Energy Optimized Simon Lightweight Security Algorithm for Internet of Medical Things (IoMT)

Somasundaram R, Mythili Thirugnanam

Abstract*: *Internet of Things (IoT)is a fast-growing technology which connects and communicates among things using the internet. Also, generated data are stored as a knowledge base, which helps things to perform automatically without human intervention. IoMT for healthcare applications provides more benefits to patients affected by diseases which required continues health observation. Security issues in IoMT applications cause major problems like allowing an unauthorized person to monitor patient information, treatment procedure, and pharmaceutical detail which makes patients life insecure. Security mechanisms adopted in the general medical devices are may not be suitable for medical IoT because of constrained resource usage such as less energy consumption, less processing power, and less memory usage. In order to secure sensitive data transmission among a lightweight IoMT device communication, this work proposes a new energy optimized lightweight security algorithm. In this work, Modified Simon's (M-Simon) mix column lightweight algorithm is proposed. M-Simon is implemented to reduce the energy consumption of the lightweight medical things using a reduced number of secure round function from 36 to 24 and the sensor data is encrypted with smaller 48-bit key mix column algorithm. Results concluded that the proposed security algorithm reduces higher energy consumption without compromising security in IoMT for healthcare applications.

Index Terms*: *Healthcare applications, Internet of medical things, Lightweight security, Medical devices, Security issues.

I. INTRODUCTION

Tremendous improvement of internet protocols, wireless communications and the development of different sensor devices have created an opportunity to improve wireless sensor network (WSN). Nowadays, the sensor network is built with new capabilities like internet-connected sensor devices, GPS based device tracking, analyzing sensor data, and predicting human behavior, etc. Consequently, IoT facilitates these functions with better performance.

IoT supports a different kind of applications. But, still more challenging to develop secure communication between IoMT because of the tradeoff between processing and power consumption. At the same time, mitigating security issues like data leakage, unauthorized data injection, and eavesdropping with traditional cryptography approach is challenging. Due to this, there is a need to use a lightweight security algorithm to overcome the challenges faced in the traditional security approach.

The parameters like data storage security, transmission channel security, intrusion detection, key management, cryptography, and firewall R. Somasundaram. (2018) [15] are key parameters which help to improve the overall security of the IoMTs. The major security issues lie in IoMT domain and several lightweight security approaches are studied. It is observed that most of the lightweight security algorithms do not satisfy the energy requirements of IoMTs. Hence, this work aims to develop a Modified Simons lightweight algorithm as a security solution to IoMT applications. The algorithm focused on securing internet enabled medical devices.

This paper organized in the following manner. In section II related works are discussed. design of lightweight security model for IoMT application is detailed in section III. In section IV, results of the proposed M-Simon algorithm are discussed.

II. RELATED WORKS

A. Review on security issues

There are several research works were carried out in the field of IoMT healthcare security enhancement. In this, very few researchers were implemented in energy efficient security implementation. A new lightweight key distribution protocol for resource-constrained IoMT sensors presented in M. A. Iqbal. (2016, August) [1] to improve authentication and key agreement. Proposed protocol mainly focused on energy resource constraints. Though, energy consumption (236µ for 80-byte transaction) of the this proposed lightweight key distribution protocol rate is not suitable for lightweight devices like IMDs, which operates on very low battery power. M. Haghi. (2017) [2] reviewed IMD authentication and Security attacks. The review reveals that IMDs are vulnerable to DoS attack which exploits the device's processor, communications, or battery. M. A. Bahnasawi. (2016, December) [3] reviewed AES, 3DES, two fish and RSA algorithms for IoT Applications.

Revised Manuscript Received on June 05, 2019.

Somasundaram R, SCOPE, Vellore Institute of Technology, Vellore, India.

Mythili Thirugnanam, SCOPE, Vellore Institute of Technology, Vellore, India.

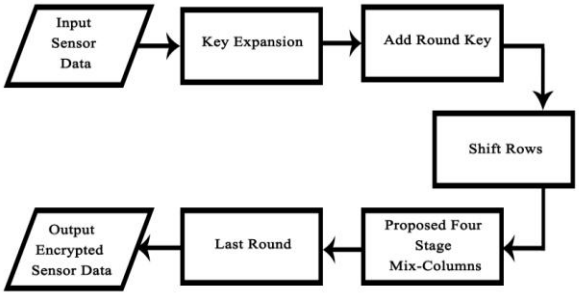


Fig 1: Steps involved in Mix Column Algorithm

Based on the review, the performance of AES algorithm implementation in RFID data encryption concludes that the requirement of 99 cycles for 128 bits keys will consumes more power. M. Haghi. (2017). [2] identified the IoMT security issues in the wearable devices like personal data leakage such as patient tracking information and medication management. S. Feizi. (2014, October) [4] presented Simon block cipher algorithm for RFID tags. In Simon 32 bit key and 64 bit data, the key size and data size are very small also, this algorithm is suitable to implement in lightweight device security. Implementation results conclude that data length is 32 bits when compared to HB algorithm occupies 278 bits and X-TEA occupies 133 bits. Implementation result concludes Simon's algorithm is more appropriate to RFID devices. But results are not suitable for passive RFID tags like implanted tag's energy consumption issues. P. A. Wortman. (2017, February) [5] conducted a study on device level security vulnerabilities like weak antenna coverage and radio attacks. also, recommended device standardization to overcome security vulnerabilities K. A. McKay. (2017) [16]. Based on the review on security issues, resource constraints like limited battery consumption, energy consumption problem in lightweight distributed key agreement protocol, vulnerabilities present in IMDs which prone to DoS attack, power consumption issues in AES implementation to RFID and implantable devices, and energy consumption issues in Simon's algorithm for passive RFID tags are still challenging to provide better security. With this intention a new security mechanism is proposed to address the energy consumption issue without compromising any security attacks.

B. Review on lightweight algorithms

S. S. S. Priya. (2015, March) [6] implemented a AES lightweight algorithm to mitigate security issues in wireless sensor networks. Using AES parallelism techniques this work achieves the overall throughput of the encryption with the help of modified mix column algorithm. This eight stages parallelism in mix-column shows 60.47 Gbps throughput, 2.117 ns delay, 3168 no of slices. The proposed design requires less area and producing high throughput compared to existing techniques. R. Beaulieu. (2015) [13] F. M. Nascimento. (2010) [7] implemented lightweight devices and RFID Tags- HIGHT (HIGH security and lightweight algorithm. Simulation results conclude that the proposed algorithm's clock cycle is 88.0 ns. This is 20.0ns faster than the previous version. And delay is 9.8ns which is 3.9ns faster than the previous version, Algorithm is more significant to RFID tags. Y. A. Abbas. (2014, November) [8] implemented lightweight PRINCE algorithm, which works based on block cipher technique in FPGA for lightweight devices (Smart cards, RFIDs, etc.). Proposed algorithm is still challenging to solve DoS and hijacking attack because of communication overhead. M. R. Z'aba. (2014) [9] presented a 64-bit I-PRESENTTM block cipher algorithm for RFID tags and WSNs. Results of the proposed algorithm support a 128-bit larger key size which requires more energy and computation power.

On the other hand, Simulating and crypt-analyzing I-PRESENTTM is a challenging task. M. Usman. (2017) [10] proposed a Secure IoT (SIT) lightweight security algorithm, the execution time of the algorithm is 0.188 milliseconds for encryption and 0.187 milliseconds for decryption. The algorithm utilizes the 22 bytes of memory on the Arduino (ATmega 328) platform. the proposed algorithm is suitable for lightweight devices to compare to existing algorithms. D. Aakash. (2016) [11] proposed a new lightweight security algorithm named PRESENT-GRP for IoT lightweight devices. The proposed algorithm implemented in a single node (Intel Galileo Gen 2) and execution time is taken for the cryptographic operation of data communication concludes that the proposed PRESENT-GRP algorithm is suitable to lightweight medical devices. S. R. Chatterjee. (2014, December) [12] proposed lightweight devices in WSN. The results conclude that the pipelined Bluefish algorithm performance better in terms of throughput and less delay.

Based on the review of various lightweight security algorithms concludes that less throughput, vulnerable to DoS attack, exposed to hijacking attack, Compatibility issues to implement on various devices, and the tradeoff between performance and energy.

III. PROPOSED M-SIMONS ALGORITHM

A new modified Simons lightweight algorithm is proposed in order to develop an energy efficient secure IoMT. Positives in AES mix-column like high throughput, less delay, and low power consumption. As well as, the smaller key length used in Simon's block cipher algorithm is more appropriate to IoT medical device security.

In view of this merits, this work aims to introduce a new modified Simon's lightweight algorithm for lightweight healthcare applications.

A. Simon's algorithm

Simon is encryption algorithm based on block cipher technique developed by

the National Security Agency (NSA). This algorithm was designed to give improved security to the energy constrained devices with a feistel structure which includes AND (x) operation, logical XOR (⊕) round functions. The security of the message guaranteed by implementing different key sizes to the different sizes of message blocks D. Halperin. (2008) [14].

Various key sizes for the respective block sizes are given in Table 1. In Simon's algorithm, word is n-bit (n must be 16, 24, 32, 48, or 64), and 2n size block with the key size m is a multiple of n (m must be 2, 3 or 4). Will form a Simon2n/nm cipher implementation. For instance, A 64-bit message block (n = 32) that uses a 128-bit key size will form a Simon64/128 cipher. The key generation logic is depending on the implementation of 2,3, or 4.

B. Round Function

The classical Feistel structured round function (RF) is shown in Figure 1. the round function RF is constructed based on the basic Feistel structure operations such as roll left bits by 1,2 or 8, XOR, AND. key schedule generates the key bit kb_i. to create a single encrypted word the round function works with two blocks in the size of n bits. the logic of the functions based on left rolls, XOR, and AND. This two-stage Feistel cipher includes the mathematical representation of the Feistel map.

$$R_k(x,y) = (y \oplus (S^1 x \wedge S^8 x) \oplus S^2 x \oplus k, x) \quad (1)$$

From the above equation R_k is the Simon's round function. Here k is round key of the round function R_k. In the i^{th} step of the process two sub-cipher (x_i+1, x_i) for the round function ns Rk(i). To strengthen the round key generation this, work attempts to incorporate AES mix column operation with the existing Simon's algorithm.

C. Key Schedule

Unlike the round function, the key schedule will not be balanced all the time. In this key schedule process, m is the message and the keyword count is will determine the structure of the key expansion ($m \times n$).

In order to do the bit operation, the keyword-expansion process includes the XOR, constant and right shift operations consist of a right shift, XOR and a constant sequence (zx). here, the constant sequence will eliminate the slide properties, that will help to maintain the balanced scheduling process.

D. Mix-column

AES is a substitution and permutation-based encryption algorithm. It has many iterations in the process which includes the substitution of input with some output. permutation process makes this process more complicated to cryptanalysis the encryption process. Instead of computing the data in bits AES computes all the plaintext data in bytes. In AES, 16 bytes of plaintext data will be modified using mix columns process.

AES mix-columns operation four separate data block with the separate operation. in a single iteration, the throughput of the execution is considerably increased by using the parallelism. the sample mix column calculation for 128 bits of plain text is explained below.

Mix-column algorithm depicted in Fig 1. In the mix-column operation, each column is considered as a four-stage polynomial. In the each and round key addition process has done with the XOR logical operation on data block using the round key. Where this round key is a cryptographic key which is generated during the key schedule process.

E. Modified Simon's Mix Column algorithm

Key Expansion:

Step 1: Begin.

Step 2: Loop throw if $i = m$ to $T-1$.

Step 3: Store if $TEMP \Leftarrow S-3k[i-1]$.

Step 4: Check the condition $m = 6TEMP \Leftarrow TEMP \oplus k[i-5]$.

Step 5: Store $TEMP \Leftarrow TEMP \oplus S-1TEMP$.

Step 6: Calculate $k[i] \Leftarrow k[i-m] \oplus TEMP \oplus z[j][(i-m) \bmod 62] \oplus c$.

Step 7: Exit from the loop.

F. Key Encryption process

In the beginning, plain text data fed into key encryption. Next, the process starts with checking the condition of the i^{th} a step is equal to keywords. Also, the range is the length of the keyword and number of rounds minus one. If the condition satisfies then the process moves into key shifting based on the number of key and number of steps minus one. Also, this value stores into the temporary variable to carry the shift key into a further process.

Table I - Parameters of M-Simon lightweight algorithm.

In step four, process control flow moves to check the condition that the number of words (m) is equal or less than to the (k -number of keys). If condition satisfies, process control moves to mix column. Finally, stored values carry forward to key shifting, mix column and encryption process.

G. Key Schedule

Step 8: Shift Rows.

Step 9: Four parallel mix column operation carried out.

Step 10: Loop throw if $i = 0 to T-1$.

Step 11: Store if $TEMP \Leftarrow x$.

Block size (2n)	Word size (n)	keywords (m)	Key size (mxn)	Rounds (T)
32	16	3	48	24
48	24	3,4	72,96	36
64	32	3,4	96,128	42,44
96	48	2,3a	96,144	52,54
128	64	2,3,4	128,192,256	68,69,72

Step 12: Calculate if
$x \Leftarrow y \oplus (S_x S^8 x) \oplus S^2 x \oplus k[i]$.

Step 13: Check condition if $y \Leftarrow TEMP$.

Step 14: Exit from the loop.

Step 15: Generate cipher text cipher
$Cipher \Leftarrow x \mid y$ (attach x to y).

Stem 16: Stop.

After a key expansion process, data are fed into four stages mix column operation then transfers into an encryption process. Encryption starts with checking the condition of the i^{th} step is equal to keywords (m) as well as the length of the keyword (n) range up to the number of rounds (T-1). If the condition satisfies then the process moves into checking the condition whether sub cyber value(x) is greater than or

Fig 3: Sky Mote

equal to the value stored in the temporary variable. To improve the key strength as well as less energy consumption, the process calculates Simon's round function, to reduce the power consumption number or rounds are reduced from 38 to 24. Finally, the cyphertext has been generated.Parameters of the M-Simon's algorithm is described in Table I. The parameters, key words, size of the key, number of keys, number of rounds, size of the and word size. In this, the block size is twice into the number of key size (n). Key size is the product of a number of keywords (m) into the size of the keyword (n) and number of rounds in the key shifting process is T.

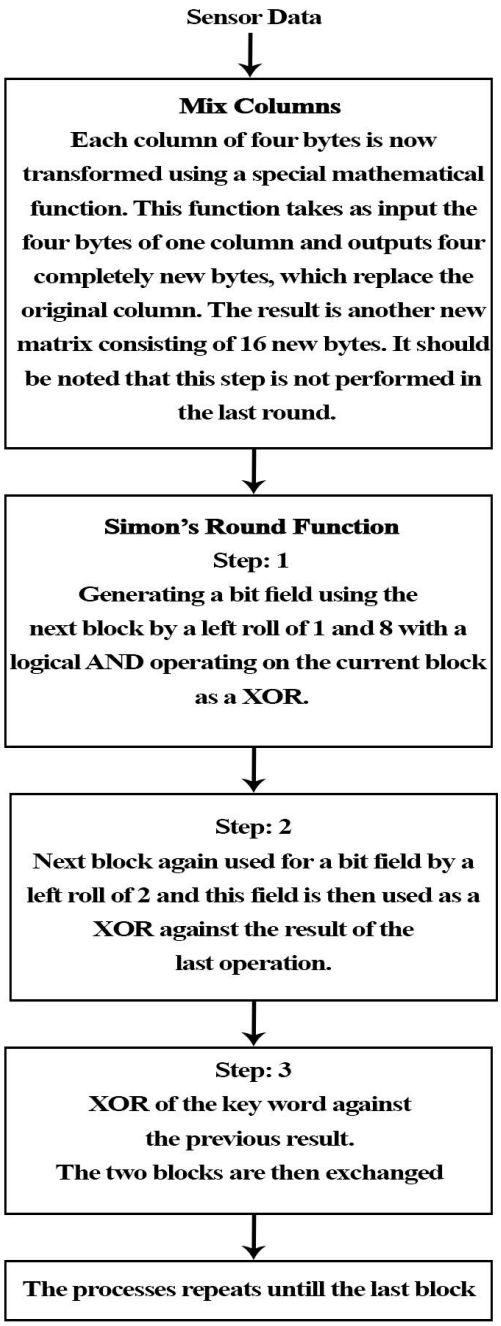
The proposed mix column ciphers algorithm is shown in Figure 2. The process receives sensor data as an input and generates the key. keywords and number of words are taken into every step of Simon's mix column algorithm. Once the mix column process is completed, round keys are generated using key scheduling. Cipher keys generates the round keys and the round key numbers totally depends on key length and block length.

M-Simon's mix column algorithm has been implemented in Cooja simulator available in the Contiki operating system. Testing made with 100 Tmote sky motes.

H. Tmote Sky mote

Tmote Sky is the lightweight WSN module. which provides a verity of sensor supports, low power consumption, and easy code development support. A software-based Tmote Sky is instantly available for Contiki OS operating system in cooja simulator. this software-based module helps to develop and test different types of parameters such as light, temperature, humidity, and various health parameters. The custom code integration in the Tmote Sky firmware helps to test heartbeat and other health parameters.Motes are deployed in the random order. all mote's firmware updated with Simon's encryption algorithm, also encrypted data transmitted between 100 motes across the network. Energy consumption.Power consumption is the major metric in IoT research, most of the energy consumption in the device process is depends upon the encryption process, data transmission, and LPM. the option of the radio transceiver.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication



Simon Mixcolumns algorithm

Fig 2: Proposed M-Simon lightweight algorithm.

NIST Standard. K. A. McKay. (2017) **[16]**	**Proposed Model**
Functionality: Cryptographic algorithm(purpose) (e.g., encryption, message authentication, authenticated encryption scheme, hashing, etc.)	Encryption
Design goals: List design goals.	**Block size:**32 bits **Key size:** 48 bits **Word size:**16 words K**eywords:**3words **Rounds:**24 rounds
Physical characteristics: Name physical characteristic(s), and provide acceptable range(s) (e.g., 64 to 128 bytes of RAM)	64 bytes of RAM
Performance characteristics: Name performance characteristic(s), and provide acceptable range(s) (e.g., latency of no more than 5 ns)	**Transmission Delay:** 10ns
Security characteristics: (e.g., relevant attack models, minimum security strength, side channel resistance requirements, etc)	Minimum security strength
Minimum Power (40mW) per CPU cycle	CPU Power consumption (mW):28.99149 mW

Table II- NIST recommended characteristics of lightweight security evaluation.

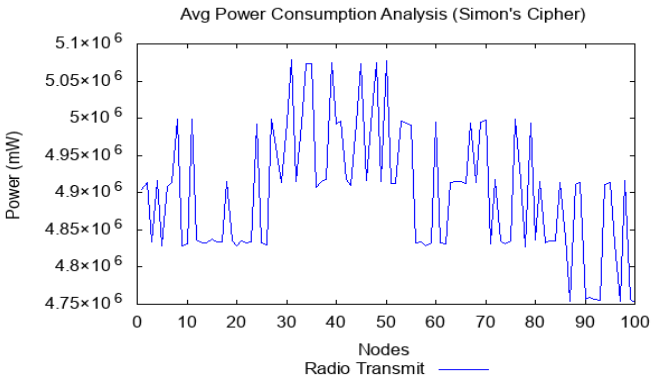


Fig 4: Simon: Radio transmission vs Power consumption

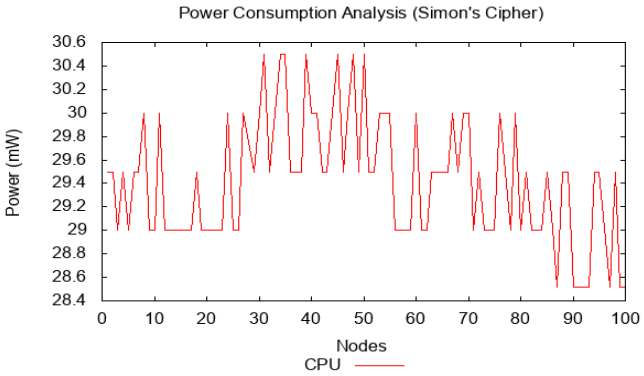


Fig 5: Simon: CPU cycle vs Power consumption

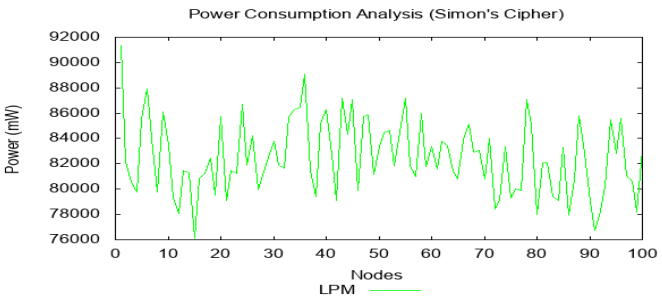


Fig 6: Simon: LPM vs Power Consumption

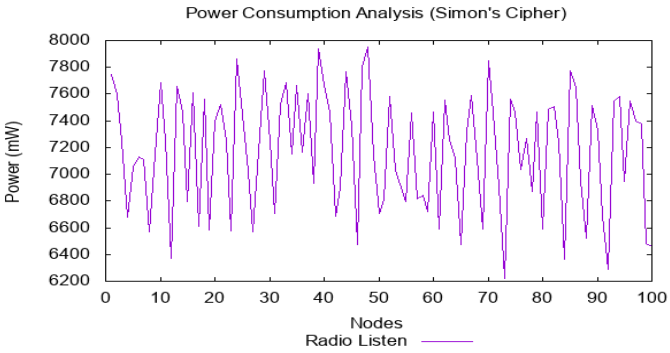


Fig 7: Simon: Radio Listen vs Power Consumption

Consumption of the nodes in the network. Figure 4 shows the average power consumption analysis of Simon's cipher with respect to radio transmit. compare to other parameters radio transmission consumes more power. Figure 5 shows the power consumption analysis of Simon's cipher CPU utilization. Similarly, Figure 6 and Figure 7 shows the average power consumption of Simon's algorithm vs Longest prefix match (LPM) and radio listen.

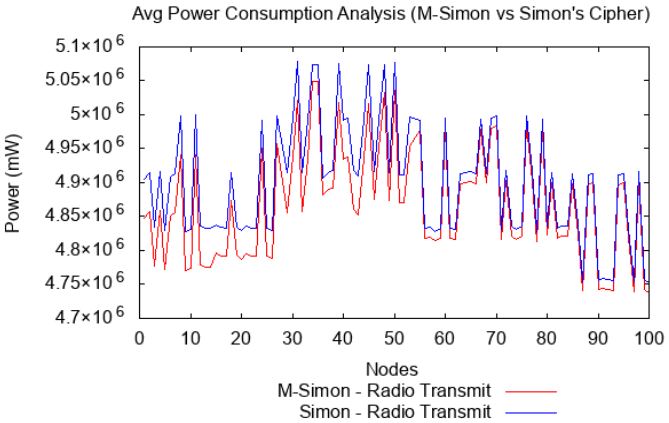


Fig 8: Radio Transmit: M-Simon vs Simon

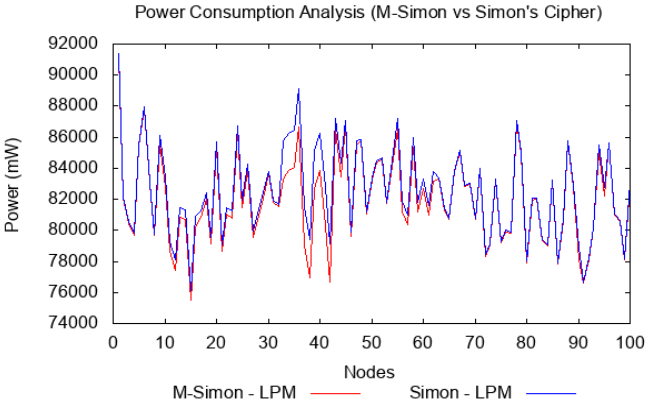


Fig 9: LPM: M-Simon vs Simon

Algorithm	LPM	RL	RT	CPU
Simon	82491.12 mW	7176.664 mW	4899218 mW	29.4186 mW
M-Simon	81994.02 mW	6747.674 mW	4867117 mW	28.99149 mW

Table III Power consumption analysis of Simon vs M-Simon

IV. RESULTS

The power consumption of M-Simon and Simon's algorithm is proposed. The encryption process is carried out for a different range of nodes from 0 to 100 with radio listen and radio transmission parameters by different cycles of CPU. The power consumption of Simon and M-Simon are analyzed. When nodes are idle, there is no difference in power consumption but when encrypted data transmitted from one device to another device, energy consumption of the M-Simons is comparatively better than the Simon's algorithm. Figure 8, 9, and 10 show the power consumption comparison of M-Simon vs Simon's algorithm with respect to radio transmission, CPU, LPM and radio listen. In all the three-energy consumption analysis shows that the power consumption of M-Simon's algorithm is very effective compare to Simon's algorithm. In Table 3 experimental results are concludes that the average of CPU cycle of Simon's algorithm 29.40986 mW and Radio listen is 7176.664 mW with one hour of heart beat data transmission among 100 T-Mote sky motes, whereas comparing with M-Simon the CPU cycle is 28.99149 mW and radio listen is 47225.85 mW which is 6747.674 mW less CPU utilization and 482.99 mW reduced radio listen.

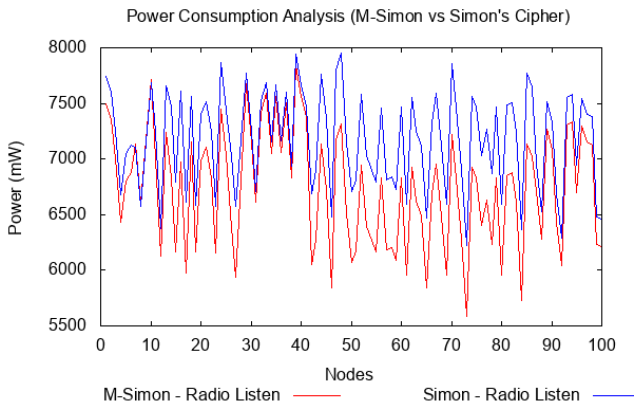


Fig 10: Radio Listen: M-Simon vs Simon

Implementation of M-Simon (24/48) concludes that the implementation of a reduced number of cycles reduces the power consumption of CPU processing, as well as radio listen. Also, the proposed model satisfies all NIST recommendations K. A. McKay. (2017) [16] for a lightweight device security approach which shown in Table 2 which shows the M-Simon's CPU power utilization is more significant which can also defend

against the side channel attack.

V. CONCLUSION

Modified Simon's mix column lightweight algorithm is implemented. M-Simon (24/48) is implemented to reduce the power consumption of the IoMTs by implementing a reduced number of secure round function from 36 to 24 and the sensor data is encrypted with smaller 48-bit key mix column algorithm. The results help the IoMTs to carry out the power efficient secure data communication among Internet of medical things. The less power consuming device will work a longer period of time with improved security.

REFERENCES

1. M. A. Iqbal. (2016, August). A novel authentication and key agreement protocol for internet of things-based resource-constrained body area sensors. In 2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW). [online]. pp. 315-320. Available: https://ieeexplore.ieee.org/document/7592744
2. M. Haghi. (2017). Wearable devices in medical internet of things: scientific research and commercially available devices. Healthcare informatics research. [online]. 23(1), pp. 4-15. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5334130/
3. M. A. Bahnasawi. (2016, December). ASIC-oriented comparative review of hardware security algorithms for internet of things applications. In 2016 28th International Conference on Microelectronics (ICM). [online]. pp. 285-288. Available: https://ieeexplore.ieee.org/document/7847871
4. S. Feizi. (2014, October). A hardware implementation of Simon cryptography algorithm. In 2014 4th International Conference on Computer and Knowledge Engineering (ICCKE). [online]. pp. 245-250. Available: https://ieeexplore.ieee.org/document/6993386
5. P. A. Wortman. (2017, February). Proposing a modeling framework for minimizing security vulnerabilities in IoT systems in the healthcare domain. In 2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI). [online]. pp. 185-188. Available: https://ieeexplore.ieee.org/document/7897236
6. S. S. S. Priya. (2015, March). FPGA implementation of efficient AES encryption. In 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). [online]. pp. 1-4. Available: https://ieeexplore.ieee.org/document/7193081
7. F. M. Nascimento. (2010). A VHDL Implementation of the Lightweight Cryptographic Algorithm HIGHT. Algorithms. [online]. 2(4), 5. Available: http://sbmicro.org.br/sforum-eventos/sforum2015/09.pdf
8. Y. A. Abbas. (2014, November). Implementation of PRINCE algorithm in FPGA. In Proceedings of the 6th International Conference on Information Technology and Multimedia. [online]. pp. 1-4. Available: https://ieeexplore.ieee.org/document/7066593
9. M. R. Z'aba. (2014). I-PRESENTTM: An involutive lightweight block cipher. Journal of Information Security. [online]. 5(03), 114. Available: https://www.scirp.org/journal/PaperInformation.aspx?PaperID=48057
10. M. Usman. (2017). SIT: a lightweight encryption algorithm for secure internet of things. arXiv preprint arXiv:1704.08688. Available: https://arxiv.org/abs/1704.08688
11. D. Aakash. (2016). Lightweight security algorithm for wireless node connected with IoT. Indian J. Sci. [online]. Technol, 9, 1-8. Available: http://www.indjst.org/index.php/indjst/article/view/99035
12. S. R. Chatterjee. (2014, December). FPGA implementation of pipelined blowfish algorithm. In 2014 Fifth International Symposium on Electronic System Design. [online]. pp. 208-209. Available: https://ieeexplore.ieee.org/document/7172778
13. R. Beaulieu. (2015). SIMON and SPECK: Block Ciphers for the Internet of Things. IACR Cryptology ePrint Archive. [online]. Available:https://csrc.nist.gov/csrc/media/events/lightweight-cryptography-workshop-2015/documents/papers/session1-shors-paper.pdf
14. D. Halperin. (2008). Security and privacy for implantable medical devices. IEEE pervasive computing. [online]. 7(1), 30-39. Available: https://ieeexplore.ieee.org/document/4431854
15. R Somasundaram. (2018). Preventing Unauthorized Access to Internet-of-Things Medical Devices Using Packet Filtering Device Level Embedded Firewall. Journal of Computational and Theoretical Nanoscience. [online]. 15(6-7), 2174-2178. Available: https://www.ingentaconnect.com/content/asp/jctn/2018/00000015/00000006/art00065
16. K. A. McKay. (2017). NISTIR 8114 report on lightweight cryptography. National Institute of Standards and Technology (NIST), Gaithersburg. [online]. Available: https://csrc.nist.gov/publications/detail/nistir/8114/final

AUTHORS PROFILE

Somasundaram Ragupathy is a research scholar in the School of Computer Science and Engineering at Vellore Institute of Technology, Vellore, India. He received his Master's in Computer Science and Engineering from Arulmigu Meenakshi Amman College of Engineering, Anna University Chennai. Currently working as an Assistant Professor in the Department of Computer Science and Engineering at Arulmigu Meenakshi Amman College of Engineering, Anna University. He has teaching experience of around Four years. His area of specialization is Network Security. He had presented six papers in national and international conferences and published two book chapters.

Mythili Thirugnanam is an Associate Professor in the School of Computer Science and Engineering at VIT University, Vellore, India. She received a Master's in Software Engineering from VIT University. She has been awarded doctorate in Computer Science and Engineering at VIT University in 2014. She has teaching experience of around 12 years. She has a research experience of 3 years in handling sponsored projects funded by Govt. of India. Her area of specialization includes Image Processing, Software Engineering and Knowledge Engineering. She has published more than 25 papers in international journals and presented around seven papers in various national and international conferences.