SAL – A Lightweight Symmetric Cipher for Internet-of-Things

Hemraj Shobharam Lamkuche, Dhanya Pramod, Vandana Onker, Shobharam Katiya, Geeta Lamkuche, Gurudevi Hiremath

Abstract— The modern era of computing demands high-speed performance and maximizing efficiency for commercial applications and business modules. Embedded systems and devices are unlike traditional computer systems. The computing power of the embedded device is very limited to perform a specific task, it consumes little hardware footprint and operates at electronic high speed by minimizing clock cycles and memory management. Embedded security issues and security challenges are a major concern resisting against advanced cryptanalytic attacks. In this research, we propose a new lightweight cryptographic algorithm SAL (Secure Advanced Lightweight). The algorithm is based on NIST (National Institute of Standards and Technology) standards and guidelines in design implementation of SAL to secure lower end of the device spectrum are internet of things (IoT), wireless sensors network (WSNs), aggregation network, pervasive devices, Radio frequency identification (RFID) tags, and embedded devices. SAL operates on the 64-bit block with a key size of 64-bit to 128-bit for 14 rounds of Feistel function. The internal structure of SAL powered with tiny 16-bit S-box using composite field arithmetic (CFA) technology. Our experimental results prove SAL has high efficiency and high throughput to achieve optimum performance when compared with standard cryptographic algorithms.

Keywords: Lightweight Cryptography, Block Cipher, Feistel Network, IoT, Energy Efficient, Embedded Device.

I. INTRODUCTION

Over the last decade, the exploration of innovative lightweight block ciphers has been proposed for the Internet of Things (IoT) devices, embedded devices, and power-constrained devices. These computational devices were exclusively penetrated in the everyday life of human beings. With reference to a research investigation conducted by Internet System Consortium, over 1012 million active hosts are connected to the internet until January 2019 (ISC, 2019). With a huge flow of information from one device to another device for communication, it leads to major challenges in preserving data security and privacy. The data security is an integrated part of information security relies on encryption and decryption process. These processes involved the mathematical and engineering process of encoding source information as plaintext into ciphertext by using a secret key of standard size. The scrambling of information can be decrypted into original plaintext information by decrypting it using the same secret key used for the encryption process (Stallings, 2010). The National Institute of Standards and Technology (NIST) has approved lightweight cryptographic standards for encryption algorithms specifically for low power-constrained devices (Kerry A. McKay, Larry Bassham, Meltem Sönmez Turan, 2017). The physical characteristics of lightweight block ciphers restricted up to 1900 gate equivalents (GEs) and the performance characteristics of hardware implementation and software implementation should be less than 20 ns latency.

According to the IHS Markit report estimate, the sum of connected IoT devices will increase to 125 Billion by 2030. The IoT is impacting nearly all market areas from raw material to end products (Graham, 2017). The number of handheld devices which includes: mobile devices, smart band, smart shoes, and tablets will reach up 23.14 billion by the end of 2018. With an increase in the number of IoT devices, it will also arise security risks and challenges to handle big data, processing a massive amount of data, and transmit big data over a secure communication channel. IoT allows us to access the object across an existing interconnection of the network remotely. Embedded systems main characteristics are Connectivity, Things (which includes sensors), Data, Communication, Intelligence, Action, and Ecosystem. Each node in IoT involves transmitting information over the communication channel, and it also processes sensitive data that needs to protect against intruders and malicious adversaries. Since the IoT devices are low-resource, low power-constrained devices, limited battery to operate, and limited power supply. As of late, the advancement of power-constrained devices in embedded systems, block ciphers are important for cryptographic applications, such as embedded systems, sensor networks, RFID labels, and pervasive hubs. Embedded devices manufactured with a lower maintenance cost, higher network toughness, stronger self-organization and broader applicability features, which has become a crucial part of the networking industry.

Designing a unique block cipher is a tedious task, A prominent lightweight block cipher must be fast and

Revised Manuscript Received on September 10, 2019.

Hemraj Shobharam Lamkuche, Research Scholar, Symbiosis International (Deemed University) Pune, Maharashtra, India.
(Email: Hemraj.lamkuche@gmail.com)
Dhanya Pramod, Director - SCIT, Symbiosis International (Deemed University) Pune, Maharashtra, India.
(Email: dhanya@scit.edu)
Vandana Onker, Research Scholar, Indira Gandhi National Open University, Bhopal, Madhya Pradesh, India.
Shobharam Katiya, Central Railway, Ministry of Railway, India.
Geeta Lamkuche, Station Master, Central Railway, Ministry of Railway, India.
Gurudevi Hiremath, Assistant Professor, Solapur, Maharashtra, India.

secure. It must resist cryptanalytic attacks to retrieve the secret key. Over past decades, many lightweight block ciphers are introduced like AES (Daemen, Rijmen, & Leuven, 1999), CHASKEY (Mouha et al., 2014), CHAM (Koo et al., 2017), CLEFIA (Shirai, Shibutani, Akishita, Moriai, & Iwata, 2007), FeW (Kumar, Pal, & Panigrahi, 2014), HIGHT (Hong et al., 2006), KLEIN (Gong, Nikova, & Law, 2012), LBLOCK (W. Wu & Zhang, 2011), LED (J. Guo, Peyrin, Poschmann, & Robshaw, 2011), MCRYPTON (Lim & Korkishko, 2006), PICCOLO (Shibutani et al., 2011), PRESENT (Leander et al., 2007), RECTANGLE (Zhang et al., 2015), ROADRUNNER (Baysal & Şahin, 2016), SKINNY (Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P. and Sim, 2016), and TWINE (Suzaki, Minematsu, Morioka, & Kobayashi, 2012).

In this paper, we have proposed a new unique lightweight block cipher SAL (*Secure Advanced Lightweight*) based on tiny compact s-box embedded with composite field arithmetic (CFA) technology to minimize hardware utilization on field-programmable gate array (FPGA) board. It takes input of 64-bit plaintext along with a key size of 64-bit extended to 128-bit using 14 and 16 rounds respectively. Moreover, the lightweight implementation of SAL shows enormous strength against various cryptanalytic attacks. The SAL encryption and decryption algorithm are useful for smart devices and power-constrained devices like the Internet of Things, Embedded devices and Radio frequency identification (RFID) tags.

II. IOTS: SECURITY CHALLENGES AND REQUIREMENTS

The IoT devices are specifically developed to perform a specific task for a long time. These devices are different from traditional computer system devices, it consumes less area for hardware implementation, it provides high hardware utilization to gain performance and it can provide resistance against known cryptographic attacks. The IoT devices operate on high electronic speed by minimizing the clock cycles and memory management. We have listed out security challenges, security requirements for IoT devices and privacy threats which were more criticized for the IoT devices

A. Security challenges in IoTs

- Critical functionality in IoTs
- Replication
- Security assumptions
- Not easily patched
- Long life cycle
- Industry specific protocols
- Deployment
- Updates

B. Security requirements for IoTs devices

- Anonymity
- Authentication
- Confidentiality
- Data Freshness
- Data Integrity
- Data Security
- Device tampering detection
- Security management in IoTs
- Fault Tolerance
- Firmware updates
- Liability
- Monitoring Intrusion detection
- Non-Repudiation
- Protection against attacks
- Physical Protection
- Privacy
- Resilience
- Secure Boot
- Secure Communication
- Secure source code updates
- Security
- Self-Healing
- System Maintenance
- Trust

C. Privacy threats in IoTs devices

- Threat Identification
- Localization and Tracking
- Profiling information
- Privacy violating interaction
- Privacy violating presentation
- Lifecycle transitions
- Inventory attacks
- Linkage

There is no single step security solution for IoT devices. Security prerequisites must consider over the expenditure of a security catastrophe, the risk of cyber-attacks, accessible attack vectors, and the expense of actualizing a security arrangement. Security is the essential enabling factor of most pervasive system applications.

III. SAL: A LIGHTWEIGHT BLOCK CIPHER

SAL is a 64-bit lightweight block cipher operates on Feistel structure which takes input as the plaintext of 64-bit and key size of 64-bit to produce ciphertext of 64-bit. The key size can be extended from 64-bit to 128-bit by increasing the number of rounds in the Feistel network of SAL. The Feistel structure of SAL operates for 14 rounds simultaneously using composite field arithmetic S-box function which is superimposed with a new unique inverter *H* function. In each round of Feistel function, there is various mathematical operation like permutation and substitution, XOR operation, key manipulation, and inverter function. At the end of 14 round of Feistel network, the extra two 32-bit keys are used as key whitening to generate the more complex ciphertext. SAL lightweight encryption algorithm has four layers of operation which is mentioned below.

- *Initialization Layer:*

In this layer of the SAL algorithm, the 64-bit input plaintext is divided into two 32-bit halves. The left half is the first XOR with the 32-bit key from the p-array. The output of the left half act as input to the next layer of the SAL algorithm.

- *P-Layer:*

This layer consists of sixteen 32-bit subkeys, which are used in each round of the Feistel function. These keys provide more complex substitution in generating the ciphertext. Out of 16 p-array subkeys, 14 subkeys are used in each round of Feistel function. The remaining 2 subkeys are used for extra swapping of information at the later stage of Feistel function in generating complex cipher.

- *S-box Layer:*

This is the most important layer in the SAL algorithm. We have implemented a compact S-box based on composite filed arithmetic technology. The 4-bit s-box consume very little hardware implementation, eight 4-bit S-box used in parallel to produce confusion property which gives non-linearity to SAL lightweight algorithm. The output of combined S-box produces 32-bit output which acts as input to the next layer of the Feistel function

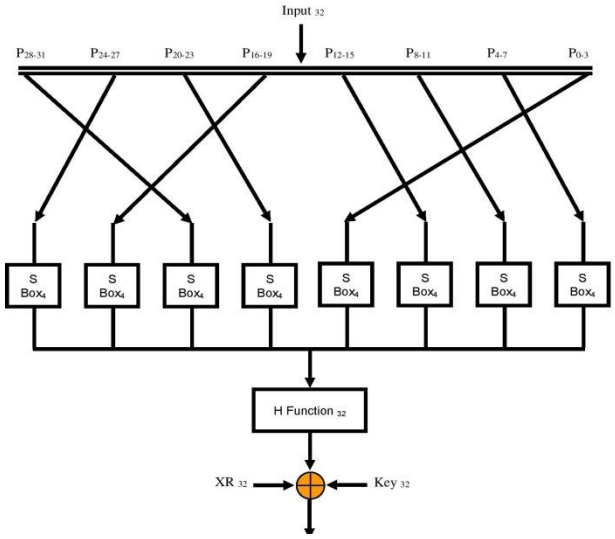

Figure 1: Feistel Function of SAL

shown in figure 1.

The implementation of compact 4-bit S-box to accomplish the highest clock frequency, we have introduced a three-layer pipelining approach which is used to divide the logic delay and effectively reduced it using pipelined registers. We have defined a static function in one 4-bit S-box which is used by other seven 4-bit S-boxes. The value of S-box is shown in Table I.

Table I: Values of compact S-box

X	0	1	2	3	4	5	6	7
S-box (X)	E	7	8	4	1	9	2	F
X	8	9	A	B	C	D	E	F
S-box (X)	5	A	B	0	6	C	D	3

- *H Function Layer:*

In addition to feistel F function, we have also introduced a new H function which is an invertible function operates at electronic speed due to usage of inverter logic gate (Y. Huang et al., 2001). The implementation of H function is given in equation 1.

$$(X): NOT\ ((X) \oplus (L/R)) \quad (1)$$

- *Key Schedule Layer:*

This layer has two important components: at first it initializes the S-boxes and then initialize p-array function. This allow all the entries to be filled in p-array and S-boxes using hexadecimal digits. All the entries from p-array block is XORed with 64-bit key. This progression will continue until all s-box and p-array entries are substituted.

- *Key Whitening Layer:*

At the end of 14 round of feistel function for SAL encryption process, the extra two 32-bit keys are used to generate more complex cipher.

The figure 2 shows feistel structure of SAL lightweight block cipher in accordance with all above layers.

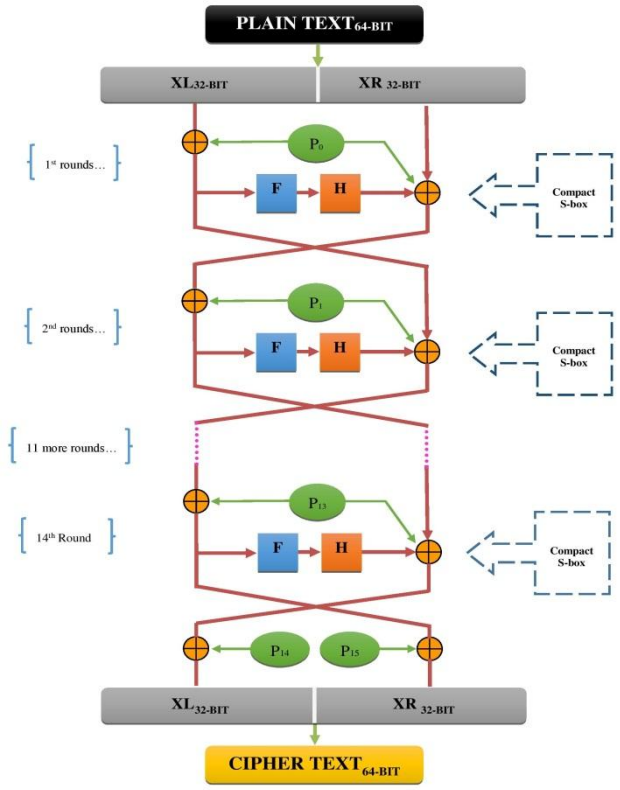

Figure 2: SAL Feistel Network

IV. IMPLEMENTATION AND RESULTS

We have proposed hardware implementation of the SAL encryption algorithm using Digilent Nexys4 DDR Artix-7 FPGA board powered with the Xilinx XC7A100T module. Our algorithm is divided into several modules that are designed, implemented, tested and authenticated

using synthesis and implementation on VHDL (VHSIC Hardware Description Language) and RTL (Register-transfer level). VHDL allows us to implement the detailed design of a synchronous digital circuit in which information flows between hardware registers and logical operations on data signals. We preferred to make use of FPGA for hardware implementation of our algorithm because it is reprogrammable capability and low cost. FPGA are nearly ideal runners for the rapid deployment of lightweight cryptography for wireless sensors networks, RFID tags, aggregation network, IoT, and pervasive computing. We proposed an optimized implementation of compact 4-bit × 4-bit S-box benchmarked using composite filed arithmetic technology which minimizes space complexity. We have also used pipelined registers that exist in the Artix-7 FPGA device to map compact S-box operations; this allows S-box to operate at extremely high speed to gain performance and achieve high throughput, it also enables in minimizing implementation cost of compact S-box over FPGA. Table II shows the initial parameters of the Digilent Nexys4 DDR Artix-7 FPGA board. Table III comprises the performance evaluation of the SAL encryption algorithm over the FPGA board. The performance metrics were evaluated using Xilinx Vivado 2019 edition in accordance with metrics like design, synthesis, implementation, post map, post route mapping, and bitstream generation.

Table II: Digilent Nexys4 DDR device features

Digilent Nexys4 DDR Artix-7 FPGA Board	
Project Family	Artix-7
Part No.	XC7A100TCSG324-1
Process Nodes	28 *nm*
Target Language	HDL / Verilog
Device Name:	Digilent Nexys4 DDR
#Slices	15,850
RAM	4,860 Kbits
Clock	6 Clock
DSP slices	240
Slice LUTs	63,400
Slice Registers	126800
F7 Muxes	31700
F8 Muxes	15850
Block RAM tile	135
Bonded IOB	210
Distributed RAM	1188
Temperature	125°C
Delay	300
I/O_LOGIC	210
Internal Clock Speed	450 MHz
On-Chip Analog to Digital Converter	XADC

TABLE III: Performance evaluation of SAL cryptographic algorithm

Profile	Values
Algorithm Name	*SAL* (Secure Advanced Lightweight)
Family	Lightweight Block Cipher
Block Size	64-bit
Key Size	64-bit / 128-bit
Number of Rounds	14
Structure	Feistel Network
CPU Name	Intel Core i5 2430M CPU @ 2.40 GHz
CPU Speed	2394 MHz
OS Name	Windows 7, 64-bit
System RAM	6.00 GB
FPGA Device	Digilent Nexys4 DDR Artix™-7 Board
Device Family	Xilinx XC7A100T
Target Package	CGS324
Target Speed	-1
Development Tool	Xilinx® Vivado 2019
Design Mode	RTL
Optimization Goal	Speed
Maximum Operating Frequency	293.8 MHz
Number of Slices Used	114
Number of Slice FFs	189
Number of LUTs	70
Number of bonded IOBs	32
Number of GCLKs	2
Latency	148
Throughput	264.48 Mbps
Critical Path Delay	4.5 ns
Power Utilization	3.57 mW
Efficiency (Mbps/Slices)	2.32

The performance comparison of a various lightweight cryptographic algorithm is explained in Table IV below. The table concludes that the SAL algorithm consumes less cost due to small compact S-box powered with composite field arithmetic technology. We have compared our SAL encryption algorithm with a standard NIST approved AES algorithm implemented on the FPGA platform. Several

other lightweight cryptographic algorithms like PRESENT, ICEBERG, HIGHT, LED, SIMON, XTEA, and PRINT cipher were implemented and compared with SAL encryption algorithms. The efficiency of various algorithms neatly explains the security tradeoff in implementing cryptographic algorithms.

Table IV: Performance comparison of lightweight cryptographic algorithms

Cipher	FPGA	freq. (MHz)	T'puts (Mbps)	Slices	Efficiency	Ref
SAL	Artix-7 XC7A100T	293.8	264.48	114	2.32	This Paper
PRESENT	Spartan-III XCS400-5	254	508	271	1.87	(X. Guo, Chen, & Schaumont, 2008)
ICEBERG	Virtex-II	-	1016	631	1.61	(Quisquater, Legat, Standaert, Rouvroy, & Piret, 2004)
SEA	Virtex-II XC2V4000	145	156	424	0.368	(Standaert, Piret, Gershenfeld, & Quisquater, 2006)
AES	Spartan-II XC2S30-6	60	166	522	0.32	(Chodowiec & Gaj, 2003)
AES	Spartan-III XC3S2000-5	196.1	25,107	17,425	1.44	(Good & Benaissa, 2005)
HIGHT	Spartan-3 XC3S50-5	163.7	65.48	91	0.72	(Yalla & Kaps, 2009)
LED	Artix-7 XC7A100T-1	378	21.6	37	0.58	(Nalla Anandakumar, Peyrin, & Poschmann, 2014)
XTEA	Spartan-3 XC3S50-5	62.6	35.77	254	0.14	(Kaps, 2008)
Simon	Spartan-3 XC3S500	136	3.60	36	0.10	(Aysu, Gulcan, & Schaumont, 2014)
PRINT	Spartan-3 XC3S700A	147.73	147.7	210	0.703	(Okabe, 2016)
PRINT	Artix-7 XC7A100T	293.51	293.5	139	2.11	(Okabe, 2016)

V. SECURITY ANALYSIS

We perform cryptanalysis on the SAL encryption algorithm to test the security of an encryption algorithm. This paper applied several attacks which include Avalanche criteria, differential cryptanalysis using NPCR (Number of changing pixel rate) and UACI (Unified average changed intensity) analysis, and Brute force attack.

A. Avalanche criteria

The avalanche criteria are used to test the muddle property of our compact S-box. If a traditional block cipher or a lightweight block cipher does not satisfy the avalanche effect to a substantial degree, then it has deprived randomization. whenever an attacker changes the input by flipping some or a single bit, then the output of ciphertext changes expressively by more than 50% output bits flip. Avalanche effect follows strict avalanche criterion (SAC) properties in encryption and decryption process shown in Table V.

Table V: Avalanche Criteria on SAL encryption algorithm

Input	Hex	64-bit Binary	#Bit Changes
Plaintext	123456789 ABCDF0	00000001 00100011 01000101 01100111 10001001 10101011 11001101 11110000	36
Ciphertext	17F6A9A89 FFFC25	00000001 01111111 01101010 10011010 10001001 11111111 11111100 00100101	
Plaintext	223456789 ABCE00	00000010 00100011 01000101 01100111 10001001 10101011 11001110 00000000	38
Ciphertext	96DAADE7B 5AAF5B	00001001 01101101 10101010 11011110 01111011 01011010 10101111 01011011	

B. Differential cryptanalysis

The differential cryptanalysis is performed on chosen-plaintext, then we analyze the various results achieve the known ciphertext. In this category of attacks, we have performed NPCR (Number of changing pixel rate) and UACI (Unified average changed intensity) on various input images used during the encryption process of the SAL algorithm. The upper-bound percentage of the NPCR score is 100%. The NPCR score of any cipher must be close enough to 99% (Y. Wu, Member, Noonan, & Member, 2011). A small alteration in the input image will reflect a vast difference in resultant cipher image then differential cryptanalysis may turn out to be unusable. UACI illustrates the average variance between two harmonizing cipher images. Table VI shows the comparison of NPCR score and UACI score based proposed encryption algorithm and related lightweight ciphers.

Table VI: Comparative analysis of NPCR and UACI Score

Algorithms	NPCR score	UACI score
SAL (This paper)	99.80	29.55
(Loukhaoukha, Chouinard, & Berdai, 2012)	99.58	28.62
(Sathishkumar, Bhoopathy bagan, & Sriraam, 2011)	98.47	32.21
(C. K. Huang & Nien, 2009)	99.42	24.94
(C. K. Huang, Liao, Hsu, & Jeng, 2013)	99.54	28.27
(Armand Eyebe Fouda, Yves Effa, Sabat, & Ali, 2014)	99.60	33.42
(Hussain, Anees, AlKhaldi, Algarni, & Aslam, 2018)	99.30	33.40

C. Brute force attack

Brute force is an exhaustive search technique, rather than rational strategies. It takes all feasible combinations to recuperate the undisclosed key used for encryption and decryption. This type of attack used when it is not possible for an attacker to yield the benefit of weakness in the encryption arrangement. We have used CrypTool 2.1 (8145.1) tool to perform a brute force attack on the proposed SAL encryption algorithm to recover the 64-bit key used during encryption and decryption respectively (Kopal, Kieselmann, Wacker, & Esslinger, 2014). Brute force attack was executed using CryptTool 2.1 (8145.1) uninterruptedly on ASUS X53S laptop powered with Intel® Inside™ Core i5 2430M CPU 2.40 GHz, 6GB RAM for several days long but it was not able to recover the full 64-bit key. Hence, our SAL algorithm resists against a brute force attack for a reasonable amount of time.

VI. CONCLUSION

In this research paper, we have proposed a new lightweight cryptographic algorithm SAL based on NIST standards and guidelines in design implementation of SAL to secure lower end of the device spectrum is internet of things (IoT), wireless sensors network (WSNs), aggregation network, pervasive devices, Radio frequency identification (RFID) tags, and embedded devices. The hardware implementation cost of the SAL algorithm in lookup tables (LUT) is extremely less when compared with NIST approved lightweight block ciphers. Our security analysis shows SAL is the most secure lightweight block cipher in terms of its efficiency. The compact S-box implementation consumes very little area due to composite field arithmetic technology. Our SAL lightweight algorithm suits well for lightweight applications which include domain-like: internet of things, wireless sensors networks and RFID tags. It overcomes the security challenges, security requirements and privacy threats in IoTs that we have listed above. In the concept of smart city and sustainable energy environment, the SAL algorithm can protect information with bare minimum energy to operate to achieve the highest security.

VII. ACKNOWLEDGMENT

I am highly obliged and hugely indebted to my research supervisor Dr. Dhanya Pramod for her constant guidance and supervision at every level of my research work. I respectfully honor her valuable suggestions. She has taken a lot of care and strain for the successful completion of this work. All the same, I would like to thank my family for motivating and supporting me in this endeavour and letting me spread my wings all over. My father who have always inspired and encouraged me to complete this work successfully.

VIII. REFERENCES

1 Armand Eyebe Fouda, J. S., Yves Effa, J., Sabat, S. L., & Ali, M. (2014). A fast chaotic block cipher for image encryption. *Communications in Nonlinear Science and Numerical Simulation*. https://doi.org/10.1016/j.cnsns.2013.07.016

2 Aysu, A., Gulcan, E., & Schaumont, P. (2014). SIMON says: Break area records of block ciphers on FPGAs. *IEEE Embedded Systems Letters*, *Vol 6*(2), 37–40.

3 Baysal, A., & Şahin, S. (2016). RoadRunneR: A small and fast bitslice block cipher for low cost 8-bit processors. *Lightweight Cryptography for Security and Privacy. LightSec 2015*, *Vol 9542*, 58–76.

4 Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P. and Sim, S. M. (2016). The SKINNY Family of Block Ciphers and its Low-Latency Variant MANTIS. In *Annual International Cryptology Conference, Advances in Cryptology – CRYPTO 2016* (Vol. 9814, pp. 123–153).

5 Chodowiec, P., & Gaj, K. (2003). Very Compact FPGA Implementation of the AES Algorithm. *Cryptographic Hardware and Embedded Systems - CHES 2003*, *Vol 2779*, 319–333.

6 Daemen, J., Rijmen, V., & Leuven, K. U. (1999). AES Proposal : Rijndael. *The Rijndael Block Cipher*, 1–45.

7 Gong, Z., Nikova, S., & Law, Y. W. (2012). KLEIN: A new family of lightweight block ciphers. In *RFID. Security and Privacy. RFIDSec 2011* (Vol. 7055, pp. 1–18).

8 Good, T., & Benaissa, M. (2005). AES on FPGA from the Fastest to the Smallest. In *Cryptographic Hardware and Embedded Systems – CHES 2005* (Vol. 3659, pp. 427–440).

9 Graham, L. (2017). Number of Connected IoT Devices Will Surge to 125 Billion by 2030, IHS Markit Says. Retrieved October 24, 2017, from http://news.ihsmarkit.com/press-release/number-connected-iot-devices-will-surge-125-billion-2030-ihs-markit-says

10 Guo, J., Peyrin, T., Poschmann, A., & Robshaw, M. (2011). The LED block cipher. *Cryptographic Hardware and Embedded Systems – CHES 2011*, *Vol 6917*, 326–341.

11 Guo, X., Chen, Z., & Schaumont, P. (2008). Energy and performance evaluation of an FPGA-based SoC platform with AES and PRESENT coprocessors. In *Embedded*

Computer Systems: Architectures, Modeling, and Simulation. SAMOS 2008. (Vol. 5114, pp. 106–115).
12 Hong, D., Sung, J., Hong, S., Lim, J., Lee, S., Koo, B., … Chee, S. (2006). HIGHT: A New Block Cipher Suitable for Low-Resource Device. *Cryptographic Hardware and Embedded Systems - CHES 2006*, *Vol 4249*, 46–59.
13 Huang, C. K., Liao, C. W., Hsu, S. L., & Jeng, Y. C. (2013). Implementation of gray image encryption with pixel shuffling and gray-level encryption by single chaotic system. *Telecommunication Systems*. https://doi.org/10.1007/s11235-011-9461-0
14 Huang, C. K., & Nien, H. H. (2009). Multi chaotic systems based pixel shuffle for image encryption. *Optics Communications*. https://doi.org/10.1016/j.optcom.2009.02.044
15 Huang, Y., Duan, X., Cui, Y., Lauhon, L. J., Kim, K.-H., & Lieber, C. M. (2001). Logic Gates and Computation from Assembled nanowires building blocks. *Science*, *Vol 294*(5545), 1313–1317.
16 Hussain, I., Anees, A., AlKhaldi, A. H., Algarni, A., & Aslam, M. (2018). Construction of chaotic quantum magnets and matrix Lorenz systems S-boxes and their applications. *Chinese Journal of Physics*. https://doi.org/10.1016/j.cjph.2018.04.013
17 ISC. (2019). Internet Systems Consortium, Internet Domain Survey. Retrieved January 13, 2019, from http://ftp.isc.org/www/survey/reports/current/
18 Kaps, J. P. (2008). Chai-tea, cryptographic hardware implementations of xTEA. *Progress in Cryptology - INDOCRYPT 2008*, *Vol 5365*, 363–375.
19 Kerry A. McKay, Larry Bassham, Meltem Sönmez Turan, N. M. (2017). Report on lightweight cryptography. https://doi.org/10.6028/NIST.IR.8114
20 Koo, B., Roh, D., Kim, H., Jung, Y., Lee, D., & Kwon, D. (2017). CHAM : A Family of Lightweight Block Ciphers for Resource-Constrained Devices. *International Conference on Information Security and Cryptology (ICISC'17)*, *Vol 10779*, 3–25.
21 Kopal, N., Kieselmann, O., Wacker, A., & Esslinger, B. (2014). CrypTool 2.0. *Datenschutz Und Datensicherheit - DuD*, *Vol 38*(10), 701–708.
22 Kumar, M., Pal, S., & Panigrahi, A. (2014). FeW: A Lightweight Block Cipher. *IACR Cryptology EPrint Archive*, *Vol 2014*(Report 326), 1–18.
23 Leander, G., Knudsen, L. R., Bogdanov, A., Seurin, Y., Poschmann, A., Paar, C., … Robshaw, M. J. B. (2007). PRESENT: An Ultra-Lightweight Block Cipher. *Cryptographic Hardware and Embedded Systems - CHES 2007*, *Vol 4727*, 450–466.
24 Lim, C. H., & Korkishko, T. (2006). mCrypton - A lightweight block cipher for security of low-cost RFID tags and Sensors. *Information Security Applications. WISA 2005*, *Vol 3786*, 243–258.
25 Loukhaoukha, K., Chouinard, J.-Y., & Berdai, A. (2012). A Secure Image Encryption Algorithm Based on Rubik's Cube Principle. *Journal of Electrical and Computer Engineering*. https://doi.org/10.1155/2012/173931
26 Mouha, N., Mennink, B., Van Herrewege, A., Watanabe, D., Preneel, B., & Verbauwhede, I. (2014). Chaskey: An efficient MAC algorithm for 32-bit microcontrollers. *Selected Areas in Cryptography - SAC 2014*, *Vol 8781*, 306–323.
27 Nalla Anandakumar, N., Peyrin, T., & Poschmann, A. (2014). A very compact FPGA implementation of LED and PHOTON. In *International Conference on Cryptology in India, Progress in Cryptology - INDOCRYPT 2014* (Vol. 8885, pp. 304–321).
28 Okabe, T. (2016). Efficient FPGA Implementations of PRINT CIPHER. *International Journal of Emerging Technologies and Innovative Research*, *Vol 3*(4), 76–85.
29 Quisquater, J.-J., Legat, J.-D., Standaert, F.-X., Rouvroy, G., & Piret, G. (2004). ICEBERG : An Involutional Cipher Efficient for Block Encryption in Reconfigurable Hardware. In *In International Workshop on Fast Software Encryption - FSE 2004* (Vol. 3017, pp. 279–298).
30 Sathishkumar, G. ., Bhoopathy bagan, K., & Sriraam, N. (2011). Image Encryption Based On Diffusion And Multiple Chaotic Maps. *International Journal of Network Security & Its Applications*. https://doi.org/10.5121/ijnsa.2011.3214
31 Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., & Shirai, T. (2011). Piccolo: An ultra-lightweight blockcipher. *Cryptographic Hardware and Embedded Systems – CHES 2011*, *Vol 6917*, 342–357.
32 Shirai, T., Shibutani, K., Akishita, T., Moriai, S., & Iwata, T. (2007). The 128-bit blockcipher CLEFIA. In *Proceedings of the 14th international conference on Fast Software Encryption - FSE 2007* (pp. 181–195).
33 Stallings, W. (2010). NIST block cipher modes of operation for confidentiality. *Cryptologia*, *Vol 34*(2), 163–175.
34 Standaert, F. X., Piret, G., Gershenfeld, N., & Quisquater, J. J. (2006). SEA: A scalable encryption algorithm for small embedded applications. In *International Conference on Smart Card Research and Advanced Applications CARDIS 2006.* (Vol. 3928, pp. 222–236).
35 Suzaki, T., Minematsu, K., Morioka, S., & Kobayashi, E. (2012). Twine: A lightweight, versatile block cipher. *Selected Areas in Cryptography*, *Vol 7707*, 146–169.
36 Wu, W., & Zhang, L. (2011). LBlock: A lightweight block cipher. *International Conference on Applied Cryptography and Network Security ACNS 2011*, *Vol 6715*, 327–344.
37 Wu, Y., Member, S., Noonan, J. P., & Member, L. (2011). NPCR and UACI Randomness Tests for Image Encryption. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, *Vol 01*(02), 31–38.
38 Yalla, P., & Kaps, J. P. (2009). Lightweight cryptography for FPGAs. In *2009 International Conference on Reconfigurable Computing and FPGAs* (pp. 225–230).
39 Zhang, W., Bao, Z., Lin, D., Rijmen, V., Yang, B., & Verbauwhede, I. (2015). RECTANGLE: a bit-slice lightweight block cipher suitable for multiple platforms. *Science China Information Sciences*, *Vol 58*(12), 1–15.

IX. AUTHORS PROFILE

Hemraj Shobharam Lamkuche is a PhD scholar at Symbiosis International University Pune India. Currently pursuing full-time PhD degree under Symbiosis International University Pune India. His research experience is around 5 years. His research area is Information Security, Cryptography, Network Security, Network Analysis, Web Security, Embedded System Security, IoT Security, Cryptanalysis of conventional block ciphers, and Data Analytics. He has completed his M. Phil (Information Security) from Bharathiar University Coimbatore, Tamil Nadu, India. Also completed Bachelor and Master Degree in Computer Science from North Maharashtra University Jalgaon, Maharashtra, India.

Dhanya Pramod is a Professor and Director at the Symbiosis Centre for Information Technology (SCIT), a constituent of the Symbiosis International University (SIU), Pune. Pramod is a Post Graduate in Computer Science from Pondicherry Central University, India and Ph.D. in Computer Science from Symbiosis International University, India. She also holds a MBA from IGNOU, India. Her teaching and research interests are information security, networks and application security and aspect oriented programming. She has published papers in refereed journals and several conferences of international repute. She is a senior member of IACSIT, Singapore. She has a strong academic foundation and was the First Rank holder of university both at undergraduate and post graduate level.

Vandana Onker is a research scholar and perceiving MCA from Indira Gandhi National Open University Regional Centre Bhopal, M.P. India. She is currently working as fulltime MIS Coordinator under Government of Madhya Pradesh, India.

Shobharam Asaram Katiya is a retired Loco Pilot (Mail) Central Railway, Bhusawal under Ministry of Railway, India. Currently he is motivational speaker, yoga enthusiast and active researcher in multi-disciplinary domain.

Geeta Shobharam Lamkuche is a bright scholar holding MBA (Finance) from North Maharashtra University Jalgaon. She also completed M-com from North Maharashtra University Jalgaon. She is currently working as a fulltime Deputy Station Master designation at Ministry of Railway, India.

Gurudevi Rahul Hiremath is a assistant professor from Solapur, India.