Encryption Based on Cellular Automata for Wireless Devices in IoT Environment

Harinee Shanmuganathan and Anand Mahendran*
School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India
anandmahendran82@gmail.com

Abstract: *A large number of physical objects are interconnected and accessed through the internet along with the existing technology like cloud computing, mobile computing, wireless sensor networks and big data forms a big paradigm called the Internet of Things (IoT). Data from remote and neglected areas are collected via wireless sensors and stored in the cloud. Growth in the number of the sensing device and the server to which these are connected leads to many security issues and malicious attacks. With mirai botnet and jeep hack about 150 million user's information from MyFitnessPal nutrition app was stolen. This paper mainly focuses on security challenges for transmitting the data in the wireless sensor network in the IoT environment. To avoid the malicious attack, eavesdropping, algebraic attacks and other attacks a new security algorithm is proposed based on Cellular Automata (CA) which has a key length of 80 bits and the text size is 64 bits. Reversible rules of CA are used in this algorithm to achieve reversibility, parallelism, stability, randomness, and uniformity. The process of encryption and decryption is performed for 15 rounds to avoid dependency between the ciphertext and the plain text. Finally, we compare the execution time, throughput and the avalanche effect of the proposed algorithm with the existing algorithm like Advanced Encryption Standard (AES), Height, Present algorithm. The proposed algorithm is verified to be a better choice for lost cost and resource-restricted devices.*

Keywords: *Cellular Automata, secure data transmission, security algorithm, IOT.*

Received May 25, 2020; accepted September 27, 2020
https://doi.org/10.34028/iajit/18/3/11

1. Introduction

The main factor of computing is the sharing of information which is done by ubiquitously connecting the smart device (like Philips hue smart light bulbs, Amazon echo dot, etc.,) through a network. With increase and development in wearable devices (like smart-bands), smart homes (Google Home, SmartMat), factories (like SiteWatch, DAQRI Smart Glasses), cities (like Noise Urban Maps, Smart Grid), hospitals (like HoloLens) and vehicles (like Fleet Management by Bransys, Parking management by ParqEx) connected through wireless sensor networks, which makes everything connected and smart, is said to be the Internet of things. Based on the choice of communication (wireless or wired) the requirement and the cost for the devices change. When IoT is considered, the sensors and actuators deployed and scattered over a wide area like in buildings, clothing, campuses, factories, houses, and even in the body, etc and are connected via a wireless network [6]. A network of distributed sensor which monitors and collects information of the environment or physical condition like humidity, sound, etc and send the information through a network is said as a wireless sensor network.

The number of devices connected will drastically increase from 20.35 billion in 2017 to 75.44 billion in 2025 worldwide according to Statista Statist [30]. With the increase in the number of devices, securely sharing the data becomes a problem. When a patient's implant medical devices [6] or smart house [13] security can be compromised, it not only leads to an economic loss to an individual but is also as a threat to that person. Recently, IoT devices are widely used in military, medical fields [22, 26], industries [32, 34], etc.

Apart, from the growth and usage of Internet of Things (IoT), there are also many security threats. The connected devices in the IoT environment are prone to attack because of the following reasons

- It is easy for the attacker to gain physical access to IoT devices, as most of them are operated without human intervention.
- As wireless networks are means of communication for IoT devices attackers can eavesdrop the network to obtain confidential data.
- IoT devices tend to have low power and low computation capabilities which in turn does not support complex security schemes.

In October 2016, an attack based on botnet which contained a large number of IoT devices like IP cameras, baby monitors, etc was carried out through Domain Name System provider Dyn. They were attacked by multiple Distributed Denial [35] Of Service (DDoS) which lead to the inaccessibility of

many websites like Twitter. Yet another incident, that targeted industrial computers based on a malicious worm called Stuxnet [19] lead the way to substantial damage to Iran's nuclear program. Efforts have been taken for addressing the security issues in IoT. Authentication and Authorization for Constrained Environments (ACE) and Internet Engineering Task Force (IETF) are working to provide world standards for privacy and security in IoT devices. For sharing the data securely among the IoT devices, an encryption algorithm is required. This algorithm must overcome the limitation of IoT devices like limited battery life and limited computing capabilities [17]. Most of the IoT devices have only a few megahertz of CPU, a few kilobytes of RAM, and several dozen kilobytes of ROM. In this environment, the device needs to operate robustly and must have a high level of security. To satisfy all the limitations of IoT devices it is required to design a lightweight algorithm like Present, Height, Lblock, etc., for low resource devices based on their resource, a new cellular automata-based encryption algorithm has been designed. The concept of cellular automata is used for this purpose as it has properties like parallelism, adaptive in nature, produces aperiodic and chaotic patten and it is discretely dynamic with simple construction but complex and varied behaviour.

The outline of the paper is as follows. Section 2 discusses the drawbacks of wireless network devices in the IoT environment and various security algorithms that can be used for wireless devices. Section 3 deals with research motivation and the drawbacks of other algorithms. Section 4 describes cellular automata and its rules. Section 5 proposes cellular automata-based encryption algorithm. It further discusses the Cellular Automata (CA) rule layer, key scheduling and selection processes. Section 6 deals with the decryption part of the proposed algorithm. The results are graphically represented and the proposed algorithm is compared with the existing algorithms in section 7. The paper concludes with scope for future work.

2. Literature Survey

To accept any technology the most important aspects are security and a sense of confidence. In an IoT environment the data privacy, security, and data compromise are the most important concern. To attain this, IoT needs to go through many challenges like compatibility, complexity, privacy, etc., Authentication problem, the capability of the device, vulnerability mismanagement, capability of the device, insecure communication, and others are the major factor that affects the growth of IoT and are shown in Figure 1 [14, 21].

The wireless network is the main source of communication for sensors in IoT architecture. With less effort, the attacker can indulge in interruption-based attacks (like overloading a server host so that it cannot respond), interception-based attacks (like eavesdropping) and modification attack (like reconfiguring system hardware or network topologies) in wireless networks. Thus, it leads to a compromise in the IoT environment and leads to privacy and security issues. To overcome this compromise, cryptographic algorithms like Data Encryption Standard (DES), Advanced Encryption Standard (AES), SPONGENT and so on are used [8].

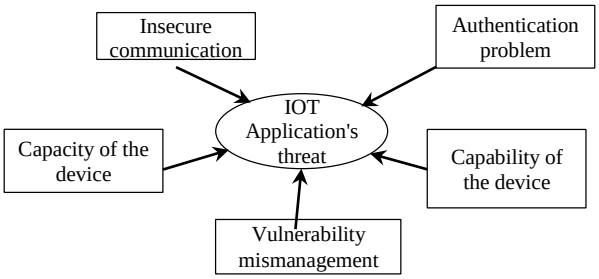

Figure 1. Threats in the IoT environment [1].

But according to Trappe *et al*. [31], there are reasons like the battery life of sensors, etc that makes it impossible to use features of traditional internet for securing the IoT device communication.

The conventional encryption algorithm AES needs a lot of memory space for computation of ciphertext from plain text. But this memory capacity cannot be provided by IoT devices. MIKEY [3] key management protocol was designed to facilitate multimedia distributions in Wireless Sensor Network (WSN) but it lacks the characters for comparability in the IoT environment. For authenticated broadcast, TESLA was proposed [25]. To lower the cost of communication by rekeying the messages in WSN Topological Key Hierarchy (TKH) [28] was used. But its cost increases linearly with an increase in the number of devices in IoT. Harn and Lin [15] and Lee *et al*. [20] proposed a protocol for secret sharing in WSN based on pairing-based computing. But both protocols are not a prevalent cipher for IoT devices.

The main challenge in the IoT environment is providing security, protection of data, the trustworthiness of data and confidentiality. Not only that there few other problems like logical and physical attack due to very less human intervention, IoT WSN also has vulnerabilities like eavesdropping, DoS attack and reply attack etc and the IoT devices has resources constraints like low memory space, limited power supply and heterogenous platform etc. To address all these issues a conventional algorithm is not suitable. Table 1 shows that the traditional protocols that are used in WSN lack security when it comes to the IoT environment. Not only that the complexity at a different layer of IoT, for setting up of cryptographic processes is different. To avoid theses weakness lightweight cipher is used. But to deliver a high level of security live traditional algorithm, lightweight algorithms have to increase the number of rounds for

confusion and diffusion during encryption and decryption processes. With an increase in the number of rounds the performance decreases. So, the research work mainly focuses on this. For this cellular automaton gave the solution with its property like randomness, aperiodic patterns, uniformity, unpredictability and parallelism etc., This led to the proposal of a lightweight encryption algorithm based on cellular automata.

Table 1. Describes about different algorithm along with its block size, key size, structure, number of rounds, merits, where they are applied and the attacks they suffer.

Algorithm	Block size	Key size	Number of rounds	Structure	Dependency	Application	Merits	Attacks
AES [23]	128	128	10	SPN	S/w	Not given	Supports larger key sizes, faster in both hardware and software.	Related key attack, Boomerang, Biclique cryptanalysis
ECC [11]	256	163/571			H/w	Pervasive computing	Increased speed, less memory requirement, optimum security level.	Side channel attacks.
HIGHT [16]	64	128	32	FN	H/w	RFID using FPGA	Ultra-lightweight, provides high security, good for RFID tagging.	Biclique cryptanalysis, Impossible differential attack on 26th round.
PRESENT [7]	64	80/128	32	SPN	H/w	RFID	Ultra-Lightweight, Energy efficient.	Side-channel attacks, truncated differential cryptanalysis, Integral, Bottleneck attacks,
SIMON [4]	128	128	64	SPN	H/w	ASIC application	Supports several key sizes, performs well in hardware.	Differential fault attacks, Attacks on reduced versions
SPECK [10]	128	128	32	SPN	S/w	Not given	Performs better in software	Key Recovery, Boomerang attack
TWINE [29]	64	80/128	36	FN	H/w	Not given	Good for small hardware, efficient software performance	Middle attacks Saturation Attack

- In this table SPN refers to substitution–permutation network:
- FN is Feistel Network is given.
- FPGA is Field-Programmable Gate Array is given.
- RFID is Radio-Frequency Identification is given.
- ASIC is application-Specific Integrated Circuit is referred.

3. Research Motivation

The vulnerability of IoT devices to attack is nearly 70% according to the study made by Hewlett-Packard Company (HP) [18]. The main problem in the IoT environment is secure communication. The man-in-middle attack can be accomplished by sensing the communication link between two nodes. No reliable solution has been proposed to capture such attacks. The use of encryption may lead in minimizing the damage done to the data. When it comes to sensors the capacity for computation is less and a large amount of data needs to be gathered and sent in less time. And yet the data transmitted must be in a secure manner. To assure data integrity while transmission and during storage in middleware leads to security mechanisms. Various encryption algorithm has been proposed but when it comes to IoT usage most of the algorithm tends to fail because of its computation capabilities. Hence, we tend to use cellular automata. CA has the property of homogeneity, reversibility, unpredictability, parallelism, and implementation of CA is easy in hardware and software. This makes it more appropriate for designing a cryptographic algorithm.

4. Cellular Automata (CA)

A cellular automata is a collection of infinite cell in grid structure [5] that can change from one state to another, based on transition function and it neighbours. A cellular automata is defined as a 4-tuples (i.e., given in Equation (1): number of cells (dimension), neighbourhood, set of finite states and set of rules.

$$A = (S, Z^d, f, V) \quad (1)$$

Where
S- finite set of states.
Z- set of integers.
Z^d - cell space of the automata.
V- set of neighbourhoods V= (v_1, v_2, v_3, ..., v_n)
f- transisiton function.

In one-dimensional CA, when three cells are considered with the finite state such as 0 and 1 there are 2^3= 8 possible binary states. For such conditions, there are 2^8=256 possible elementary CA rules (or the Boolean functions). Let us consider A_i be the present state of the cell *i* (where *i* can be 0 or 1) at time *t*, the value of A cell at *t+1* is a functional value of its neighbourhood cells at *t* [37]. It is mathematically represented as

$$A_i^{t+1}=f_i\,(A^t_{i-1}, A_i^t, A^t_{i+1}) \quad (2)$$

CA mechanism was first used by Wolfram for encryption technique using Rule 30 (used as random

number generator) [36]. Some of the CA rules are listed in Table 2. In Table 2 the three bits in *PS* represents the present state of the cell with its neighbourhood cells. The first bit represents the left neighbourhood state of the cell at *t*, the second bit represents the current cell state at *t* and the third bit represents the right neighbourhood state of the current cell at *t*. The *NS* represents the state of the current cell in *t+1* time. There are many kinds of cellular automata like fuzzy CA, programmable CA, periodic boundary CA, etc., One among them is reversible CA. When a CA can return to its initials state is termed as reversible CA [27]. This property of CA is used in the proposed algorithm.

Table 2. Cellular rules.

***PS* (x,y,z)**	**111**	**110**	**101**	**100**	**011**	**010**	**001**	**000**	**Rule No**
NS	0	0	0	1	1	1	1	0	30
	0	0	1	0	1	1	0	1	45
	0	1	0	1	1	0	1	0	90
	1	1	0	1	1	0	1	0	218

Here in Table 2, PS and NS represent present state and next state respectively x, y and z represent the left neighbourhood cell state, current cell state and right neighbourhood cell state respectively.

Uniform CA also known as Linear CA is where the rule is applied uniformly on a data matrix of cells. All the cells in the matrix get operated with the same rule as CA rule 60, rule 90, rule 102, etc., Non-Uniform CA also is known as hybrid CA is one in which all the cells of the matrix have their own local rule that may be different from the rule applied to other cells of that matrix. Some examples of non-Uniform CA are rule 15, rule 85, rule 204, etc.

5. The Encryption Algorithm

Encryption is the process of converting the plain text into ciphertext. The principle objective of the proposed system is to provide authenticated cryptographic security based on CA. The proposed system is a block cipher-based encryption. We aim to provide a high level of security against tampering by the attackers, for which linear as well as nonlinear CA rule. The proposed encryption algorithm makes use of an 80-bit key length, 64-bit plain text and 15 rounds, it is shown in Figure 2. The algorithm consists of two main parts expansion of key and iteration with CA rules.

5.1. Working of the Proposed System

The algorithm consists of two main parts expansion of key and iteration with CA rules. Each round in this algorithm has two steps, they are key updating layer and nonlinear CA rule. The proposed algorithm general view is given in Figure 2.

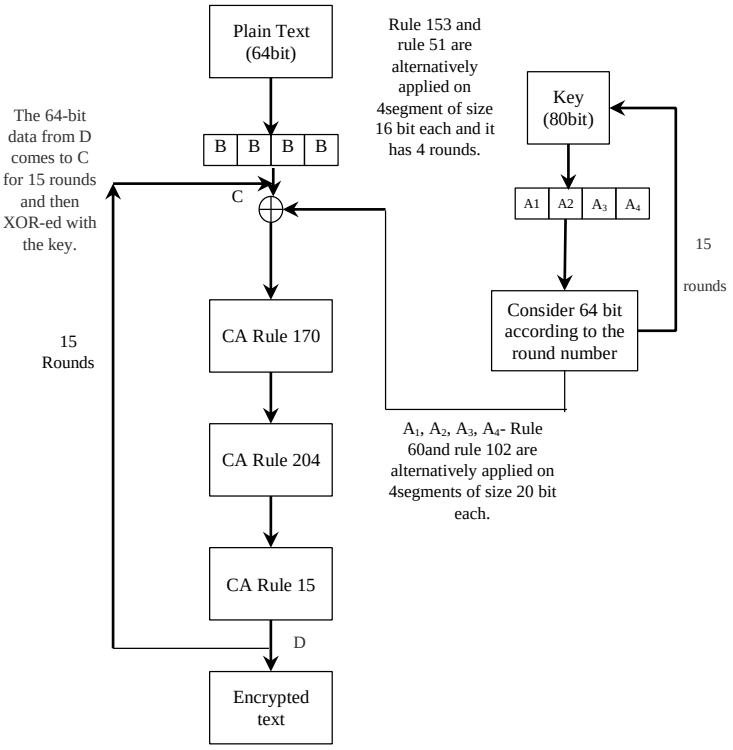

Figure 2. Encryption process of the proposed algorithm.

5.1.1. Key Scheduling and Selection

For key scheduling it is assumed that automata array consists of R_1, R_2, R_3, ……., R_n cells for each segment, the extreme cells are adjacent to each other forming a cyclic condition, which is simply periodic boundary condition. In key scheduling the 80-bit key is updated for each round. The key is split into 4 segments of size 20 bits each. CA rule 60 and 102 are applied in alternative segments. If the key is split into A1, A2, A3 and A4 segments, then CA rule 60 is applied to the A1 and A3 and for A2 and A4 CA rule 102 is applied. In this 80-bit key, only the first 64-bit, i.e., key_{79}, key_{78}, key_{77}, ..., key_{16} are considered for odd round of encryption and the last 64 bit i.e., key_{63}, key_{62}, key_{61}, ..., key_0 are considered for even rounds. The selected keys from each round is considered for xor operation. The key scheduling algorithm is given in Algorithm 1.

Algorithm 1: Key Scheduling and Selection Algorithm

INPUT: $key_r[] \rightarrow key_{79}, key_{78}, key_{77}, ..., key_0$ *(80-bit key)*
OUTPUT: $key_s[]$ *(64-bit key)*
1: Procedure
2: for each round i: 1→15 do
3: Split the key into 4 segments of size 20 bits each.
4: Apply CA rule 60 and 102 in alternative segment
5: if i is odd then
6: for each j: 1→ 64 do
7: keys []→ keyr[j]. Selecting leftmost 64 bit for encryption.
8: end for
9: Else
10: for each j:16 →80 do
11: Keys[]→keyr[j]. Selecting rightmost 64 bit for encryption.
12: end for

13: end if
14: end for
15: end procedure

5.1.2. CA Rule Layer

The most important property of encryption is confusion and diffusion. These two properties can be achieved using CA non-linear (CA rule 15, rule 85, rule 204, etc.,) and linear rules (CA rule 60, rule 90, rule 102, etc).

To obtain a better cryptographic algorithm minimum number of liner rules and many numbers of non-linear rules [27] are used in the proposed algorithm. The rules used in the proposed algorithm are described in Table 3.

Table 3. Cellular rules.

Rule No	BooleanFunction
15	$A_i^{t+1} = A_{i-1}^t$
51	$\overline{A_i^{t+1}} = A_i^t$
60	$A_i^{t+1} = A_{i-1}^t \oplus A_i^t$
85	$A_i^{t+1} = A_{i+1}^t$
102	$A_i^{t+1} = A_i^t \oplus A_{i+1}^t$
153	$A_i^{t+1} = \overline{A_{i+1}^t \oplus A_i^t}$
170	$A_i^{t+1} = A_{i+1}^t$
204	$A_i^{t+1} = A_i^t$
240	$A_i^{t+1} = A_{i-1}^t$

5.2. Proposed Algorithm

In the proposed algorithm, the plain text is diffused based on the additive rule of CA. The plain text is split into 4 (B_1, B_2, B_3, B_4) segments each of size 16 bits. It is assumed that for these four segments the extreme cells are connected to logical 0, which in turn is the null boundary condition. CA Rule 153 is applied to odd segments (B_1 and B_3) with null boundary condition. CA rule 51 is applied to even segments (B_2 and B_4).

CA rule 153 and 51 are linear rules that make the plain text diffused at the beginning. And this process of diffusion is carried out for 4 rounds. Then the key from the key scheduling part is XORed with the text. When the text is received after XORing with the key, a reversible CA rule is applied and it is assumed to have periodic boundary condition. Reversible CA rule 170 is first applied followed by CA rule 204 and at the end stage rule 15 is applied, this is given in Algorithm 2. The application of reversible rule gives the ciphertext the property of nonlinearity (or confusion). The plain text is xor-ed with the key and the application of reversible CA rule are done for 15 rounds. The algorithm is iterated for 15 times to avoid the dependency of ciphertext with the plain text.

Algorithm 2: Encryption Algorithm

INPUT:P 64-bit plain text.
key_s[] 64-bit key, from key scheduling algorithm.
OUTPUT: c 64-bit cipher text.
1: Procedure
2: Split P into 4 segments of each size 16-bits.
3: for each round i: 1→4 do
3: for each odd segment do
4: Apply CA rule 153
5: end for
6: for each even segment do
7: Apply CA rule 51
8: end for
9: end for
10: Combine the segments to get a new plaint text P of size 64-bits.
11: for each round i: 1→15 do
12: Ps = Ps⊕keys[]
13: apply CA rule 170 followed by 204 and then 15 to P
14: end for
15: end procedure

6. Decryption Algorithm

The processes of encoding the ciphertext to plain text are called as decryption. The same processes of encryption are done for decryption, but in the reverse order and the inverse of the CA rule 170, 204 and 15 are applied and this algorithm is given in Figure 3. The decryption algorithm is given in Algorithm 3. The inverse rule for 170, 204 and 15 are 85, 204, and 240 respectively.

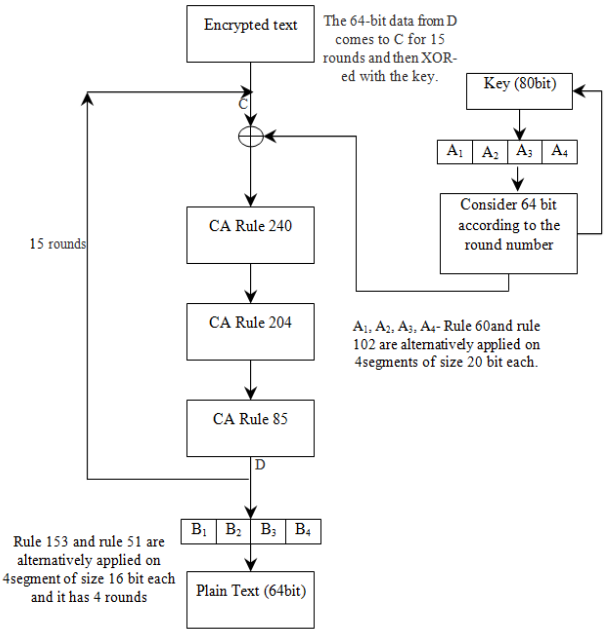

Figure 3. Decryption processes of the proposed algorithm.

Algorithm 3: Decryption Algorithm

INPUT: c \\ 64-bit cipher text.
key_s [] \\ 64-bit key, from key scheduling algorithm.
OUTPUT: c \\ 64-bit plain text.
1: Procedure
2: for each round i: 1→15 do
3: Cs = Cs⊕keys[]
4: apply CA rule 240 followed by 204 and then 85 to C
5: end for
6: plit C into 16 segment of each size 4 bit
7: for each round i: 1→4 do
7: for each odd segment do
8: Apply CA rule 153
9: end for
10: for each even segments do

11: Apply CA rule 51
12: end for
13: end for
13: Combine the segments to form p of size 64-bits.
14: end procedure

7. Implementation and Result

The problem of encryption and decryption for devices in the IoT environment is reviewed in the background work. The limitation of those algorithms is presented in Table 1. The code is implemented in C language and the sample code is provided [12]. The proposed algorithm is compared with Klein, LBlock, Present, Height, and AES algorithms. The time complexity of the algorithm is $O(n^2)$, i.e., it can be executed in polynomial time. Throughput, execution time for encryption of the algorithm and avalanche effect are considered as the metrics for evaluation of the algorithm.

7.1. Execution Time

The execution time of an algorithm is defined as the time taken for converting the plain text to ciphertext and vice versa. For an IoT device, this parameter is considered important, as they have less power supply and the execution of the algorithm must be quick and secure. The comparison of the proposed algorithm with other algorithm is given in Figure 4. The proposed algorithm consumes less time for execution and thus satisfies the power and time constrain in IoT device.

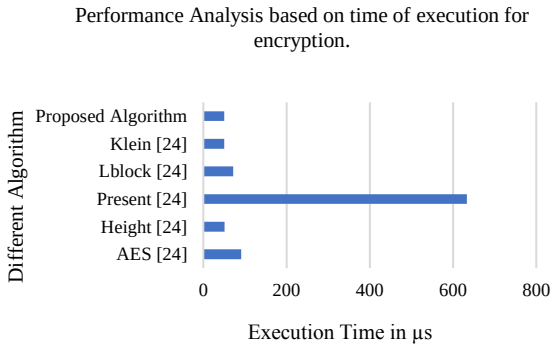

Figure 4. Time for execution of encryption algorithm.

7.2. Throughput

The throughput of an algorithm is defined as the ratio between the total plain text in bytes to the encryption time in milliseconds. The throughput of the encryption is represented as below,

Throughput of an algorithm

$$\eta = \frac{\sum_{i=0}^{n} D_n}{T} \quad (3)$$

Where
η -Throughput of an algorithm
D_n- data in kilobytes
T - Encryption time in milliseconds

When compared to Klein and Present algorithm the proposed algorithm has better performance and it can be seen in Figure 5. AES gives highest performance when compared to other algorithms.

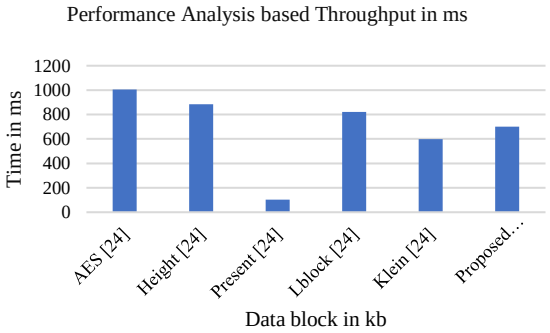

Figure 5. Throughput for encryption algorithm.

7.3. Key Sensitivity Analysis

The key plays a vital role in encryption and decryption processes. Even a trivial change in the bits of the key, must not lead to the recovery of the original plain text. For this to be verified avalanche effect is used as a metric for this. Avalanche effect is defined as the change in a single bit of the plain text or the key will change many bits in the corresponding ciphertext.

The avalanche effect must be greater than 50% for a better cryptographic algorithm [9]. If the avalanche effect is high then many attacks can be resisted. The avalanche effect for the proposed algorithm and other algorithm is shown in Figure 6. It is comparatively more when compared to the other algorithms like Present and LBock.

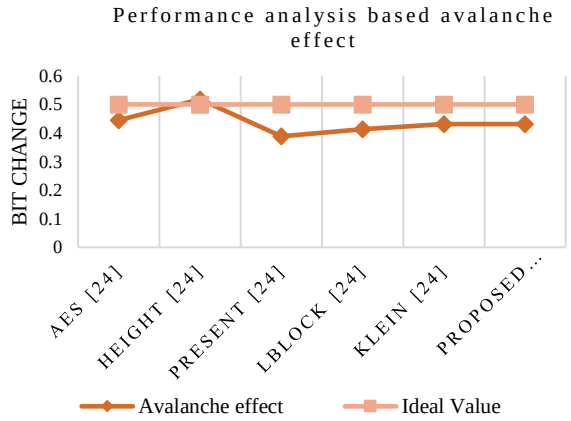

Figure 6. Avalanche effect for different algorithm.

7.4. Security Analysis

This section of the paper deals with the evaluation of the proposed algorithm based on security attacks. In differential cryptanalysis, the attacker makes use of selected plain text and for each set of encryptions, a slight modification in the platin text is made to obtain the equivalent ciphertext which makes a pair. After analysing several such pairs of similar key can be

obtained. To avoid this, the algorithm must respond drastically for a slight modification in the plain text or in the key. The proposed algorithm has an avalanche effect higher than forty-two percentage, which means the algorithm has a high response to a slight change in key or plain text.

7.4.1. Confusions

Confusion technique makes the relation between the key and ciphertext more complex and non-linear in nature. In the proposed algorithm, the output of each round depends on the previous output and the selected key from the key scheduling algorithm. The key scheduling and the algorithm are mainly based on linear and non-linear rules of CA. This results in a highly non-linear generation on ciphertext. Henceforth the algorithm is prevented from linear cryptanalysis attacks.

7.4.2. Diffusion

Diffusion method helps in making the ciphertext dissimilar from the plain text by scattering the redundant bit of the plain text. This property is achieved by automata by XOR/XNOR-ing the neighbour cells and the cell itself. The algorithm uses a null boundary for CA rule 153 and 51 for encryption and decryption processes. It is observed that from the initial processes the ciphertext depends only on key and plain text. At each round of the algorithm, each output bit involves 64 key bits and 64 input bits. So, the degree of a polynomial would be 128 for each bit at each round. Higher the order of diffusion and the more the number of rounds, the algorithm is resistant to attacks.

7.4.3. Boomerang Cryptanalysis

In this attack, the attacker tries to generate a quartet structure in the midway through the cipher. The cipher is divided into two half p and q where p is the first half p and q is the second half. A valid discriminator should satisfy $(pq)^2>2^{-N}$ [33] where N is the number of plain text pairs. A ten round boomerang discriminator is constructed by using five round differential trails. The probability of each trial is $p=q=(2^{-2})^{10}=2^{-20}$ Therefore the total probability for eight rounds is $(2^{-20}*2^{-20})^2=2^{-80}$. This is very low when compared to 2^{-64}. Thus, the proposed algorithm is resistant to boomerang attacks.

7.4.4. Interpolation and Algebraic Attack

The key weakness of the algorithm is the algebraic property. This is exploited by interpolation attacks. Fortunately, the combination of cellular automata reversible rules and cellular automata additive rules makes the result of the proposed algorithm to resist algebraic and interpolation attack.

7.4.5. Weak Key

The algorithm in which the non-linear operation depends on the original key to generate the cipher text is spotted to have a weakness. However, in the proposed algorithm for each round the key is right-shifted and only selective bits are selected for the encryption processes. So, the algorithm doesn't produce a weak key.

7.4.6. Timing Analysis Attack

Analysing the power or the time for execution an algorithm may lead in the compromise of the encryption algorithm [2]. This attack is possible only if the algorithm is data dependant. The proposed algorithm makes use of the three-neighbour periodic boundary. This results in no change in the time requirement of each round based on the number of input or output bits.

The main advantage of the proposed system is it can generate the cipher text in polynomial time. The logical circuit that represents the CA rules can be easily combined to produce the hardware implementation of the proposed algorithm. The strength of the algorithm is based on the fact that for each round of encryption the key changes. In addition to this algorithm has better avalanche effect and less execution time. This led to a better fit of the proposed algorithm for IoT environment.

8. Conclusions

This paper presents an efficient lightweight encryption algorithm that makes use of cellular automation. The algorithm was developed mainly focusing on the IoT architecture at the perception layer where sensor to wireless network communication takes place. The performance of the algorithm is also compared with some algorithms like AES, Present, etc., As this algorithm has a better avalanche effect it can withstand various attacks like ciphertext attack, chosen plain text attack, etc. In the future, the algorithm can be extended by making use of two-dimensional cellular automata. And the concept of cellular automata can be used for a key exchange algorithm also.

Acknowledgement

The authors thank Vellore Institute of Technology (VIT) for providing 'VIT SEED GRANT' for carrying out this research work.

Reference

[1] Anurupam K., "Dynamic S-box implementation in PRESENT Cipher" *in International Journal of Computer Sciences and Engineering*," vol. 6, no. 9, pp. 426-431, 2018.

[2] Arjunan A., Narayanan P., and Ramu K., "Securing RSA Algorithm against Timing Attack," *The International Arab Journal of Information Technology*, vol. 13, no. 4, pp. 471-476, 2016.

[3] Arkko J., Carrara E., Lindholm F., Naslund M., and Norrman K., *MIKEY: Multimedia Internet Keying*, in RFC 3830, 2004.

[4] Beaulieu R., Shors D., Smith J., Treatment-Clark S., Weeks B., and Wingers L., "Simon and Speck: Block Ciphers for the Internet of Things," *IACR Cryptology ePrint Archive*, pp. 585, 2015.

[5] Bern M., Flaherty J., and Luskin M., *Grid Generation and Adaptive Algorithms*, Springer Link, 1999.

[6] Big think Edge. Hacking the Human Heart, http://bigthink.com/future-crimes/hacking-the-human-heart, Last Visited, 2016.

[7] Bogdanov A., Knudsen L., Leander G., Paar C., Poschmann A., Robshaw M., Seurin Y., and Vikkelsoe C., "PRESENT: An Ultra-Lightweight Block Cipher," *in Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems*, Vienna, pp. 450-466, 2007.

[8] Bogdanov A., Knežević M., Leander G., Toz D., Varıcı K., and Verbauwhede I., "Sponging: A Lightweight Hash Function," *in Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems*, Nara, pp. 312-325, 2011.

[9] Castro J., Sierra J., Seznec A., Izquierdo A., and Ribagorda A., "The Strict Avalanche Criterion Randomness Test," *Mathematics and Computers in Simulation*, vol. 68, no. 1, pp. 1-7, 2005.

[10] Duka A. and Genge B., "Implementation of SIMON and SPECK Lightweight Block Ciphers on Programmable Logic Controllers," *in Proceedings of 5th International Symposium on Digital Forensic and Security*, Tirgu Mures, pp. 1-6, 2017.

[11] Eisenbarth T., Kumar S., Paar C., Poschmann A., and Uhsadel L., "A survey of Lightweight Cryptography Implementation," *in IEEE Design and Test of Computers*, vol. 24, no. 6, pp. 522-533, 2007.

[12] Encryption Based on Cellular Automata for Wireless Devices in Iot Environment Available: https://smartharinee.github.io/Encryption-/, Last Visited, 2020.

[13] Envista Forensics. The Most Hackable Cars on the Road., Available:http://www.envistaforensics.com/news/the-most-hackable-cars-on-the-road-1, Last Visited, 2015.

[14] Frustaci M., Pace P., Aloi G., and Fortino G., "Evaluating Critical Security Issues of the IoT World: Present and Future Challenges," *in IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2483-2495, 2018.

[15] Harn L. and Lin C., "Authenticated Group Key Transfer Protocol Based on Secret Sharing," *IEEE Transactions on Computers*, vol. 59, no. 6, pp. 842-846, 2010.

[16] Hong D., Sung J., Hong S., Lim J., Lee S., Koo B., Lee C., Chang D., Lee J., Jeong K., Kim H., Kim J., and Chee S., "HIGHT: A New Block Cipher Suitable for Low Resource Device," *in Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems*, Yokohama, pp. 46-59, 2006.

[17] Køien G., "Reflections on Trust in Devices: An Informal Survey of Human Trust in an Internet-Of-Things Context," *in Wireless Personal Communications*, vol. 61, no. 3, pp. 495-510, 2011.

[18] Kumar S., Vealey T., and Srivastava H., "Security in Internet of Things: Challenges, Solutions and Future Directions," *in Proceedings of 49th Hawaii International Conference on System Sciences*, Koloa, pp. 5772-5781, 2016.

[19] Langner R., "Stuxnet: Dissecting a Cyber warfare Weapon," *in IEEE Security and Privacy*, vol. 9, no. 3, pp. 49-51, 2011.

[20] Lee C., Wang Z., Harn L., and Chang C., "Secure Key Transfer Protocol Based on Secret Sharing for Group Communications," *in IEICE Transactions on Information and Systems*, vol. 94, no. 11, pp. 2069-2076, 2011.

[21] Mehta D., "Internet of Things: Applications and Challenges," *in International Journal of Computer Sciences and Engineering*, vol. 6, no. 8, pp. 289-293, 2018.

[22] Michaelsen K., Sanders J., Zimmer S., and Bump G., "Over-Coming Patient Barriers to Discussing Physician Hand Hygiene: Do Patientsprefer Electronic Reminders to Other Methods?" *Infection Control and Hospital Epidemiology*, vol. 34, no. 9, pp. 929-934, 2015.

[23] Moradi A., Poschmann A., Ling S., Paar C., and Wang H., "Pushing the Limits: A Very Compact and a Threshold Implementation of AES," *in Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Tallinn , pp. 69-88, 2011.

[24] Pei C., Xiao Y., Liang, W., and Han X., "Trade-Off of Security and Performance of Lightweight Block Ciphers" *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, pp. 1-18, 2018.

[25] Perrig A., Song D., Canetti R., Tygar J. D., and Briscoe B., "Timed Efficient Stream Loss-Tolerant Authentication (TESLA): Multicast Source Authentication Transform Introduction," *in RFC*, vol. 4082, pp. 1-22, 2005.

[26] Sensors Remind Doctors to Wash Up, https://www.ibm.com/blogs/research/2013/11/sensors-remind-doctors-to-wash-up/, Last Visited, 2013.

[27] Seredynski M. and Bouvary P., "Block Cipher Based on Reversible Cellular Automata," *New Generation Computing*, vol. 23, pp. 245-258, 2005.

[28] Son J., Lee J., and Seo S., "Topological Key Hierarchy for Energy-Efficient Group Key Management in Wireless Sensor Networks," *Wireless Personal Communications*, vol. 52, no. 2, pp. 359-382, 2010.

[29] Suzaki T., Minematsu K., Morioka S., and Kobayashi E., "TWINE: A Lightweight Block Cipher for Multiple Platforms," *in Proceedings of International Conference on Selected Areas in Cryptography*, Windsor, pp. 339-354, 2012.

[30] The Statistics Portal., Internet of Things (IoT) Connected Devices Installed Base Worldwide from 2015 to 2025, https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/, Last Visited, 2020.

[31] Trappe W., Howard R., and Moore R., "Low-Energy Security: Limits and Opportunities in the Internet of Things," *in IEEE Security Privacy*, vol. 13, no. 1, pp. 14-21, 2015.

[32] Ungurean I., Gaitan N., and Gaitan V., ''An IoT Architecture for things from Industrial Environment,'' *in Proceedings of 10th International Conference on Communications*, Bucharest, pp. 1-4, 2014.

[33] Wagner D., "The Boomerang Attack," *in Proceedings of International Workshop on Fast Software Encryption*, Rome, 156-170, 1999.

[34] Wang C., Bi Z., and Xu L., ''Iot and Cloud Computing in Automation of Assembly Modelling Systems,'' *IEEE Transactions on Industrial Informatics*, vol. 10, pp. 1426-1434, 2014.

[35] Wikipedia. Dyn Cyberattack, https://en.wikipedia.org/w/index.phptitle, Last Visited, 2016.

[36] Wolfram S., *A New Kind of Science. Champaign, IL*, Wolfram Media, 2002.

[37] Wolfram S., "Statistical Mechanics of Cellular Automata," *Reviews of Modern Physics*, vol. 55, no. 3, pp. 601-644, 1983.

Harinee Shanmuganathan is an aspiring scholar in the field of Computer Science and Engineering at Vellore Institute of Technology, Vellore. Her research interests include health care, cloud computing, information security, robotics and technology innovation. She received her Bachelor's degree in Computer Science and Engineering from Pondicherry University in 2016.

Anand Mahendran received his Ph.D (Computer Science and Engineering) degree from VIT University, India in the year 2012, M.E (Computer Science and Engineering) degree from Anna University, India in the year 2005 and B.E (Computer Science and Engineering) degree from VIT University, India in the year 2003. His research interests include formal language theory and automata, bio-inspired computing models. He has published more than 30 papers in international journals and refereed international conferences. He is currently working as an Associate Professor in School of Computer Science and Engineering in VIT Vellore, India.