A novel block cipher for enhancing data security in healthcare internet of things

R Shantha Mary Joshitta¹ and L Arockiam²

¹ Research scholar, Department of Computer Science, St. Joseph’s College, Trichy -2, Tamilnadu, India.
² Associate Professor, Department of Computer Science, St. Joseph’s College, Trichy Tamilnadu, India.

Abstract. The modern technologies and their contributions have made monitoring and communication possible from everything around us with each other with no or little human intervention. Experts named these connected devices as Internet of Things (IoT), or the Internet of Everything which is considered as the next industrial revolution. This open network posts quite good number of security challenges because of the nature of the IoT environment. Thus, this paper proposes a lightweight block cipher called, JAC_Jo, to enhance the security of data transmitted in the healthcare environment with reduced hardware footprints. The master key for data encryption is derived using soft set based key generation algorithm SoftKeyGen() and encrypts 32-bit data using 64 bit key. The proposed block cipher occupies 2109 GEs of area and requires 360.05mW power without compromising the security. The FPGA implementation of the proposed block cipher is presented and the performance of the block cipher is evaluated using Atmel STK-600 device with 128 kb app / boot memory. The security evolution is presented with respect to differential, linear, slide and algebraic attacks.

1. Introduction
Connecting things, objects and devices with internet is emerging today to link people, processes and things to create a “Connected World”. It is more pertinent and useful than ever before to collect data and use them to build a smarter life driven by these data. It inter-relates many objects such as actuators, sensors, RFID tags, mobile phones and even any digital machines with people which offer the competency to transmit data over a network. It is estimated that IoT market is expected to grow 267 billion US dollars and 30 billion devices by 2020 with the worth of almost $9 trillion [1]. However, these connected devices post diverse challenges and create concerns over memory space, power consumption, battery power, security, performance and cost [2], [3]. In this paper, a state-of-art lightweight cryptographic block cipher, JAC_Jo, is proposed for transferring data securely in the healthcare Internet of Things (IoT) environment. It satisfies the philosophy of lightweight cryptography specified in literature [4] and offers higher level of security to the data. The proposed block cipher is executed using a python script and execution time is compared with the other two standard lightweight block ciphers Simon and Speck and Simeck. And the FPGA implementation result of the proposed block cipher is compared and the comparison results show that the proposed block cipher has higher throughput with requires less space and power consumption [5].

This paper is organized as follows. Section 2 presents the related research works and section 3 elucidates the motivation for designing a block cipher. Objectives of this research are presented in Section 4 and Section 5 explains the design specifications, round function and key schedule of the proposed block cipher. Results are discussed in Section 6. Finally, conclusion is given in Section 7.

2. Related Research Works
Saurabh Singh et al., discussed the primitives of lightweight cryptography [6]. The authors analyzed many lightweight cryptographic algorithms in terms of their number of rounds, key size, structures and
block size. The security architecture, research issues, challenges and solutions were outlined. Security scheme and its improvement for resource constrained IoT devices were explained with a service scenario.

Gang qiang Yang et al., introduced Simeck family of block ciphers with its design components [7]. The researchers had attained 505 Gate Equivalents before the Place and Route cycle and 549 GEs after the Place and Route cycle for Simeck 32 / 64 block cipher. They calculated the power consumption and GE with CMOS 130nm and 65nm ASIC for before and after the Place and Route phase. Furthermore, the authors evaluated the security of Simeck with many other traditional cryptanalysis methods and differential, slide, meet-in-the-middle, impossible differential and linear attacks.

In [8], Ray Beaulieu et al., summarized the Simon and Speck lightweight block cipher algorithm. These block ciphers were developed for securing applications in any constrained environment where AES was inappropriate. These researchers presented the design rationale along with their analysis and implementation results. Experimental results were also demonstrated for the existing software and hardware platforms.

Alkhzaimi et al., presented a cryptanalytic research on Simon family of block ciphers [9]. It provided the initial and final results of the proposed block cipher family. Simon construction was experimented several times and evaluated for classical differential and truncated impossible differential attacks. Reduced-round versions of all Simon variants were tested for differential attacks and were simulated for impossible differential attacks. The analysis on weak key classes and rotational cryptanalysis were also presented.

The differential properties of the Simon and Speck lightweight block ciphers were presented by Biryukov et al., [10]. The authors prolonged the search technique for the differential properties and enhanced the formerly reported differentials on Simon32, Simon48 and Simon64. The researchers exploited the strong differential effect of Simon more effectively.

3. Motivation
As the wings of IoT spread in many fold, there comes many risks and challenges in handling huge amounts of data. The processing power, energy consumption and encryption / decryption of data before / after transmitting are also of major concern [11], [12]. This creates a mounting demand for the use of appropriate cryptographic solution into the embedded devices of IoT applications. In general, these embedded devices are resource-constrained in terms of computation power, battery life, size, memory and power supply. Hence, cryptographic algorithm like RSA with larger key size and complex operations cannot be implemented in these devices. At the same time, the hardware footprints of an algorithm also have to be considered while designing a security algorithm for the resource-constrained devices. Moreover, the tight constrains of smart applications impede the performance of security mechanism and its other functionalities too. Security vulnerabilities such as Denial of Service (DoS) / Distributed Denial of Service (DDoS), eavesdropping, reply attacks and many more are also already explored by many researchers [13]. Therefore, any security algorithm that is designed for IoT applications must have maximum data complexity and active S-boxes to provide optimum security with less Gate Equivalences (GE), memory size, robust design of permutation layer and power dissipation.

4. Objective
The basic design rationale of the lightweight block cipher algorithm for resource-constrained platform is, having not only very small hardware implementations but also software implementations on low-power microcontrollers with minimum flash and SRAM utilization. The software specific metrics such as number of register in RAM, ROM and the hardware specific metrics like gate area and flexibility also are to be given priority. Hence, the primary aim of this paper is to design a lightweight block cipher for data security in healthcare IoT environment. It is built on the base line of the Simeck
lightweight block cipher with better performance in terms of power consumption, waiting time or latency and throughput without sacrificing security, data trustworthiness and protection.

5. The Proposed JAC_Jo Block Cipher

The design specifications and rationales are given in this section. The notations used in the definition of the JAC_Jo block cipher are listed in Table 1.

| Notations | Usages |
|-----------|--------|
| n         | A Word size |
| 2n        | A Block size |
| T         | Number of Rounds |
| PT        | 32-bit output plain text block |
| CT        | 32-bit output cipher text block |
| k_i       | 16-bit Round sub-key for round i |
| K         | Master key generated by SoftKeyGen algorithm [7] |
| F_1, F_2 & F_3 | Intermediate Functions |
| RC_i      | Round constant of round i |
| ⊕         | Bitwise exclusive OR operation |
| <<<n      | Left cyclic shift by n bits |
| ○         | Bitwise AND |
| ||         | Concatenation of two strings |
| ↔         | Swapping Operator |
| X_i       | LSB of the 2n-bit input of the i\textsuperscript{th} round, i=0,1,2,...,T-1 |
| X_{i+1}   | MSB of the 2n-bit input of the i\textsuperscript{th} round, i=0,1,2,...,T-1 |
| F_{X_i}   | 2n-bit intermediate output of the round i, i=0,1,2,...,T-1 |

5.1 Design Specifications of JAC_Jo Block Cipher

Though the Simeck lightweight block cipher is absolutely designed for IoT environment, there are rooms for enhancement in terms of decreasing hardware footprints and improving the security of the key schedule. The round function of Simeck is boosted for hardware enactment and its key schedule algorithm is optimized for software execution. The Simeck block ciphers are implemented in two dissimilar hardware architectures namely, parallel and fully serialized architecture. In the parallel architecture, the round function of Simeck is performed using three gates such as 2n flip-flops, one n-bit width 2-to-1 multiplexer and a combinational circuit to calculate the feedback data for the multiplexer. This combinational circuit comprises of three n-bit XOR gates, one n-bit AND gate and two shift modules [7]. Moreover, the key schedule of this block cipher uses one input combinational circuit and four n-bit key blocks. In fully serialized architecture, two more multiplexers are further added to choose the cyclic shift inputs. Moreover, in the proposed JAC_Jo block cipher, hardware footprints of the ciphers are reduced by substituting two two-input XOR gates with a single three-input XOR gate and removing a 2-to-1 MUX which is used for circular shift operations without damaging the productivity of the existing algorithm.

5.2 Round Function and Key Schedule of the JAC_Jo Algorithm

The 32-bit input plaintext is divided into two halves PT\textsuperscript{L} and PT\textsuperscript{R} where every slot holds 16-bit data. The round function F has three actions namely F_1, F_2 and F_3. F_1 executes left circular shift by 5-bit on PL\textsuperscript{L} and F_2 will execute bitwise AND operation on F_1 and PL\textsuperscript{L}. F_3 holds the result of XOR operation on PT\textsuperscript{R} and F_2. F_3 performs the left circular shift by 1-bit of PL\textsuperscript{L}. Again this result is XOR with F_3 and the round key K_i which is created by key scheduling algorithm. Totally sixteen diverse keys are generated from the 64-bit key scheduling algorithm and these keys are applied at each round of the
JAC_Jo cipher. The cipher text will be composed at the end of round 16. The algorithm for computing round function of the JAC_Jo cipher is depicted in Figure 1.

**Algorithm JAC_Jo**

**Input:** Plain Text : $PT_{T-1}^{31} PT_{T-1}^{30} \ldots \ldots \ldots PT_1^2 PT_1^1 PT_0^0$

**Output:** Cipher Text : $CT_{T-1}^{31} CT_{T-1}^{30} \ldots \ldots \ldots CT_1 CT_1 CT_0^0$

for $i = 0$ to $T - 1$

\[ F_i \leftarrow ((\lll{5} PT_i)) \]

\[ F_2 = F_1 \oplus PT_i \]

\[ F_x = F_2 \oplus PT_{R_i} \]

\[ F_3 \leftarrow ((\lll{1} PT_i)) \]

\[ PT_{R_{i+1}} \leftarrow F_x \oplus F_3 \oplus k_i \]

\[ PT_{L_{i+1}} \leftarrow PT_i \]

\[ PT_{L_i} \leftarrow PT_{R_i} \]

end

CT $\leftarrow PT_{L_{T-1}} || PT_{R_{T-1}}$

return CT

**Figure 1.** Algorithm JAC_Jo.

The block diagram of the round function and key schedule of the JAC_Jo cipher is depicted in Figure 2.

**Figure 2.** Round Function and Key Schedule of the JAC_Jo Block Cipher.

5.3 Procedure of JAC_Jo Algorithm

The procedure of the proposed JAC_Jo algorithm is presented in Figure 3.
5.4 The Encryption and Decryption Cycle

The encryption and decryption cycle of the proposed JAC_Jo algorithm is clearly explained in the forthcoming section.

5.4.1 Encryption Cycle

Thirty two bit input plaintext is alienated into two 16-bit plaintexts, PT\text{L}_i and PT\text{R}_i. PT\text{R}_i comprises of Least Significant Bits (LSB) 16-bits and PT\text{L}_i is of Most Significant Bits (MSB) 16-bits.

\[
\text{PT} \leftarrow \text{PT}_i \oplus \text{PT}_i
\]

1. Perform Left cyclic shift by 5 bits on PT\text{L}_i
   \[
   F_1 \leftarrow (\lll{5}(\text{PT}_i))
   \]
2. Perform Bitwise AND on F_1 and PT\text{L}_i
   \[
   F_2 = F_1 \land \text{PT}_i
   \]
3. Perform XOR operation with F_2 and PT\text{R}_i
   \[
   F_3 = F_2 \oplus \text{PT}_i
   \]
4. Perform Left cyclic shift by 1 bit on PT\text{L}_i
   \[
   F_4 \leftarrow (\lll{1}(\text{PT}_i))
   \]
5. Perform XOR with Fx, F_3 and Key k_i
   \[
   \text{PT}_{i+1} = F_3 \oplus F_4 \oplus k_i
   \]
6. Swap PT\text{R}_i with PT\text{L}_i
   \[
   \text{PT}_i \leftrightarrow \text{PT}_{i+1}
   \]
7. Repeat Step 1 through 6 for T times where T = 0, 1, 2,...15
8. Concatenate PT\text{L}_{15} and PT\text{R}_{15} to obtain the required cipher text CT.
   \[
   CT \leftarrow \text{PT}_{15} \oplus \text{PT}_{15}
   \]

5.4.2 Decryption Cycle

Thirty two bit input cipher text CT is divided into two 16-bit cipher texts, CT\text{L}_i and CT\text{R}_i. CT\text{R}_i consists of LSB 16-bits and CT\text{L}_i is of MSB 16-bits.

\[
\text{CT} \leftarrow \text{CT}_i \oplus \text{CT}_i
\]

1. Perform Left cyclic shift by 1 on CT\text{R}_i
   \[
   F_3 \leftarrow (\lll{1}(\text{CT}_i))
   \]
2. Apply XOR with CT\text{L}_i, F_3 and k_i
   \[
   F_4 = \text{CT}_i \oplus k_i \oplus F_3
   \]
3. Perform Left cyclic shift by 5 on CT\text{R}_i
   \[
   F_1 \leftarrow (\lll{5}(\text{CT}_i))
   \]
4. Perform Bitwise AND on F_1 and (CT\text{R}_i)
   \[
   F_2 = F_1 \land (\text{CT}_i)
   \]

**Figure 3.** Procedure of the JAC_Jo.
5. Apply XOR on $F_2$ and $F_X$: $CT_i \leftarrow F_2 \oplus F_X$

6. Swap $CT_i^R$ with $CT_i^L$: $CT_i^R \leftrightarrow CT_i^L$

7. Repeat Step 1 through 6 for $T$ times where $T = 0, 1, 2, ..., 15$

8. Concatenate $CT_{15}^L$ and $CT_{15}^R$ to obtain the required plain text $PT$

$$PT \leftarrow CT_{15}^L \| CT_{15}^R$$

### 5.5 Key Scheduling

Key scheduling process of the JAC_Jo block cipher is driven from the Simeck cipher. It produces 16 sub-keys of each size 16. The round key $k_i$ is generated from the master key $K$ which is generated by SoftKeyGen algorithm [14]. The proposed SoftKeyGen algorithm uses theory of soft sets to generate the master key $K$. Later, $K$ is segmented into four words and is given as input to four opening state variables ($t_2, t_1, t_0, k_i$). The round constant $C \oplus (z_i)$ and the round key $R_{C \oplus (z_i)}$ of Simon block ciphers are reused in JAC_Jo block cipher with boosted master key $K$. The variables and the round key $k_i$ will be updated using the following operations.

$$k_{i+1} = t_i$$

$$t_{i+3} = k_i \oplus f(t_i) \oplus C \oplus (Z_i)$$

Where $0 \leq i \leq T-1$.

$k_i$ is the round key for the $i$-th round and the value of $C$ is calculated by $C=2^n - 4$. Here, $n$ is the word size and $(z_i)$ represents the $i$-th bit of the sequence $z_i$. The other JAC_Jo family block ciphers can also use the same sequence $Z_i$ only. For JAC_Jo 32/64 cipher uses m-sequence with period 31. If the number of round is greater than 31, then the sequence starts repeating.

### 6. Result and Discussion

#### 6.1 Performance Analysis

The proposed lightweight block cipher is calculated for the time taken to perform the encryption and decryption operations [15]. The run time of encryption operation is calculated using a python script designed for this block cipher and compared with the existing Simeck and Simon block ciphers with 32-bit key size and 64-bit block size. The python script compared the messages of varying sizes. Messages of 32, 64, 128, 256 and 512 bits are considered for execution and the time taken to encrypt these messages are calculated for the Simon, Simeck and the proposed JAC_Jo. The execution window of the python script is given in Figure 4.

**Figure 4.** Execution of Python Script of the JAC_Jo.
The analysis of the result is presented in Figure 5.

![Figure 5](image)

**Figure 5.** Execution time of Encryption operation.

In Figure 5, it is evident that the JAC_Jo is faster than the existing block ciphers such as Simon and Simeck. Even though the time taken for performing encryption operation of 32 and 64 bits messages do not vary much, but it makes considerable difference in 512 bits of messages. This difference will increase if the message size is high. Execution time of decryption operation is presented in Figure 6.

![Figure 6](image)

**Figure 6.** Execution time of Decryption operation.
The JAC_Jo deception algorithm runs faster than the other two algorithms. It is clearly presented in Figure 6.

6.2 Analysis for proving Lightweight phenomena
A cryptographic algorithm can be said to be light weighted if it requires less memory, smaller gate area and energy. But the throughput has to be higher [16]. The memory consumption and the implementation size are coupled together into its gate area and are measured in Gate Equivalents (GE). Throughput is the processing time of an algorithm. It is calculated by the amount of plaintext processed per time unit and measured in bits or bytes per second. Power consumption is measured by the amount of power needed to use the circuit.

Simeck block cipher is designed to have a smaller area because of the simplified key schedule, the simplified linear-feedback shift register (LFSR) for generating key constant, and the decreased shift numbers in the round function. But JAC_Jo block cipher reduces a XOR gate used in round function and a multiplexer in key schedule which in turn reduces the Gate Equivalence of the block cipher. The number of gates required for the construction of the combinational circuit in the key schedule of Simon, Simeck and JAC_Jo in the parallel architecture are listed in Table 2.

Table 2. Comparison of Gates used in JAC_Jo Algorithm with other block ciphers.

| Gate Type | Simon | Simeck | JAC_Jo |
|-----------|-------|--------|--------|
| XOR       | 2n+1  | n+1    | N      |
| XNOR      | n-1   | n-1    | n-1    |
| AND       | -     | n      | N      |

The experimental analysis for lightweight cryptography was conducted using Atmel STK-600 device with 128 KB app / boot memory, 4 KB data memory and 4 KB EEPROM. The comparative result of the block ciphers Simon 32/64, Simeck 32/64 and JAC_Jo 32/64 is presented in Table 3.

Table 3. Comparison of FPGA implementation results of Simon 32/64, Simeck 32/64 and JAC_Jo 32/64.

| Block Ciphers | Area [GE] | Latency [Cycle] | Throughput [Mbps] | Power [mW] |
|---------------|-----------|-----------------|-------------------|------------|
| Simon 32/64 [18] | 2612      | 4045            | 3.60              | 443.1      |
| Simeck 32/64 [17, 19] | 2345      | 3729            | 5.60              | 417.58     |
| JAC_Jo 32/64 | 2109      | 3499            | 7.93              | 360.05     |

From Table III, it is proved that the proposed JAC_Jo lightweight block cipher is in line with the other standard lightweight block ciphers and requires less area and power. And also the block cipher offers higher throughput than the other ciphers.

6.3 Security Analysis
Cryptanalytic attacks are evaluated based on the parameters such as space, time and data complexity of the proposed block cipher [20]. Space complexity is nothing but the amount of memory required to save internal data during the execution of the block cipher whereas data complexity is the quantity of plaintext and/or ciphertext pairs essential for executing an attack. Time complexity is calculated with the computation time needed to perform the attack. The attacks that could appear on a symmetric block cipher and mechanism to avoid such attacks in JAC_Jo block ciphers are explained below.

6.3.1 Differential and Linear Attacks: The differences in the plain and cipher texts are analyzed by the attacker to trace the plain text from the known plain texts. But the proposed JAC_Jo uses
shift operations in its round function and key schedule to harden the differential and linear parameters of the cipher. Though the reduced round function of JAC_Jo is vulnerable to attack but the diffusion operation has the success probability of 0.997 [21]. But it is difficult to attack the full round function. Thus the proposed cipher is sufficiently secured against differential and linear attacks. Differential attack on the proposed JAC_Jo block cipher is compared based on the space, data and time complexities against various Lightweight Block Ciphers and are summarized in Table 4.

Table 4. Comparison of space, data and time complexities on Differential Attack.

| Block cipher   | Number of rounds attacked | Complexity |
|----------------|---------------------------|------------|
|                |                           | Space     | Data    | Time    |
| Speck 128/128  | 17                        | $2^{22}$  | $2^{113}$ | $2^{113}$ |
| Simon 128/128  | 48                        | $2^{24}$  | $2^{124.796}$ | $2^{120.474}$ |
| Simeck 128/128 | 19                        | $2^{31}$  | $2^{141}$ | $2^{115}$ |
| JAC_Jo 128/128 | 27                        | $2^{33}$  | $2^{153}$ | $2^{117}$ |

6.3.2 **Slide Channel Attack:** It is an attack of making the cipher to be weaker by increasing the number of rounds. It analyses the key schedule to exploit the weaknesses of it by repeating the keys in a cyclic manner [26]. But efficient designing of round constant and key schedule normally reduces the slide channel attack. In the proposed block cipher, key schedule is enhanced with the application of soft set theory and helps in preventing any kind of slide attack. Thus, the JAC_Jo block cipher is resistant against slide channel attack. A comparison of JAC_Jo with other block ciphers in terms of space, data and time complexities are presented in Table 5.

Table 5. Comparison of space, data and time complexities on Slide Channel Attack.

| Block cipher   | Number of rounds attacked | Complexity |
|----------------|---------------------------|------------|
|                |                           | Space     | Data    | Time    |
| Speck 128/128  | Full Round                | $2^{15}$  | $2^{127}$ | $2^{58}$ |
| Simon 128/128  | Full Round                | $2^{27}$  | $2^{101.96}$ | $2^{48}$ |
| Simeck 128/128 | Full Round                | $2^{24}$  | $2^{79.54}$ | $2^{48}$ |
| JAC_Jo 128/128 | Full Round                | $2^{19}$  | $2^{63.58}$ | $2^{56}$ |

6.3.3 **Algebraic Attack:** In this attack, the attacker may use any algebraic system to break the cipher [29]. He considers the text as a vector of bits and uses any Boolean algebra such as arithmetic modulo 28. For another case, he may consider the same text as a vector of bytes and performs some other Boolean system of equations. But in any algebra, solving a non-linear equation is quite harder and difficult to break. In the proposed JAC_Jo cipher, the equation for round function is the combination of arithmetic and logical operations and cannot be easily described with linear equations in Boolean algebra. This makes the algebraic degree of the cipher to be stronger against this higher-order differential attack.

From the above Table 4 and 5, Even though JAC_Jo block cipher uses very simple round function, it offer higher order security because of its large number of rounds. It is a good choice of block cipher where energy, area and security are given higher priority other than latency.

7. **Conclusion**
This paper presents a lightweight block cipher JAC_Jo with its design specifications and rationales. It is designed to enrich the data security of the IoT data collected from the resource constrained devices used in the IoT environment. JAC_Jo is designed on the base line of the Simeck block cipher and modifies
its hardware implementation. It enhances Simon key schedule with a master key generated using soft set theory. Even though JAC_Jo requires less computation power and memory size with small key size, but it does not compromise the security level. The master key generation algorithm enhances the security and makes the key guessing and dictionary attacks impossible. Moreover, the lightweight characteristics in terms of area, energy and throughput are also proved by the researchers.

REFERENCES

[1] Gartner 2013 https://www.gartner.com/newsroom/id/2636073
[2] Saurabh Singh, Pradip Kumar Sharma, Seo Yeon Moon and Jong Hyuk Park 2017 Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions Journal of Ambient Intelligence and Humanized Computing 1–18.
[3] S. Santiago and L. Arockiam 2016 Energy Efficiency in Internet of Things: An Overview International Journal of Recent Trends in Engineering & Research (IJRTER) 2 475-482.
[4] Tarun Kumar Goyal and Vineet Sahula 2016 Lightweight security algorithm for low power IoT devices Proc. of International Conference on Advances in Computing, Communications and Informatics (ICACCI) 1-5. DOI: 10.1109/ICACCI.2016.7732296
[5] Bansod G, Raval, N. and Pisharoty N 2015 Implementation of a new lightweight encryption design for embedded security IEEE Transactions on Information Forensics and Security, 10 1 142-151.
[6] Saurabh Singh, Pradip Kumar Sharma, Seo Yeon Moon and Jong Hyuk Park 2017 Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions Journal of Ambient Intelligent Human Computing (Springer) 1-18.
[7] Gang qiang Yang, Bo Zhu, Valentin Suder, Mark D. Aagaard and Guang Gong 2015 The Simeck Family of Lightweight Block Ciphers Proceeding of CHES-2015 Lecture Notes in Computer Science, 9293. Springer, Berlin, Heidelberg 307-329.
[8] 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 Proc. of NIST Lightweight Cryptography Workshop 1-15.
[9] Alkhzaimi H, & Lauridsen M 2013 Cryptanalysis of the SIMON Family of Block Ciphers IACR Cryptology ePrint Archive 1-26.
[10] Biryukov A. Roy A. and Velichkov V 2015 Differential analysis of block ciphers Simon and Speck Lecture Notes in Computer Science 8540 546–570.
[11] Shantha Mary Joshiata R and Arockiam L 2017 Secure Two-Tier User Authentication Mechanism For IoT Enabled Smart Healthcare System International Journal of Recent Scientifc Research, 8, 7 1825-1826.
[12] N. Veeararagavan, Dr. L.Arockiam and Dr. S. S. Manikandasaran 2017 Enhanced Encryption Algorithm (EEA) for Protecting Users’ Credentials in Public Cloud IEEE sponsored 2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies 1-6.
[13] S. S. Manikandasaran and Dr L Arockiam 2016 Security Attacks and Cryptography Solutions for Data Stored in Public Cloud Storage International Journal of Computer Science and Information Technology & Security (IJCITS) 6 498 - 503.
[14] Shantha Mary Joshiata R and Arockiam L 2018 Key Generation Algorithm using Soft Set for data security in Internet of Things Proc. of the 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT) ELSEVIER 1-6.
[15] Pereira G. C. C. F. Alves R. C. A. da Silva F. L. Azevedo R. M Aliberti B. C. and Margi C. B. 2017 Performance evaluation of cryptographic algorithms over IoT platforms and operating system Security and Communication Networks 1-16. DOI: https://doi.org/10.1155/2017/2046735
[16] Biryukov A. and Perrin L. 2017 State of the Art in Lightweight Symmetric Cryptography IACR Cryptology ePrint Archive 1-55
[17] Bo Zhu et al. 2015 The implementations of the Simeck Family of Block Ciphers Available: https://github.com/bozhu/Simeck

10
[18] Park T. Seo H. Bae B. and Kim H. 2016 Efficient Implementation of Simeck Family Block Cipher on 8-Bit Processor 14 3 177–183.
[19] Okabe T 2017 FPGA Implementation and Evaluation of lightweight block cipher BORON 5 1 207–216.
[20] Sadhuakhan, R. Patranabis S. Ghoshal A. et al. 2017 An Evaluation of Lightweight Block Ciphers for Resource-Constrained Applications: Area, Performance, and Security Journal of Hardware and Systems Security 1 3, 203–218 https://doi.org/10.1007/s41635-017-0021-2
[21] Kexin Qiao Lei Hu and Siwei Sun 2015 Differential Analysis on Simeck and SIMON with Dynamic Key-guessing Technique Springer 1-20.
[22] Dinur I 2014 Improved differential cryptanalysis of round reduced speck. In: Selected areas in cryptography 8781 of LNCS. Springer 147–164
[23] Abed F, List E, Lucks S and Wenzel J 2013 Cryptanalysis of the speck family of block ciphers. IACR Cryptology ePrint Archive 2013:568
[24] AlKhzaimi H and Lauridsen MM 2013 Cryptanalysis of the SIMON family of block ciphers. IACR Cryptology ePrint Archive 2013:543
[25] S. Kolbl and A. Roy 2015 A Brief Comparison of SIMON and Simeck Cryptology ePrint Archive, Report 2015/706.
[26] Albrecht M. R. Driessen B. Kavun E.B. Leander G. Paar C. and Yalçın T. 2014 Block Ciphers – Focus on the Linear Layer (feat. PRIDE) Advances in Cryptology – CRYPTO 2014. Lecture Notes in Computer Science, Springer, 8616 57-76. DOI: 10.1007/978-3-662-44371-2_4
[27] Annelie Heuser, Stjepan Picek, Sylvain Guilley and Nele Mentens 2016 Side-channel Analysis of Lightweight Ciphers: Does Lightweight Equal Easy?. RFIDSec 2016: 12th Workshop on RFID and IoT Security, Nov 2016, Hong Kong, Hong Kong SAR China
[28] Guilley S., Heuser A and Rioul O. 2015 A Key to Success - Success Exponents for Side-Channel Distinguishers. In Biryukov, A., Goyal, V., eds.: Progress in Cryptology Proceedings. Volume 9462 of Lecture Notes in Computer Science, Springer (2015) 270{290
[29] Arockiam L and Monikandan S 2015 AROMO security framework to enhance security of data in public cloud International Journal of Applied Engineering Research 10 9 6740–6746.