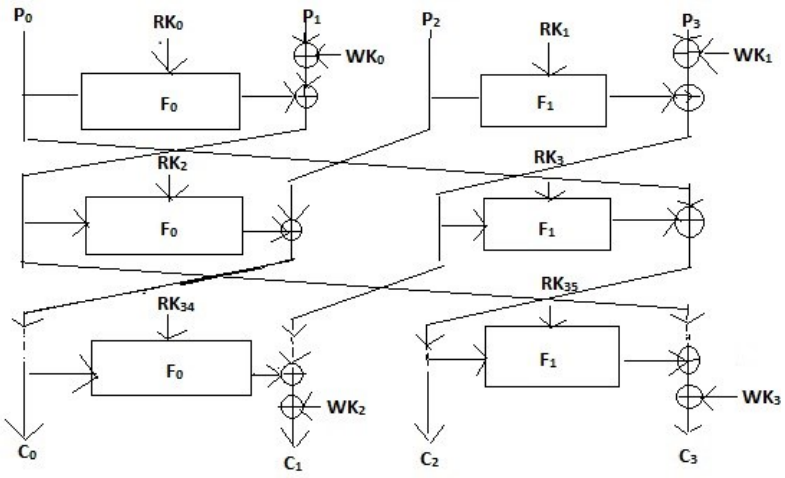


Fig 4: CLEFIA encryption structure [41]

The encrypted text is obtained by passing the original message through two layers of s-boxes and permutation levels.

6.2 KATAN

KATAN inspired by KeeLoq, is a subsection of the block cipher family which consists of KATAN32, KATAN48 and KATAN64 with block sizes 32, 48 and 65 bits respectively and a fixed key size of 80 bis [42]. It is highly notable for its simplicity and is more efficient as the encryption process runs in a parallel fashion consisting of three pipelined stages [40]. It follows a linear structure (LFSR) rather than NLFSR proposed by KeeLoq [43]. The encryption process takes a total of 254 rounds. The plain text is loaded into two registers L1 and L2. In each round a few bits form L1 and L2 are computed using predefined non-linear functions and stored in the LSB (Least significant bit) of L1 and l2 respectively [44]. The non-linear functions are defined as follows:

$$fa(L1) = L1[x1] \oplus L1[x2] \oplus (L1[x3] \cdot L1[x4]) \oplus (L1[x5] \cdot IR) \oplus keya \quad (1)$$

$$fb(L2) = L2[y1] \oplus L2[y2] \oplus (L2[y3] \cdot L2[y4]) \oplus (L2[y5] \cdot L2[y6]) \oplus keyb \quad (2)$$

Where {x} ands {y} are defined in table 2 and Keya and keyb are sub-key bits. The number of times the nonlinear function is applied in a round varied from 1 in KATAN32 to 2 and 3 in KATAN 48 and KATAN64 respectively [45].

Table 2: Parameters of KATAN

	BLOCK SIZE (BITS)		
PARAMETER	**32**	**48**	**64**
|L1|	13	19	25
|L2|	19	29	39
{x1}, {x2}, {x3}, {x4}, {x5}	12, 7, 8, 5, 3	18, 12, 15, 7, 6	24, 15, 20, 11, 9
{y1}, {y2}, {y3}, {y4}, {y5}	18, 7, 12, 10, 8, 3	29, 19, 21, 13, 15, 6	38, 25, 33, 21, 14, 9

Similar to any block cipher, KATAN algorithm includes various intermediate process taking place during encryption (see Fig. 5 and 6).

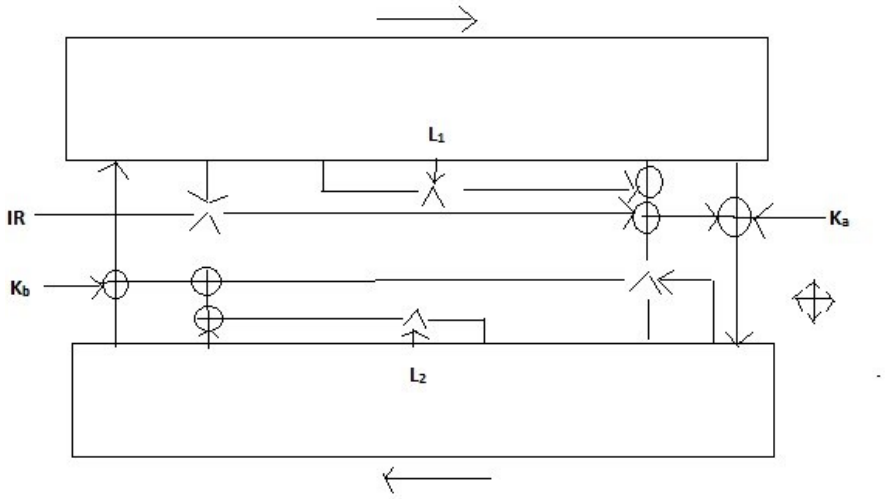


Fig. 5: KATAN Structure [39]

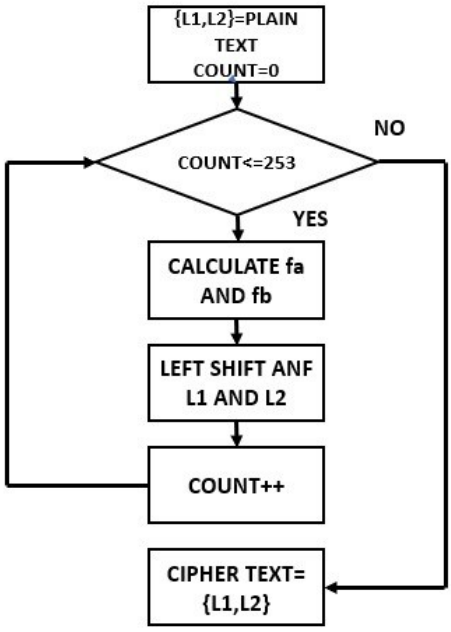


Fig. 6: KATAN flowchart

The IR (Irregular update) rule in KATAN ensures that not mor than seven rounds share the same pattern of updates. This ensures than not more than 7 rounds can be utilized by self-similarity attacks. Due to this, these kinds of attacks often fail in KATAN family of ciphers.

6.3 SIMON

Simon is a lightweight lock cipher proposed by National Security Agency (NSA) in 2013 as a part of Simon and Speck family of ciphers [46]. The Simon cipher can be represented as Simon 2n/mn where, 2n is the block size, n is the word size and m are the number of key words. The encryption process is done using the round function which performs the following operations on the plaintext [47].

- BITWISE XOR
- BITWISE AND
- LEFT CIRCULAR SHIFT

The plaintext block is split into two equal parts namely left and right block. Each round performs three left shift operations of the left block then ANDs it and the resultant value is XORed with the right block and is stored in the left block (see Fig. 7 and 8).

$$F(x, y, k) = (y \oplus ((S^1 x \,\&\, S^8 x) \oplus S^2 x) \oplus k) \tag{3}$$

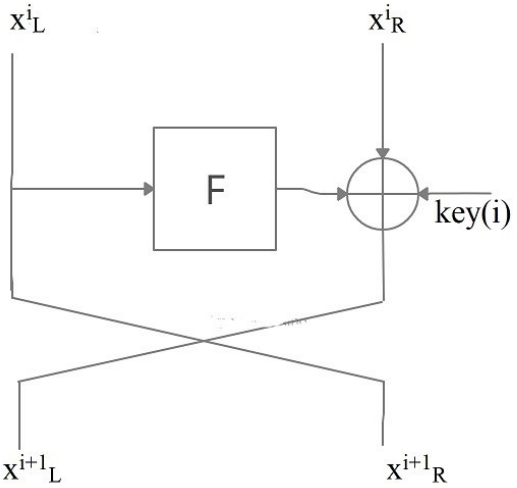


Fig. 7: Structure of SIMON [45]

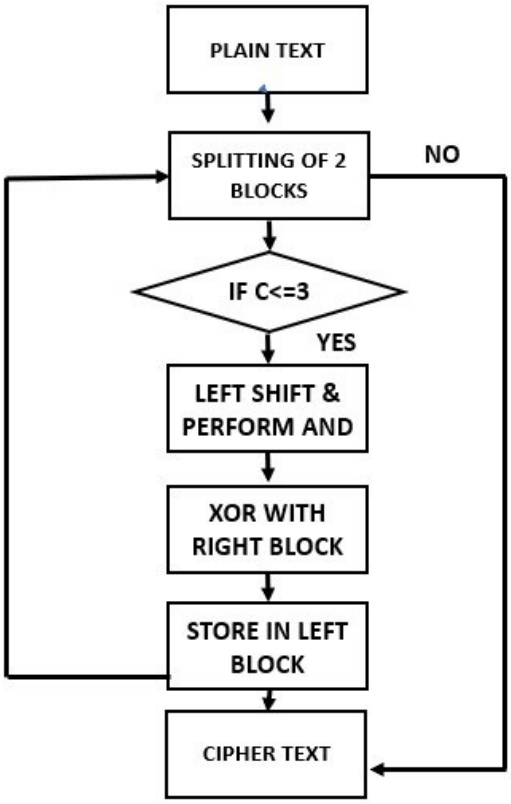


Fig. 8: SIMON cipher flowchart

Where, k is the key size, x and y ae the left and right blocks respectively. Simon is more suitable for hardware applications as it requires a greater number of rounds for encryption [48]. Due to the multiple rounds of encryption process its highly secure against integral, man in the middle and differential attacks [49]. SIMON can be implemented using variable block size (see table 3).

Table 3: Parameters of SIMON

BLOCK SIZE (m bits)	KEY WORDS (n bits)	KEY SIZE (mn bits)	NUMBER OF ROUNDS
32	4	64	32
48	3	72	36
64	3	96	42
96	2	92	52
128	2	128	68
128	3	192	69
128	4	256	72

7 RESULTS AND DISCUSSION

Execution time, encryption and decryption time, memory usage during execution, and size are all used to evaluate the performance of existing cryptographic algorithms in software. These algorithms were tested using the Python programming language on a variety of platforms, including Conda, Python IDLE, and Google Colab, using an Intel i5 core CPU with a clock speed of 1.6 GHz.

7.1 SIMULATION RESULTS

The following is a sample of the results produced using CLEFIA 128-bit key encryption and decryption (see Fig. 9).

```
-----LIGHTWEIGHT CRYPTOGRAPHY-----
'------------CLEFIA------------
Plain text  52331006062428060509553957313612 95

Encrypted text 167983052394116682426995 17
Average elapsed time for 16-byte block 128 encryption: 2.508ms
Deccrypted text 52331006062428060509553957313612 95
Average elapsed time for 16-byte block 128 decryption: 2.430ms
Total memory consumption during execution is  60.30078125 MiB


Clefia file size 16.635 KB
execution time 5.156162977218628 s
```

Fig 9a: Performance of CLEFIA algorithm on Conda

```
-----LIGHTWEIGHT CRYPTOGRAPHY-----
'------------CLEFIA------------
Plain text  52331006062428060509553957313612 95

Encrypted text 167983052394116682426995 17
Average elapsed time for 16-byte block 128 encryption: 2.285ms
Deccrypted text 52331006062428060509553957313612 95
Average elapsed time for 16-byte block 128 decryption: 1.678ms
Total memory consumption during execution is  25.25 MiB


Clefia file size 16.173 KB
execution time 4.4595160484313965 s
```

Fig 9b: Performance of CLEFIA algorithm on Python IDLE

```
-----LIGHTWEIGHT CRYPTOGRAPHY-----
'------------CLEFIA------------
Plain text  52331006062428060509553957313612 95

Encrypted text 167983052394116682426995 17
Average elapsed time for 16-byte block 128 encryption: 2.703ms
Deccrypted text 52331006062428060509553957313612 95
Average elapsed time for 16-byte block 128 decryption: 2.648ms
Total memory consumption during execution is  117.00390625 MiB
execution time 5.561190605163574 s
```

Fig 9c: Performance of CLEFIA algorithm on Colab

A tabulation of the simulated results in three platforms: Conda, Python IDLE and Google Colab is given below in table 2. The simulation is performed for variable key sizes: 128-bit, 192-bit and 256-bit with standard block size of 128-bit.

Table 4: Comparison of CLEFIA on different platforms

PLATFORM	BLOCK SIZE	KEY SIZE	NO. OF ROUNDS	EXECUTION TIME (s)	ENCRYPTION TIME m(s)	DECRYPTION TIME (ms)	MEMORY OCCUPIED DURING EXECUTION (MiB)
CONDA	128	128	18	5.159	2.508	2.430	
		192	22	7.061	3.636	3.218	59.45
		256	26	7.938	3.655	4.079	
PYTHON IDLE	128	128	18	4.46	2.285	1.678	
		192	22	5.293	2.638	2.250	25.29
		256	26	5.952	3.022	2.453	
GOOGLE COLAB	128	128	18	5.56	2.703	2.648	
		192	22	7.486	3.634	3.640	117.98
		256	26	8.275	4.024	4.040	

It is clear from the data that the parameters change as the platform changes. Python IDLE has the fastest algorithm, with an execution time of 4.56 seconds and encryption and decryption times of 2.285 seconds and 1.678 seconds, respectively. When compared to other platforms, Python IDLE consumes the least amount of memory. Python IDLE consumes 25 MiB of RAM, while Colab consumes 117 MiB. A total of 16.173 KB of RAM memory is used. As the size of the key increases, so does the time it takes to execute it. Because the number of rounds increases as the key size grows, encrypting the data takes longer. KATAN algorithm is analyzed using similar parameters used to measure the performance of CLEFIA (see Fig. 10). The block size of the data is varied rather than key size whereas the opposite takes place in CLEFIA.

```
-----LIGHTWEIGHT CRYPTOGRAPHY-----
------------KATAN-----------


Key (length of 80 bits)  0x123456789abcdef123fd
Plain text (length of 48 bits)  0x1d34a6782345
Encrypted text  0xf79ffc1c
Average time for encryption of 48 bit block:  0.6027052402496338 s
Decrypted text  0x1d34a6782345
Average time for decryption of 48 bit block:  0.6272363662719727 s
Total memory consumption during execution is  59.88671875 MiB
Katan file size 6.12 KB
execcution time is 1.7730042934417725 s
```

Fig 10a: Performance of KATAN algorithm in Conda

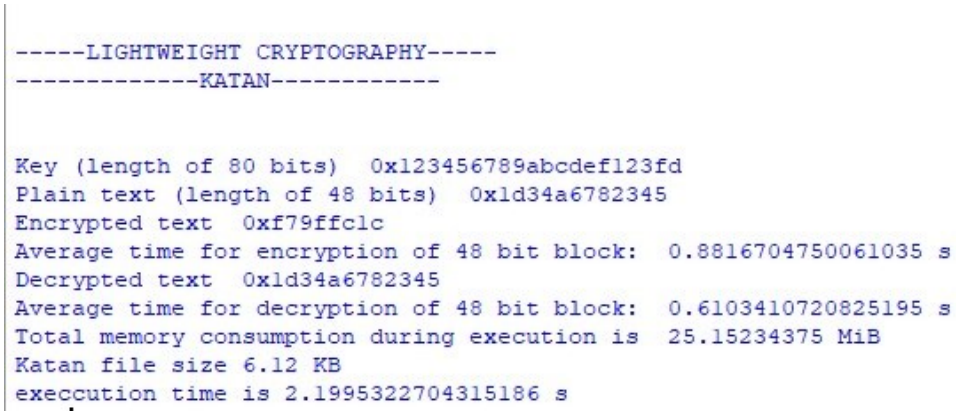


Fig 10b: Performance of KATAN algorithm in Python IDLE

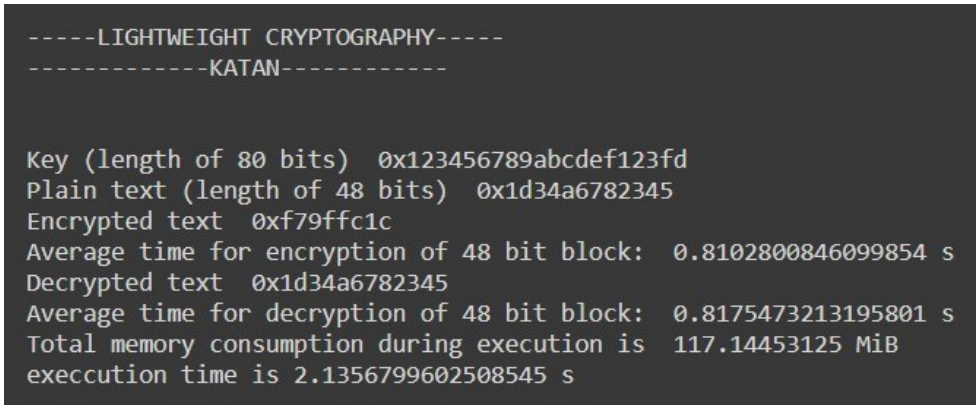


Fig 10c: Performance of KATAN algorithm in Colab

The above-mentioned figures show that the time of execution is really fast compared to CLEFIA. Table 3 explains the simulation results obtained by varying the platform and block size of the plain text. The block size is varied from 32-bits to 48 and 64 bits.

Table 5: Comparison of KATAN on different platforms

PLATFORM	WORD SIZE	KEY SIZE	EXECUTION TIME (s)	ENCRYPTION TIME (s)	DECRYPTION TIME (s)	MEMORY OCCUPIED DURING EXECUTION (MiB)
CONDA	32	80	1.763	0.632	0.628	55.32
	48		1.771	0.619	0.616	
	64		1.835	0.654	0.665	
PYTHON IDLE	32	80	2.268	1.039	0.482	25.254
	48		2.32	0.946	0.535	
	64		2.386	0.977	0.569	
GOOGLE COLAB	32	80	2.128	0.809	0.810	118.46
	48		2.153	0.799	0.846	
	64		2.132	0.805	0.816	

In contrary to the results observed in CLEFIA, KATAN performs better in Conda environment compared to Python IDLE and Google Colab. The memory occupation during execution is slightly higher than CLEFIA. KATAN algorithm executes faster with a time of just 1.763 seconds and requires only 0.6 seconds approximately to encrypt and decrypt the data. Similar results are obtained for SIMON cipher (see Fig. 11).

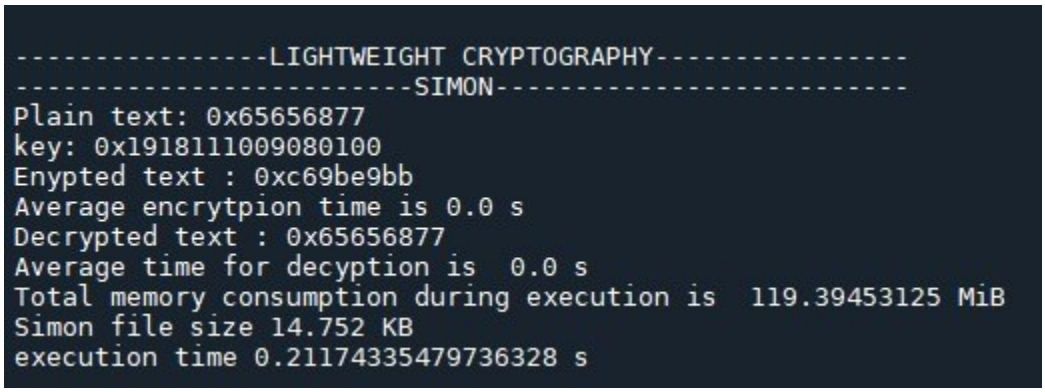


Fig 11a: Performance of SIMON in Conda

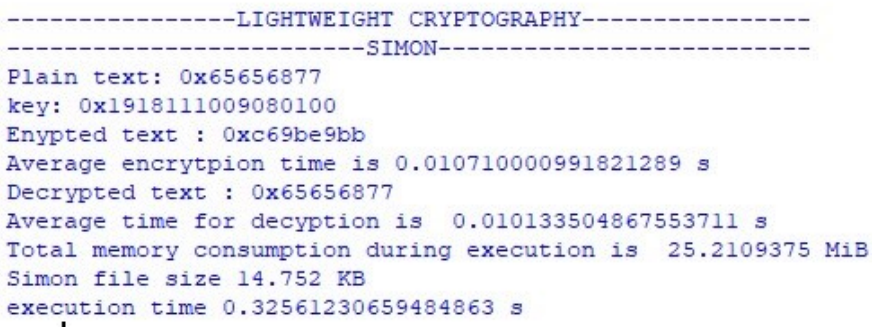


Fig 11b: Performance of SIMON in Python IDLE

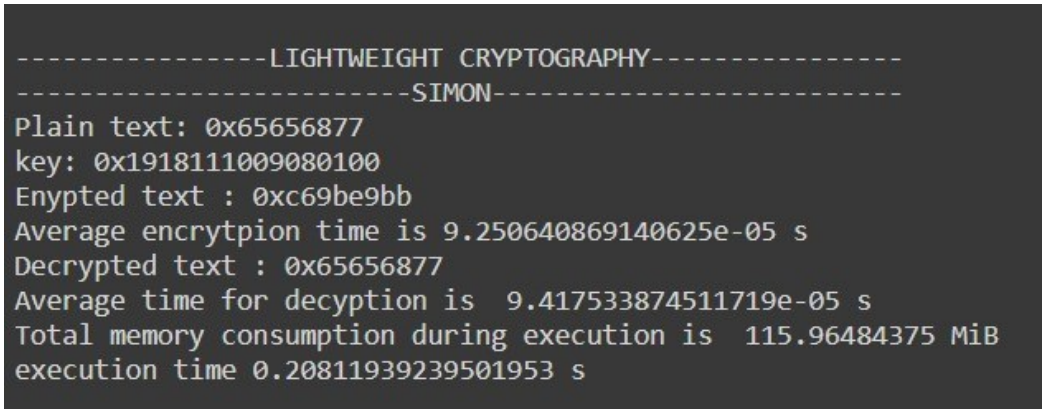


Fig 11c: Performance of SIMON in Colab

It can be observed from the figure that SIMON cipher performs better than CLEFIA but not KATAN in term of execution time.

Table 6: Comparison of SIMON on different platforms

PLATFORM	BLOCK SIZE	KEY SIZE	EXECUTION TIME (s)	ENCYRPTION TIME (ms)	DECYRTOPIN TIME (ms)	MEMORY OCCUPIED DURING EXECUTION (MiB)
PYHTON IDLE	32	64	0.326	9.85	8.69	25.211
	48	72	0.339	10.12	9.98	25.17
	64	96	0.318	10.71	10.13	25.18
	128	128	0.323	11.356	9.569	25.141
	128	192	0.347	13.33	10.59	25.24
	128	256	0.331	11.67	11.39	25.15
CONDA	32	64	0.212	Failed	Failed	54.32
	48	72	0.227	Failed	Failed	55.76
	64	96	0.231	9.95	Failed	55.86
	128	128	0.221	Failed	7.43	55.82
	128	192	0.223	Failed	Failed	55.68
	128	256	0.215	Failed	Failed	56.016
GOOGLE COLAB	32	64	0.208	0.0925	0.0941	115.96
	48	72	0.209	0.0954	0.107	115.29
	64	96	0.212	0.0942	0.159	116.97
	128	128	0.208	0.133	0.156	116.44
	128	192	0.209	1.39	0.346	117.16
	128	256	0.210	2.17	0.938	117.17

Table 4 illustrates that the simulation results obtained for SIMON cipher does not flow a particular trend and not all simulations were successful. However, the simulations executed using Python IDLE and Colab are quite successful. The encryption and decryption times vary almost linearly with the key and block sizes.

7.2 PERFORMANCE ANALYSIS

The performance of the techniques is analyzed using the following parameters: file size, execution time, encryption and decryption times and number of rounds. A comparison of the simulated cryptographic techniques on the basis of number of rounds with key sizes and block sizes is studied (see Fig. 12). The strength of a cryptographic technique increases with the increase in number of encryptions rounds as it becomes difficult to attack and go through multiple layers According to thumb rule the security of any block cipher is half the key size provided there are no possible methods better than brute force to crack the cipher [50]. The future research in hardware and software is motivated by least memory consumption as IoT devices are highly memory constrained. The efficiency of any LWC algorithm highly depends on the memory occupation.

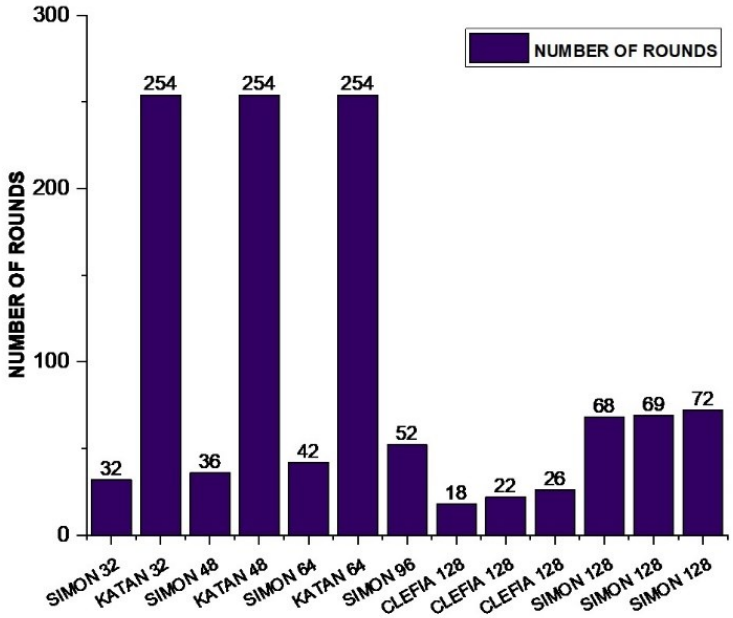


Fig 12: Number of rounds of encryption

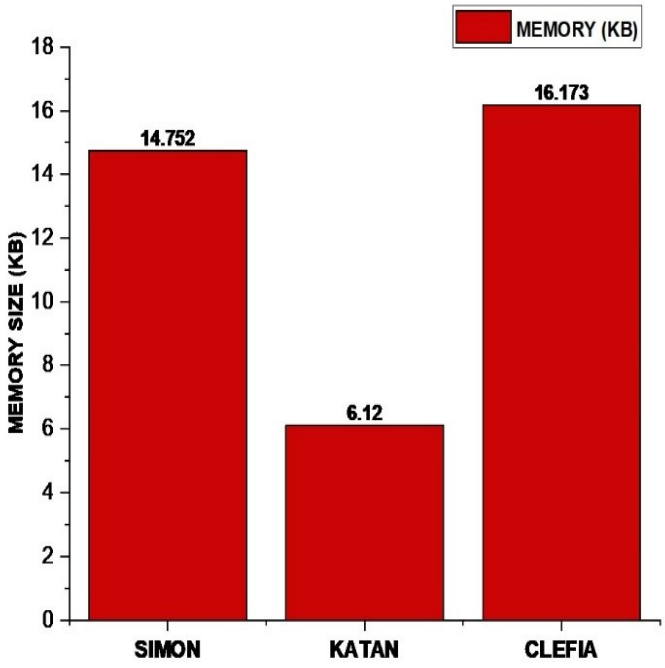


Fig 13: File size

KATAN is more secure than CLEFIA and SIMON due to a greater number of encryption rounds. KATAN is also more compact than the other techniques as it occupies the least memory space of 6.12 Kbytes and surpasses SIMON and CLEFIA by 58.5% and 62.15% respectively (see Fig 12 and 13). According to the graphs, KATAN is more suitable to be applied in IoHT devices due to greater security and lesser memory occupation followed by SIMON and CLEFIA. Another important performance metric is the execution time. The total execution time includes the time taken for encryption, decryption and generation of round keys. Individual performance of each algorithm is tested in various platforms. Comparison of the execution times of the KATAN, CLEFIA and SIMON tested on various platforms is studied (see Fig. 14).

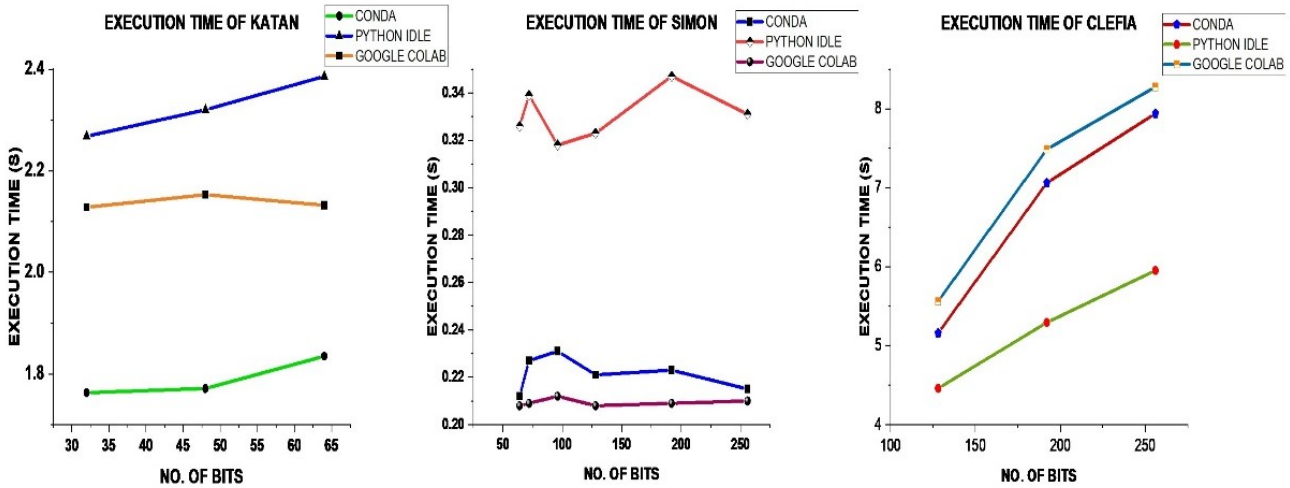
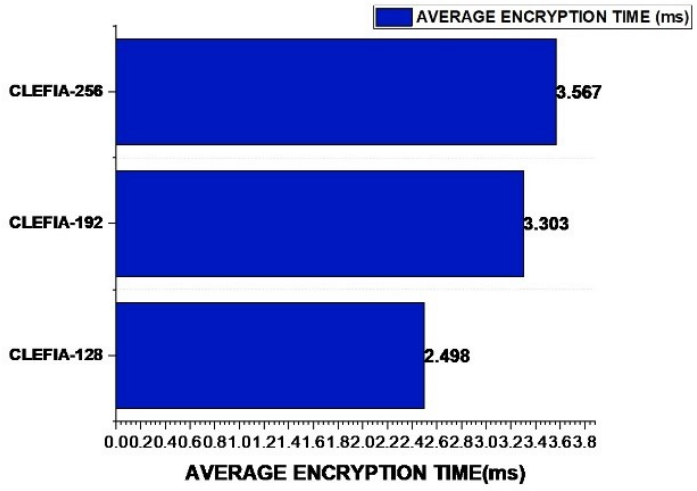
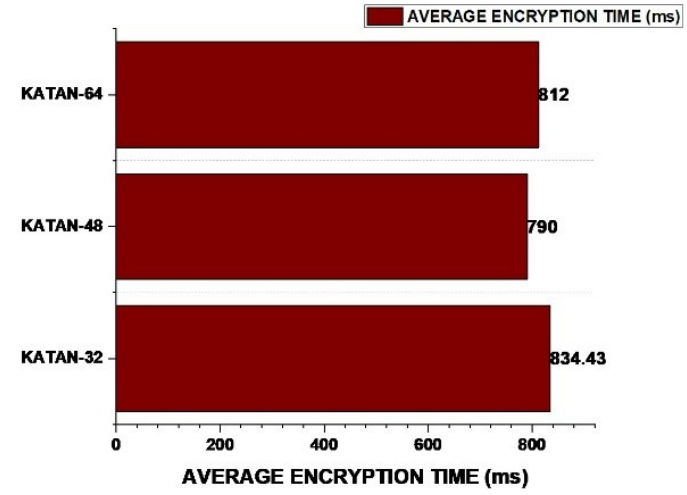


Fig. 14: Execution time

SIMON appears to be the fastest, followed by KATAN and CLEFIA, according to the graph (see Fig. 12). SIMON is approximately 88% and 96% faster than KATAN and CLEFIA respectively. The execution time varies with platform and method used in the LWC algorithms. SIMON performs best in Google Colab whereas KATAN and CLEFIA perform better in Conda and IDLE respectively. The execution speed of the algorithm depends on the number of keys generated and the number of rounds. SIMON uses simple XOR and AND logical operations and is implemented in text sizes of 32,48,64 and 128 bits with minimum key size and number of rounds of encryption [51]. This simple yet secure implementation makes SIMON the fastest algorithm compared to other techniques. Another parameter of relevance is the encryption time of various LWC algorithms (see Fig 15).


AVERAGE ENCRYPTION TIME (ms)
CLEFIA-256
CLEFIA-192
CLEFIA-128
3.567
3.303
2.498
0.00.20.40.60.81.01.21.41.61.82.02.22.42.62.83.03.23.43.63.8
AVERAGE ENCRYPTION TIME(ms)


AVERAGE ENCRYPTION TIME (ms)
KATAN-64
KATAN-48
KATAN-32
812
790
834.43
0
200
400
600
800
AVERAGE ENCRYPTION TIME (ms)

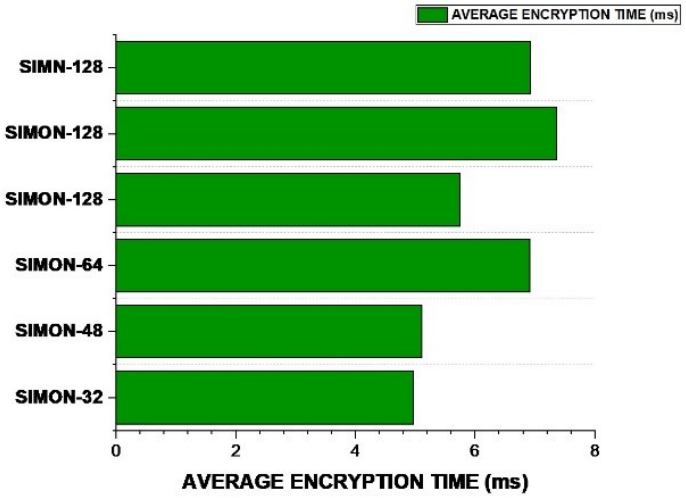

AVERAGE ENCRYPTION TIME (ms)
SIMN-128
SIMON-128
SIMON-128
SIMON-64
SIMON-48
SIMON-32
0
2
4
6
8
AVERAGE ENCRYPTION TIME (ms)

Fig 15: Average encryption time

From the graph we can say that SIMON again demands the least encryption time with a minor difference from CLEFIA. Even though CLEFIA has lesser rounds of encryption compared to SIMON and KATAN, the multiple key scheduling processes and permutation levels takes more time than encryption of data using simple logical expressions and shifting operations. Whereas on the other hand, KATAN is very slow compared to other techniques as it goes through 254 algorithm rounds for all block sizes [52].

8 FUTURE SCOPE

IoT applications are growing rapidly day by day and as most of the industries are moving towards IoT, energy consumption is one of the main constraints of the IoT world [53-54]. The limited computational capabilities and resource constraints make it a vulnerable target for hackers [55]. IoHT is a field that is widely in use now. It deals with millions of patients' health information which needs to be secured in order to prevent misuse. In future, light weight cryptographic encryption in IoHT can improve the security level of the system and help to make the devices more efficient and secure.

Lightweight block ciphers are efficient in both hardware as well as software. Asymmetric block ciphers, stream, hash and elliptical curve functions are other available techniques which have a high potential to be employed in these devices to secure patients' information in IoHT [56]. The future work includes study of software performance using other programming languages such as C++, Java and other operating systems such as Linux. Further the hardware performance can be studied by implementing these techniques in real time embedded systems, ARM-based microprocessors and dedicated integrated circuits which are widely used in IoHT industry to observe various parameters like circuit-footprint, throughput, latency, energy and power consumptions in order to design ultra-low power IoT devices.

In the last, in [57-65] authors have recommended others to read these research efforts, to know more information about IoHT and role of AI, Computer Vision, or Machine learning techniques respect to this sensitive area/ useful application. We hope that readers/ researchers will find suitable problem for themselves/ for their research work (from these research works).

9. CONCLUSIONS

This paper analyses the software implementation of some widely used symmetric block ciphers in IoHT devices namely CLEFIA, KATAN and SIMON. The performance of these algorithms in terms of speed, and memory consumption was studied platforms like Python IDLE, Colab and Conda. The simulation results showed that KATAN occupies less memory than CLEFIA and SIMON by 58.5% and 62.15% and a greater number of rounds for more secure encryption. Whereas SIMON proves to be better in terms of speed by 88% and 96% than KATAN and CLEFIA due to a simpler yet secure method of encryption. As a result, it can be stated that KATAN is more suitable for devices with high resource constraints in terms of memory and SIMON is preferable for solutions which can respond to sensory inputs within specified times like real-time embedded systems.

ACKNOWLEDGEMENT

The authors want to thank Centre for Advanced Data Science and School of Computer Science and Engineering, Vellore Institute of Technology, Chennai for providing their kind support to complete this research work on time.

REFERENCES

1. V. A. Thakor, M. A. Razzaque and M. R. A. Khandaker, "Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review, Comparison and Research Opportunities," *in* IEEE Access, vol. 9, pp. 28177-28193, (2021).

2. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of things (iot): A vision, architectural elements, and future directions,” Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, (2013).
3. Singh, S., Sharma, P.K., Moon, S.Y. *et al.* Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions. J Ambient Intell Human Comput (2017).
4. L. Tawalbeh, F. Muheidat, M. Tawalbeh, and M. Quwaider, “IoT Privacy and Security: Challenges and Solutions,” Applied Sciences, vol. 10, no. 12, p. 4102, Jun. (2020).
5. A. o. A. B. Engineering. (Frost & Sullivan). Internet of Medical Things Revolutionizing Healthcare, https://aabme.asme.org/posts/internet-of-medical-thingsrevolutionizing-healthcare, (2017).
6. Fuzon, "Internet of Medical Things (IoMT): New Era in Healthcare Industry," (2019).
7. K. McKay, L. Bassham, M. S. Turan, and N. Mouha, Report on Lightweight Cryptography (Nistir8114). Gaithersburg, MD, USA: NIST, (2017).
8. O. Toshihiko, ‘‘Lightweight cryptography applicable to various IoT devices,’’ NEC Tech. J., vol. 12, no. 1, pp. 67–71, (2017).
9. A. Biryukov and L. P. Perrin, ‘‘State of the art in lightweight symmetric cryptography,’’ Univ. Luxembourg Library, Esch-sur-Alzette, Luxembourg, Tech. Rep. 10993/31319, (2017).
10. Ullah A, Sehr I, Akbar M, Ning H FoG assisted secure De-duplicated data dissemination in smart healthcare IoT. In: 2018 IEEE international conference on smart internet of things (SmartIoT). IEEE, pp 166–171, (2018).
11. C. Butpheng, K.-H. Yeh, and H. Xiong, “Security and Privacy in IoT-Cloud-Based e-Health Systems—A Comprehensive Review,” Symmetry, vol. 12, no. 7, p. 1191, Jul. (2020).
12. K. McKay, L. Bassham, M. S. Turan, and N. Mouha, Report on Lightweight Cryptography (Nistir8114). Gaithersburg, MD, USA: NIST, (2017).
13. Chaudhury S, Paul D, Mukherjee R, Haldar S Internet of Thing based healthcare monitoring system. In: Industrial automation and electromechanical engineering conference (IEMECON), 2017 8th annual. IEEE, pp 346–349, (2017).
14. El-hajj M, Chamoun M, Fadlallah A, Serhrouchni A (2017) Analysis of authentication techniques in internet of things (IoT). In: Cyber security in networking conference (CSNet), 1st. IEEE, pp 1–3, (2017).
15. B. Aboushosha, R. A. Ramadan, A. D. Dwivedi, A. El-Sayed and M. M. Dessouky, "SLIM: A Lightweight Block Cipher for Internet of Health Things," in IEEE Access, vol. 8, pp. 203747-203757, (2020).
16. Fan X, Mandal K, Gong G Wg-8: a lightweight stream cipher for resource-constrained smart devices. In: Proceeding of International Conference on Heterogeneous Networking for Quality, Reliability, (2013).
17. Li L, Liu B, Wang H. QTL: a new ultra-lightweight block cipher. Microproc Microsys 45: 45–55, (2016).
18. Biswas K, Muthukkumarasamy V, Singh K An encryption scheme using chaotic map and genetic operations for wireless sensor networks. IEEE Sens J 15(5): 2801–2809, (2015).
19. Tao H, Bhuiyan MZA, Abdalla AN, Hassan MM, Zain JM, Hayajneh T Secured data collection with hardware-based ciphers for IoT-based healthcare. IEEE Internet Things J:1–10, (2018).
20. L. Ismail, H. Materwala and S. Zeadally, "Lightweight Blockchain for Healthcare," in *IEEE Access*, vol. 7, pp. 149935-149951, (2019).
21. Harikrishnan T and C. Babu, "Cryptanalysis of hummingbird algorithm with improved security and throughput," 2015 International Conference on VLSI Systems, Architecture, Technology and Applications (VLSI-SATA), pp. 1-6, (2015).
22. C. C. Tan, H. Wang, S. Zhong and Q. Li, "IBE-Lite: A Lightweight Identity-Based Cryptography for Body Sensor Networks," in IEEE Transactions on Information Technology in Biomedicine, vol. 13, no. 6, pp. 926-932, Nov. 2009.

23. Raza, Abdur Rehman, et al. "On the efficiency of software implementations of lightweight block ciphers from the perspective of programming languages." Future Generation Computer Systems 104, 43-59, (2020).
24. Alassaf, N., Gutub, A., Parah, S.A. et al. Enhancing speed of SIMON: A light-weight-cryptographic algorithm for IoT applications. Multimed Tools Appl **78,** 32633–32657 (2019).
25. J. J. P. C. Rodrigues et al., "Enabling Technologies for the Internet of Health Things," in *IEEE Access*, vol. 6, pp. 13129-13141, (2018).
26. A.M. Khairuddin, K.N.F.K. Azir and P.E. Kan, "Limitations and future of electrocardiography devices: A review and the perspective from the Internet of Things". International Conference on Research and Innovation in Information Systems, pp. 1-7, (2017).
27. L. Xiao, X. Wan, X. Lu, Y. Zhang and D. Wu, "IoT Security Techniques Based on Machine Learning: How Do IoT Devices Use AI to Enhance Security?" in IEEE Signal Processing Magazine, vol. 35, no. 5, pp. 41-49, Sept. (2018).
28. M. M. Nair, A. K. Tyagi and N. Sreenath, "The Future with Industry 4.0 at the Core of Society 5.0: Open Issues, Future Opportunities and Challenges," 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021, pp. 1-7, doi: 10.1109/ICCCI50826.2021.9402498.
29. W. Feng, Y. Qin, S. Zhao, and D. Feng, ''AAoT: Lightweight attestation and authentication of low-resource things in IoT and CPS,'' Comput. Netw., vol. 134, pp. 167–182, (2018).
30. B. J. Mohd, T. Hayajneh, and A. V. Vasilakos, ''A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues,'' J. Netw. Comput. Appl., vol. 58, pp. 73–93, (2015).
31. S. Singh, P. K. Sharma, S. Y. Moon, and J. H. Park, ''Advanced lightweight encryption algorithms for IoT devices: Survey, challenges and solutions,'' J. Ambient Intell. Hum. Comput., vol. 4, pp. 1–18,(2017).
32. W. Stallings, Cryptography and Network Security: Principles and Practice, (2017).
33. I. Bhardwaj, A. Kumar, and M. Bansal, ''A review on lightweight cryptography algorithms for data security and authentication in IoTs,'' in Proc. 4th Int. Conf. Signal Process., Comput. Control (ISPCC), pp. 504–509, (2017).
34. Rashidi, Bahram. "Efficient and flexible hardware structures of the 128-bit CLEFIA block cipher." IET Computers & Digital Techniques 14.2, 69-79, (2020).
35. P. Saravanan, S. S. Rani, S. S. Rekha and H. S. Jatana, "An Efficient ASIC Implementation of CLEFIA Encryption/Decryption Algorithm with Novel S-Box Architectures," 2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP), pp. 1-6, (2019).
36. Jangra, Monika; Singh, Buddha. Performance analysis of CLEFIA and PRESENT lightweight block ciphers. Journal of Discrete Mathematical Sciences and Cryptography, 22(8), 1489–1499, (2019).
37. Ertaul, L., & Rajegowda, S. K. Performance Analysis of CLEFIA, PICCOLO, TWINE Lightweight Block Ciphers in IoT Environment. In Proceedings of the International Conference on Security and Management (SAM). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp) (pp. 25-31), (2017).
38. Shirai, T., Shibutani, K., Akishita, T., Moriai, S., & Iwata, T. (2007, January). The 128-bit blockcipher CLEFIA. In Fast software encryption (pp. 181-195). Springer Berlin Heidelberg.
39. Tezcan, C. (2010). The improbable differential attack: Cryptanalysis of reduced round CLEFIA. In Progress in Cryptology-INDOCRYPT 2010 (pp. 197-209). Springer Berlin Heidelberg.
40. Shuai Su, Hang Dong, Ge Fu, Chengpeng Zhang and Miao Zhang, "A White-Box CLEFIA implementation for mobile devices," 2014 Communications Security Conference (CSC 2014), pp. 1-8, (2014).

41. Mohd, Bassam Jamil, et al. "Hardware design and modeling of lightweight block ciphers for secure communications." Future Generation Computer Systems 83, (2018)
42. F. M. Qatan and I. W. Damaj, "High-speed KATAN ciphers on-a-chip," 2012 International Conference on Computer Systems and Industrial Informatics, pp. 1-6, (2012).
43. C. De Canniere, O. Dunkelman, and M. Knežević, ‘‘KATAN and KTANTAN—A family of small and efficient hardware-oriented block ciphers,’’ in Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst. Berlin, Germany: Springer, 2009, pp. 272–288.
44. B. J. Mohd, T. Hayajneh and Z. Abu Khalaf, "Optimization and modeling of FPGA implementation of the KATAN Cipher," 2015 6th International Conference on Information and Communication Systems (ICICS), pp. 68-72, (2015).
45. Abed, S.; Jaffal, R.; Mohd, B.J.; Alshayeji, M. FPGA Modeling and Optimization of a SIMON Lightweight Block Cipher. *Sensors*, (**2019**)
46. S. Abed, R. Jaffal, B. Mohd, and M. Alshayeji, “FPGA Modeling and Optimization of a SIMON Lightweight Block Cipher,” Sensors, vol. 19, no. 4, p. 913, (2019).
47. AlKhzaimi, H., & Lauridsen, M. M. Cryptanalysis of the SIMON Family of Block Ciphers. IACR Cryptol. ePrint Arch., 543, (2013).
48. AlAssaf N, AlKazemi B, Gutub A. Applicable Light-Weight Cryptography to Secure Medical Data in IoT Systems. Journal of Research in Engineering and Applied Sciences (JREAS) 2(2):50–58, (2017).
49. Kondo K., Sasaki Y., Iwata T.On the Design Rationale of Simon Block Cipher: Integral Attacks and Impossible Differential Attacks against Simon Variants. In: Manulis M., Sadeghi AR., Schneider S. (eds) Applied Cryptography and Network Security. ACNS 2016. Lecture Notes in Computer Science, vol 9696, (2016).
50. Secure Programming Cookbook for C and C++: Recipes for Cryptography, Authentication, Input Validation & More” by John Viega and Matt Messier . (2003). In M. M. John Viega.
51. Kölbl S., Leander G., Tiessen T. Observations on the SIMON Block Cipher Family. In: Gennaro R., Robshaw M. (eds) Advances in Cryptology -- CRYPTO 2015. CRYPTO 2015. Lecture Notes in Computer Science, vol 9215, (2015).
52. H. Tao, M. Z. A. Bhuiyan, A. N. Abdalla, M. M. Hassan, J. M. Zain and T. Hayajneh, "Secured Data Collection with Hardware-Based Ciphers for IoT-Based Healthcare," in IEEE Internet of Things Journal, vol. 6, no. 1, pp. 410-420, (2019).
53. Dhanda, S.S., Singh, B. & Jindal, P. Lightweight Cryptography: A Solution to Secure IoT. Wireless Pers Commun **112,** 1947–1980 (2020).
54. Singh, S., Sharma, P.K., Moon, S.Y. et al. Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions. J Ambient Intell Human Comput (2017).
55. Alladi, T.; Chamola, V.; Sikdar, B.; Choo, K.R. Consumer IoT: Security Vulnerability Case Studies and Solutions. IEEE Consum. Electron., 17–25, Mag. 2020, 9.
56. L. Chen et al., "Robustness, Security and Privacy in Location-Based Services for Future IoT: A Survey," in IEEE Access, vol. 5, pp. 8956-8977, (2017).
57. B. Gudeti, S. Mishra, S. Malik, T. F. Fernandez, A. K. Tyagi and S. Kumari, "A Novel Approach to Predict Chronic Kidney Disease using Machine Learning Algorithms," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, pp. 1630-1635, (2020).
58. Gillala Rekha, V. Krishna Reddy, and Amit Kumar Tyagi, “KDOS - Kernel Density based Over Sampling - A Solution to Skewed Class Distribution”, Journal of Information Assurance and Security (JIAS), Vol. 15 Issue 2, p44-52. 9p, (2020).
59. Gillala Rekha, Amit Kumar Tyagi, and V Krishna Reddy, “Solving Class Imbalance Problem Using Bagging, Boosting Techniques, with and without Noise Filter Method”, International Journal of Hybrid Intelligent Systems, vol. 15, no. 2, pp. 67-76, (2019).
60. Nair M.M., Kumari S., Tyagi A.K., Sravanthi K. Deep Learning for Medical Image Recognition: Open Issues and a Way to Forward. In: Goyal D., Gupta A.K., Piuri V., Ganzha

M., Paprzycki M. (eds) Proceedings of the Second International Conference on Information Management and Machine Intelligence. Lecture Notes in Networks and Systems, vol 166, (2021).
61. Akshara Pramod, Harsh Sankar Naicker, Amit Kumar Tyagi, “Machine Learning and Deep Learning: Open Issues and Future Research Directions for Next Ten Years”, Book: Computational Analysis and Understanding of Deep Learning for Medical Care: Principles, Methods, and Applications, 2020, Wiley Scrivener, (2020).
62. Amit Kumar Tyagi, Aswathy S U, G Aghila, N Sreenath "AARIN: Affordable, Accurate, Reliable and INnovative Mechanism to Protect a Medical Cyber-Physical System using Blockchain Technology" IJIN, Volume 2, Pages 175-183, October 2021.
63. Madhav A.V.S., Tyagi A.K. (2022) The World with Future Technologies (Post-COVID-19): Open Issues, Challenges, and the Road Ahead. In: Tyagi A.K., Abraham A., Kaklauskas A. (eds) Intelligent Interactive Multimedia Systems for e-Healthcare Applications. Springer, Singapore. https://doi.org/10.1007/978-981-16-6542-4_22
64. Mishra S., Tyagi A.K. (2022) The Role of Machine Learning Techniques in Internet of Things-Based Cloud Applications. In: Pal S., De D., Buyya R. (eds) Artificial Intelligence-based Internet of Things Systems. Internet of Things (Technology, Communications and Computing). Springer, Cham. https://doi.org/10.1007/978-3-030-87059-1_4
65. Nair, Meghna Manoj; Tyagi, Amit Kumar "Privacy: History, Statistics, Policy, Laws, Preservation and Threat Analysis", Journal of Information Assurance & Security. 2021, Vol. 16 Issue 1, p24-34. 11p.