International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-1S3, December 2019

Energy Efficient Light Weight Security Algorithm for Low Power IOT Devices

B. Nagajayanthi

Abstract: ***Internet of Things (IoT) is the state of art which connects, communicates, intelligently resolves and processes data between physical devices and smart phone or to a centralized server. Billions of users are centrally coordinated via the internet. The number of ubiquitous IoT devices will surpass the number of humans. For secured data transfer, IoT requires strenuous focus on security. Inspite of the secured IoT layered approach integrated in its architecture, yet they are susceptible to thwarting attacks. With proliferating applications and innovations, there is a stringent need to preserve user privacy and anonymize interactions using a lightweight cryptographic algorithm. Existing cryptographic algorithms have constraints on power, limited battery, real time execution, latency, code length and memory. In this research, initially comparison of the existing algorithms is made. Subsequently, Augmented Security and Optimized memory space is achieved for the data channelized via IoT by using the combination of the Light weight masked AES (Advanced Encryption Standard) and MD5 (Message Digest) hash algorithm. This chaining technique is implemented using VHDL Coding, Xilinx ISE and ModelSim 6.5 software tool. In the proposed algorithm, area, power and timing factors are reduced using optimization techniques, which drastically reduces the power consumed, and chip area. Chip area is calculated in terms of gate equivalents and power consumption is reduced through clock gating and operand isolation techniques.***

Index Terms—Chip Area, Gate Equivalents, Light Weight, S-Box.

I. INTRODUCTION

Internet of Things connects internet enabled heterogeneous devices in a wireless environment and strives to protect the devices and the network against cybercrime [1]. As per statistics from Statista, IoT, the network of internet enabled devices, will profoundly upsurge from 20.35 billion to 75.44 billion by 2025 [18]. Devices range from wearable entrenched devices in healthcare to industrial gadgets and military applications. Encryption has permeated in our day to day activities and involves sharing of personalized user details, location, and software etc., for ease of communication and data sharing [14]. IoT being open, is vulnerable to Denial-of-Service attacks (DoS), eavesdropping, Man-in-the Middle Attack (MITM), Masquerading and so on. Hackers can jeopardize public and private data. In Internet-of-Things, the devices are connected via Bluetooth, Wi-Fi, WLAN etc., to the internet. These serve as loopholes for hackers. Due to resource constraints in IoT, at times, data is transmitted even Without enciphering. Licensed bands are used by 2G/3G and unlicensed bands are used by Wi-Fi (2.4 GHz) for IoT. To deploy IoT, devices use sensors, RFID tags, etc., Radio-Frequency Identification tags encode digital data. RFID does not require line-of-sight. It uses electromagnetic waves to track the device using electronic barcode in real time. Sensors collect information such as pressure, temperature etc., Each layer of IoT, has an inbuilt security layer for securing the data.

In IoT architecture, the Perception Layer has RFID, GPRS, sensors, etc., connected to it. IEEE 802.15.4 provides security solutions at this layer. From this layer, data is sensed and transferred to the Network Layer. Data is divided into packets and is sent from the source to the destination using IPv6. It uses networks such as wireless network, satellite etc., AES is implemented in this layer using IPSec protocol. From this the data is collected by the Support Layer. It sets the support platform for the application layer [17]. IoT architecture has the unreliable User Datagram Protocol. As this is unreliable, Datagram Transport Layer Security (DTLS) is used for security in this layer. The Application layer involves smart IoT applications like healthcare. This layer has Constrained Application Protocol (COAP) for security.

Inspite of the secured IoT architecture, data channelized via internet is susceptible to data snooping attacks. So cryptographic algorithms are used to ensure data privacy. Message transmitted is referred to as the plain-text. Encrypted data is referred to as cipher-text. This cryptosystem uses algorithms for encryption and decryption. The secrecy of the message depends on the security level of the key. This depends on the length of the key. The larger the key size, the more time it takes for the hacker to undergo a laborious search of the key. Cryptographic algorithms include symmetric algorithm, asymmetric algorithm and hash function. Symmetric algorithm uses similar key for encryption and decryption .It ensures confidentiality, integrity, less key size but denies authentication of the sender. E.g., AES. Asymmetric algorithms involve different keys for encryption and decryption. It ensures confidentiality, authenticity and integrity. On the other hand, it consumes longest encryption time and occupies more space thereby making it less preferred for IoT. E.g., RSA. Its execution time, code length and memory space is more. Execution time depends on the key management and distribution. Cryptographic algorithms use Block Cipher or Stream Cipher. Block Cipher is a symmetric key cipher that operates on fixed length group of bits. On contrast, Stream cipher operates on one bit at a time.

Manuscript published on 30 December 2019.
***** Correspondence Author (s)
B. Nagajayanthi*, School of Electronics Engineering, VIT Chennai Campus, Chennai, India. Email: nagajayanthi .b@vit.ac.in

Hash function or digest, generates a reduced irreversible encrypted message digest from the large sized message input without using a key.

IoT data is preferably transferred over RFID rather than sensors or smart cards [6] [7]. RFID has RFID tag, reader and a server. RFID Tags are microchips mostly used in IoT transfer. Sensors do not provide authentication. Smart cards require human interaction. With the existing active and passive RFID, passive RFID is preferred as it does not require power and works on electromagnetic induction. Passive RFID operates on different radio frequency ranges. RFID reader reads information from the tag and sends it to the server. Standalone RFID is still not secure. RFID along with cryptographic algorithm is used for IoT.

A. Research Directions

IoT is different from internet enabled computers as it is resource constrained. It has components for sensing, heterogeneous access, information processing, applications and services [17]. IoT connects devices wirelessly, so they are vulnerable to attacks and has constraints on bandwidth and power. These devices are interdependent and follow,' if this then that', (IFTTT) strategy [12]. Devices are independently protected. But hackers use interdependence .For example, by controlling the temperature an air conditioner could be switched off remotely. IoT devices are managed by cloud platforms such as Alexa, which is common among smart home customers. Framing permission boundaries for a diversity of heterogeneous IoT devices is a challenge. Another major constraint is the use of implanted sensors in healthcare which poses a serious security threat in IoT. Traditional symmetric algorithms, asymmetric algorithms are not suitable for IoT due to its limited resources, low power and reduced memory space. This also sets operability limits. To secure IoT, the best features of cryptographic algorithms are customized to generate a light weight algorithm suitable for IoT. Light Weight algorithms are compared based on the key size, block size, number of rounds, structure, memory space, execution time, power dissipation, area and cost [13]. AES is used as an engrained solution for COAP inside the application layer. Dearth of human intervention leads to logical attacks. Challenges could be IoT devices related or could be network related [16]. Device related concern are heterogeneous platforms, power limitations, security and privacy. Network related issues are scalability, bandwidth, security and privacy.

B. Literature Review

Among the existing cryptographic algorithms, comparison was done based on the pre-requisites of IoT. With bourgeoning IoT devices, the amount of data generated and transited has reached astronomical heights. Researchers analyzed on unpredictable varying pattern of the heterogeneous user, device and network security constraints which proved unstable. By monitoring data such as carbon monoxide and smoke, presence of the user can be estimated. By monitoring the usage of computer, privacy details could be maltreated. For advertisement, there is an expected risk of the data being shared by the service providers. Hackers can easily get these device sensitive details and modify them as per their needs. Recent solutions focus towards hybrid cryptographic algorithms and data masking. This increases key size and latency. A detailed analysis of the application, data collection, processing, sharing and transfer could make it highly secured. Mobile IoT devices are more likely to share and communicate data with the social network. Android smartphones have limited resources. AES utilizes CPU efficiently [10]. Smartphones are used in Smart Home applications .Number of IoT devices upsurge day by day. With default username and password the devices are remotely monitored and controlled. Device manufacturers do not update the firmware needed for protection against malware threats. Smart Home applications, health and agriculture depend on wireless sensor network which is prone to attacks. Certain smart home applications are web based and mobile application based. Mostly privacy leak is due to interdependence and constrained features of IoT.

C. Analysis of Light Weight Cryptography Algorithms

Asymmetric algorithms are not preferred for IoT. Among symmetric algorithms, based on the hardware features, as shown in Table I, DES block cipher is not preferred due to its 56 bit smaller key size. DES was broken in less than a day using brute force attacks [2]. AES includes the Galois field in each round. AES is preferred because of its large key size which makes it difficult for the hacker to intercept [19]. 3DES is preferred for online transactions but still it is not preferred as it consumes three times as much of CPU power. For hardware implementation, lightweight cipher should occupy less memory space. To calculate area required, gate equivalents (GE) are used. Block cipher should have lesser gate equivalents. Based on the gate equivalents, compared in Fig.1. , PRESENT is the preferred lightweight algorithm. But it requires more number of rounds to make it secure. In view of security and optimization, AES is preferred based on the key length. It's Substitution –Permutation combination makes it less prone to attack [3] which makes it the preferred lean symmetric algorithm.

Table I. Comparison of Symmetric Algorithms (Hardware)

Algorithm	Key Size (bits)	Block Size	Rounds	GE	Attacks
DES	56	64	16	2309	Brute Force
3DES	168	64	48	2168	MITM
AES	128/ 192/ 256	128	10/12/ 14	2400	Less Prone
PRESENT	128	64	31	1884	Differential Attack

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

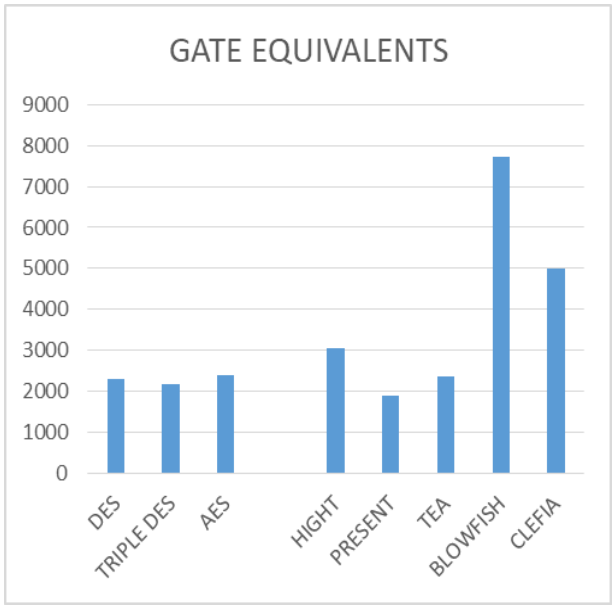

Fig. 1. Comparison of algorithms based on gate equivalents.

As shown in Table II, based on the software features, the number of clock cycles are compared. As the number of clock cycles decreases, total power consumption reduces and the speed of the system increases.

Table II. Comparison of Symmetric Algorithm (Software)

Algorithm	Encryption clock cycles	Decryption clock cycles	Throughput
DES	8,633	8,154	29.6
AES	6,637	7,429	77.1
PRESENT	10,723	11,239	23.7

AES outperforms in both hardware and software and is preferably chained with MD5 (hash) algorithm because of its low memory space requirements, security, hardware software performance, energy efficiency [9] and resistance to implementation attacks.

II. PROPOSED LIGHT WEIGHT IOT SECURITY MODEL

Hybrid Encryption using encryption algorithm and hash function provides confidentiality and integrity. AES is used to generate the key for encryption. Hashing ensures integrity. The proposed model includes AES symmetric algorithm and MD5 hash algorithm.

A. AES Encryption and Decryption

Encryption converts readable plaintext into pseudo random cipher text. There is no single viable solution for IoT systems
as they operate on dissimilar control platforms, protocols, connectivity domains and servers.

Before encryption, the data is arranged in the form of a matrix of bytes or states. AES breaks 128 bit block of data into states or matrix of bytes of predetermined size and further encrypts each state independent of the other. AES has 10 rounds for a 128 bit key. A matrix with 4 rows and 4 columns
is formed with each entry as a byte or 8 bits with a total of 16 bytes. AES performs computation in rounds. The number of rounds depends on the key length. Each round transforms its state. AES algorithm consists of smaller sub algorithms namely Sub-Bytes, ShiftRows, MixColumns and AddRoundKey [8].

Initially [15] it starts with the AddRoundKey transformation and final round has no MixColumn transformation. Other rounds undergo (N-1) transformations. SubBytes transformation, is a non-linear byte substitution where each byte or each element in the matrix is replaced by an S-Box. This is based on a fixed 8 bit look-up table. In ShiftRows, bytes in each row are cyclically shifted to the left and the shift varies with offset. In MixColumns, four bytes of each column of the state are combined using invertible linear transformation. AddRoundKey performs the major encryption process by generating sub keys from the main key. For each round, Subkey of the same size as the state is derived from the main key. Each byte of the state is X-NOR ed with each byte of the Subkey.

Traditional Look-up table S-Box approach, assigns each element of the plaintext (message) to the S-Box. For decryption, the message is retrieved using inverse S-Box and the same procedures are followed as that of encryption. ROM stores pre-computed values as look-up table. Input bytes are wired to the address bus of ROM. ROM has fixed read and write access time. This increases the area and delay. When implemented in hardware, AES results in increased cost.

To overcome this, S-Box transformation is achieved using Galois field. In this method, Composite field arithmetic is applied in the SubBytes portion of AES Algorithm. Representing data as a vector allows the data to be scrambled easily. This involves two sub stages. In the first stage, each byte is replaced by its multiplicative inverse. In the second stage, inverse S-Box is applied to each byte of the state. This pipelining reduces the complexity and drastically increases the speed of the system. Isomorphic mapping converts GF (2^{8}) into lower complex GF (2^{1}) GF (2^{2}) GF $(2^{2)2})$ in order to obtain multiplicative inverse in the SubBytes transformation of AES algorithm. Binary of GF (2^{8}) is GF (2^{8}) is 100011011. The input to encryption is a 128 bit block of data. This is copied into a state array which changes with each stage of encryption. Key is a square matrix of bytes. Each byte in the state matrix is a Galois field.

MD5 Hash Algorithm

MD5 takes up an input message and converts it into a 128 bit message digest. The input message is divided into blocks. The algorithm initializes a vector with a fixed value. The vector value and the first block value are hashed using a compression function. The resulting value is hashed with the next block value and this process continues till the last block.

The last block is padded with zeros. An attacker requires 2^{128} trials to retrieve the message. Hash function is unique for each message. It is not possible to find two different messages
having the same hash function. Hash or message digest is also referred to as digital fingerprint.

III. IMPLEMENTATION

Hardware implementations of cryptography such as smart cards are vulnerable to side channel attacks. Analysis of power consumption could be used to derive the secret key. As a countermeasure, data is masked using S-Box [4]. AES could be programmed in software or could be implemented in hardware. In Galois field, Addition/Subtraction is performed by using the X-NOR operation and Multiplication /Division is performed by using the AND operation. The multiplication product of the polynomials would result in a finite field. Field programmable gate arrays (FPGA) provide customized solutions. Architectural design for encryption and decryption [5] is formulated with FPGA using Very High Speed Integrated Circuit Hardware Description Language (VHDL) coding. Using Verilog, both the algorithms are applied and the message is encrypted. S-Box is based on composite arithmetic field. Design is coded using Verilog, and is subsequently simulated using ModelSim of Xilinx. There are three input clock cycles. Power is validated.

RFID reader emits radio waves at a particular frequency, which in turn powers up the transponder inside the RFID Tag. It gives all the data inside the microchip. The reader receives this data and sends it for processing. RFID Tag obtains it from the reader and applies AES encryption on the data. This generates a cipher text. MD5 is also applied on the information and this generates a 128 bit digest value. The cipher text and the digest function are sent to the reader. In the decryption side, using these, a message is generated. This should match with the message sent by the RFID Tag. Change in just one bit in the hash generates a 50 % wrong message. Efficiency is improved by chaining of these algorithms and by using S-Box.

IV. SIMULATION AND INFERENCES

The vulnerability is usually measured by comparing S-box implemented through a simple look up table and also the S-box derived from the Galois Field. The Verilog HDL for both of these cryptographic algorithms is applied to the Cadence encounter design flow to synthesize, simulate and to get the final Layout and GDSII stream format. Fig. 2 indicates S-Box implementation.

.

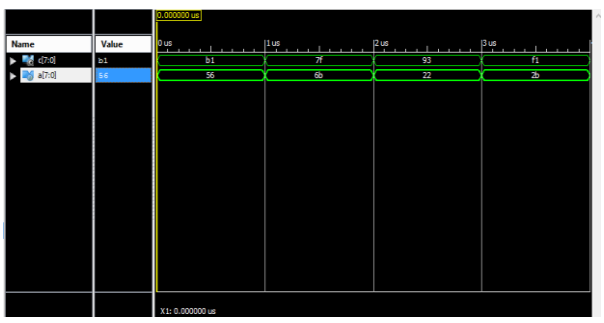

Fig. 2 Implementation of S-Box

Fig. 3 indicates SubBytes Transformation, Fig. 4 indicates ShiftRows transformation, Fig. 5 indicates MixColumn transform, Fig. 6 indicates Encrypted Output and Fig. 7 indicates the encrypted and decrypted output using ModelSim simulation.

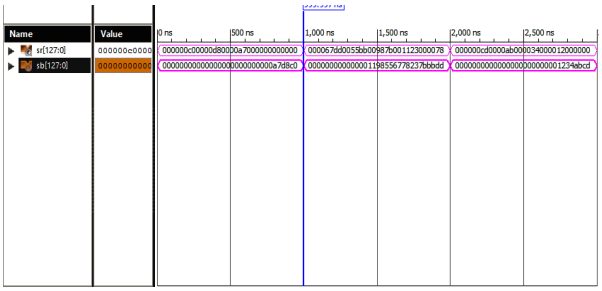

Fig.3. SubBytes Transformation

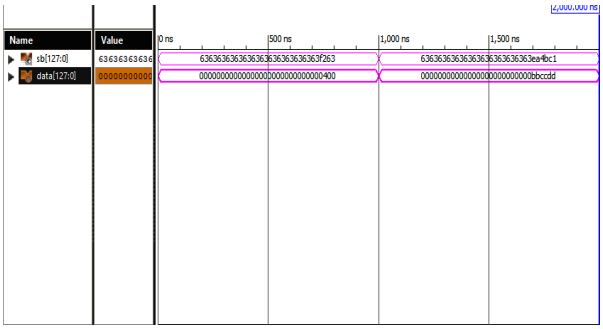

Fig.4. ShiftRows Transformation

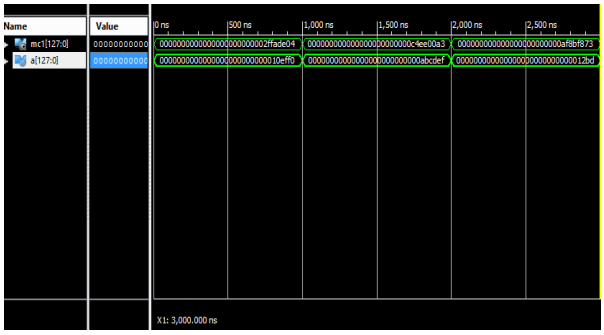

Fig.5 .MixColumn Transformation

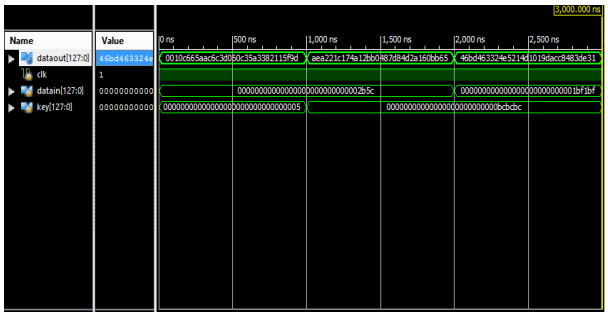

Fig.6. Encrypted Output

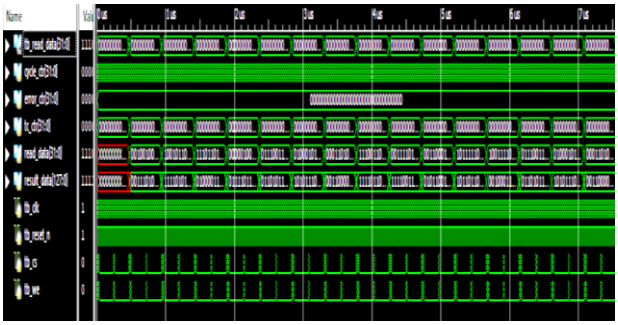

Fig. 7. Encrypted and Decrypted Output.

From VHDL Simulation results, the procedures and inferences are as follows. Input clock is set. Byte Substitution is performed. Reset is set high. 128 bit state is set as std_logic_vector. Waveforms are transformed by RowShift, MixColumn and AddRoundKey transformations. Similarly waveforms are transformed by Inverse ByteSubstitution, Inverse RowShift, Inverse MixColumn and Key Expansion transformations. Waveforms are reversed. Waveforms for the chained Cipher and Decipher light weight IoT algorithm was obtained and analysed. Further power and gate optimization involves utilization of certain commands such as clock gating and operand isolation. The clock gating technique keeps the clock shut down for particular blocks when it is not required. Similarly, operand isolation isolates the block from the rest of the other circuitry when it is not in use. This results in optimized power and area and in addition to this timing slack is also reduced thus resulting in greater speed for encryption and decryption. Once after the synthesis is done, the Verilog file is simulated to check for perfect outputs for the given test bench. The final step involves creating a netlist for the Verilog HDL and it is then incorporated to generate the layout and GDSII stream. For generating the layout, netlist along with Library Exchange Format (LEF) files are also added which provides the tool with wire-load models and cell description.

V. RESULT AND DISCUSSION

Power dissipation is an important design parameter in personal communication systems. It depends on the circuit nodal points, transition of current and leakage current. This is implemented in the circuit level and logic level. VLSI circuits have processors containing Arithmetic units, Control units and Register files. These register files are not accessed in each clock cycle. Power reduction is achieved by gating the clock pulses of such registers. By varying the ordering of the transistors power dissipation is optimized. For power estimation, gate level power analysis was performed. Simulations were performed using ModelSim.

FPGA is used to implement the Light Weight Cryptographic algorithm. It has several configurable logic blocks (CLB) which are interconnected by programmable interconnect.

Chaining of symmetric and hash algorithm increases the number of gate equivalents which further increases the area and power. Using data path optimization, clock gating and operation isolation this drawback is overcome.

The Look-up Table based S-box and Compact Galois field S-box are compared and the parameters such as power, area and delay are analysed. From Table III, we infer that Compact Galois field S-box results in reduced area and power.

Table III: Area, Power and Delay for LUT based vs Galois Field based S-box

Parameters	LUT based S-BOX	Galois Field based S-BOX
Power	17470787.458 (nW)	10278857.050 (nW)
Area	374828.125 (um^2)	351028.339 (um^2)
Delay	8.199 ns	8.85 ns

VI. CONCLUSION

A highly secured data sharing is achieved using lightweight algorithm in colossal IoT. Security and privacy predicts the astronomical growth of IoT.Chaining technique is implemented with two different S-box structures of AES to minimize power, area and delay factors thereby reducing implementation complexity. The advent of compact Galois Field structure minimizes the side channel attacks when compared to that of Look- up table based S-box. This along with the MD5 algorithm further reduce the power consumption and area. The experimental results display the optimized values of power, area and timing factors which are the pre-requisites for an energy efficient enhanced light weight secured algorithm in IoT. This algorithm is tested using Xilinx software using FPGA. This can be extended in hardware implementations in future.

REFERENCE

1. Safi ,A .(2017). 'Improving the Security of Internet of Things using Encryption Algorithms ', *International Journal of Computer and Information Engineering*, vol. 11(5), pp .546-549.
2. Al-Anazi, H., Zaidan, B. B., Zaidan, A. A., Jalab, H. A., Shabbir, M., & Al-Nabhani, Y.(March 2010).' New Comparative Study between DES, 3DES and AES within nine factors', *Journal of Computing* , vol. 2 (3), ISSN 2151-9617 , pp.152-157.
3. Bansod, G., Raval, N., & Pisharoty, N.(January 2015).' Implementation of a new lightweight encryption design for embedded security', *IEEE Transactions on information forensics and security*,vol.10 (1), pp.142-151.
4. Canright, D. (2007).'Masking a compact AES S-box',*Calhoun :Institutional Archive of the Naval Postgraduate School, Monterey CA* ,pp.1-19.
5. Devadas, S., Malik, S. (1995).' A survey of optimization techniques targeting low power VLSI circuits', *Proceedings of the 32nd annual ACM/IEEE Design Automation Conference* .
6. Eisenbarth, T., & Kumar, S. (2007). 'A survey of lightweight-cryptography implementations', *IEEE Design & Test of Computers*, vol.24(6),pp.522-533.
7. Erguler, I., & Anarim, E.(March 2012). 'Security flaws in a recent RFID delegation protocol' , *Personal and Ubiquitous Computing* , vol.16(3),pp.337-349.
8. Hamalainen, P., Alho, T., Hannikainen, M., & Hamalainen, T. D. (2006).'Design and implementation of low-area and low-power AES encryption hardware core' , *9th EUROMICRO Conference on Digital System Design: Architectures, Methods and Tools*, DSD 2006,pp.577-583.

9. Kaps, J.P., & Sunar, B. (2006). 'Energy comparison of AES and SHA-1 for ubiquitous computing' , *International Conference on Embedded and Ubiquitous Computing*, EUC 2006,pp.372-381.
10. Montoyo .B.A.,Munoz G .M., & Kofuji, S. T. (2013).' Performance analysis of encryption algorithms on mobile devices ', *47th International Carnahan Conference on Security Technology (ICCST), 2013*.
11. Misra, S., Maheswaran, M., & Hashmi, S. (2017).' Security challenges and approaches in internet of things', *Springer Briefs in Electrical and Computer Engineering.*
12. Perera, C., Liu, C. H., Jayawardena, S., & Chen, M. (2014). 'A survey on internet of things from industrial market perspective', *IEEE Access*,vol.2,pp.1660-1669.
13. Poschmann, A., Leander, G., Schramm, K., & Paar, C. (2007). ' New light-weight crypto algorithms for RFID', *IEEE International Symposium on Circuits and Systems, ISCAS 2007*.
14. Seth, S. M., & Mishra, R. (June 2011). 'Comparative analysis of encryption algorithms for data communication' ,*International Journal of Computer Science and Technology*,Vol.2 (2),pp.292-294.
15. Stallings, W. (2003). 'Cryptography and network security: principles and practice', *Pearson Education India*.
16. Stankovic, J. A. (2014). 'Research directions for the internet of things', *IEEE Internet of Things Journal*.
17. Suo, H., Wan, J., Zou, C., & Liu, J. (2012).' Security in the internet of things', *International conference on Computer Science and Electronics Engineering (ICCSEE)*,pp.648-651.
18. The Statistics Portal. (2017). Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions). [Online]. Available: https://www.statista.com/statistics/471264/iot-number-of-connected-dev ices-worldwide/
19. Zodpe, H. D., Wani, P. W., & Mehta, R. R. (2012). 'Design and implementation of algorithm for DES cryptanalysis', *12th International Conference on Hybrid Intelligent Systems (HIS)*,pp.278-282.

AUTHOR PROFILE

B. Nagajayanthi is currently serving as Associate Professor, in the School of Electronics Engineering, Vellore Institute of Technology, Chennai Campus. She has 20 years of academic experience. Her research area expertise is in the field of Wireless Networking, Network Security, IoT, Analog Communication Systems and Digital Communication Systems.