I. J. Computer Network and Information Security, **2019, 2, 28-35**
Published Online February 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2019.02.04

An Efficient Image Block Encryption for Key Generation using Non-Uniform Cellular Automata

G. Kumaresan
Department of Computer Applications, National Institute of Technology, Tiruchirappalli - 620015, India
E-mail: kumareshtce@gmail.com

N. P. Gopalan
Department of Computer Applications, National Institute of Technology, Tiruchirappalli - 620015, India
E-mail: npgopalan@nitt.edu

T. Vetriselvi
Department of CSE, K. Ramakrishnan College of Technology, Tiruchirappalli-621112, India
E-mail: vetriselvi09@gmail.com

Received: 29 November 2018; Accepted: 14 December 2018; Published: 08 Febuary 2019

Abstract—Cryptographic image block encryption schemes play a significant role in information enabled services. This paper proposes an image block encryption scheme based on a novel three stage selection (TSS) method in a public cloud with reversible cellular automata. Due to the openness of public cloud, different attacks are possible over user sensitive information. The TSS method has three stages and they generate a robust master key with user plaintext as input and produces an encrypted block as key to be sent to authenticated users. An analysis of experimental results shows that this new method has a large key space and immune to brute force attacks, statistical cryptanalysis attacks and chosen plaintext attacks. Also, the encrypted image entropy value could be increased to 7.9988 making it ideal for a best image block encryption for key generation.

Index Terms—Cellular Automata, Cryptography, Security, Session Key Agreement, Reversible Rule.

I. INTRODUCTION

Image cryptography plays a momentous role in securing confidential images and are usually based on username and password. As user authentication techniques do not usually provide enough security for information enabled services, there is a need for an efficient method to protect user sensitive information, more so in a public cloud [22]. Hence, a novel TSS based image block encryption using reversible cellular automata in public cloud has been attempted in this paper. Two achievements are achieved in the proposed method that are (i) It is easy to implement in cloud storage applications and the keys strength is much better than others (ii) It provides better security with less computations.

A. Cloud Computing

Cloud computing is an internet based computing that virtualized, runtime accessible and the resources are delivered as a service across the web. It can be accessed anywhere, anytime and anyplace in the world over the internet. As stated in the NIST [13] description, “Cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction".

Cloud computing includes three typical services such as software as a service, infrastructure as a service and platform as a service and its demand can be executed as to four deployment models: private cloud, public cloud, hybrid cloud and community cloud as shown in Fig:1. Also, it includes scalability, pay-per-use, data storage, service level agreements [4, 12]. Cloud Service Provider (CSP) takes the responsibility to provides high security for customer’s secret information and compact with many services that can assist users to access their data anywhere, anytime and anyplace [17].

B. Celluar Automata

A cellular automata is a regular grid of cells which depends on neighborhood cells on its left, its right and the cell itself. It is reversible if it is global map is returnable [9]. In case the cellular automata is returnable which is settled in both ways is called as encryption. Hence, each rule considers as an inverse rule in the CA method. By executing CA rule in both directions such as forward and backward and then iterating for n times, the original configurations can be obtained which are explained in Table 1 and Table 2.

For example Rule 15 has an inverse to Rule 85. Cellular automata have become the significant tool in developing high speed encryption like mobile devices. One of the vital skin tones of using reversible cellular automata is the capability to create highly random generators that are appropriate for high speed applications [3] especially in the cloud environment.

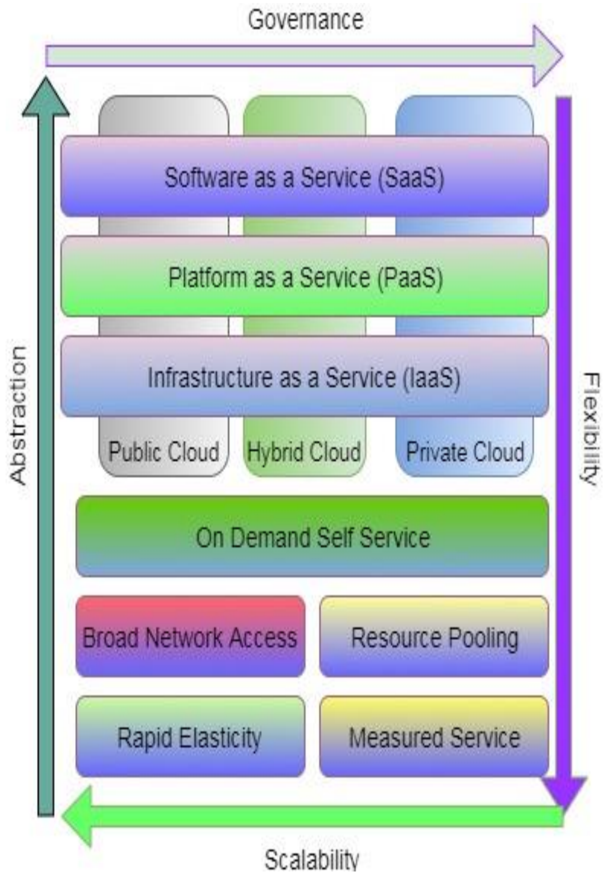


Fig.1. Cloud Computing

Table 1. Rule 15 has an inverse to Rule 85

R-15	111	110	101	100	011	010	001	000
	0	0	0	0	0	0	0	0
T	**1**	**0**	**1**	**1**	**1**	**0**	**0**	**0**
T+1	1	0	1	0	0	0	1	1
T+2	0	0	1	0	1	1	1	0

Table 2. Rule 85 has an inverse to Rule 15

R-85	111	110	101	100	011	010	001	000
	0	1	0	1	0	1	0	1
T	0	0	1	0	1	1	1	0
T+1	1	0	1	0	0	0	1	1
T+2	**1**	**0**	**1**	**1**	**1**	**0**	**0**	**0**

According to Kerckhoff's principle the secrecy of key only provides security, even though the cryptographic system is more complicated [8]. Many researchers have studied different methods and innovative approaches for key generation in public clouds [5, 16, 21]. Still, attackers prove too smart to overthrow all known security systems. The proposed image block encryption based TSS method using reversible cellular automata has been observed to provide better security and appear to be robust in cryptographic systems.

The rest of the paper is organized as follows: Section 2 introduces the literature review. In Section 3 we briefly illustrate the new TSS method, in Section 4 we provides the security analysis. In Section 5 we explain the experimental results, in Section 6 we conclude the paper.

II. Literature Review

In this part, we study the previous methods before presenting our proposed TSS scheme for image encryption. Recently many researchers have studied different cryptosystem approaches based on CA [1, 2]. Wolfram had presented first CA based secret key generation [20]. However, CA has the minimum number of reversible rules and inefficient to provide the long sequence of numbers [15]. To overcome these drawbacks, Sen *et al.* designed a stream cipher based random number generation based on CA [14]. Though, the system has increased the complexity.

To overcome this shortcoming Wang *et al.* introduced bit level image encryption depend on CA [18]. Also, it is vulnerable to chosen plaintext attack. Due to this disadvantages, many researchers focused on block cipher based cryptosystems in recent years. In general, the cryptographic method is used to generates random keys are remembered and the adversary can crack it. However, the proposed system to overcome these drawbacks by using a reversible CA mechanism based image key generation. Without knowing the corresponding CA transition rule, the adversary cannot obtain the original information from the proposed method as CA has undecidability in nature.

III. Proposed TSS Method

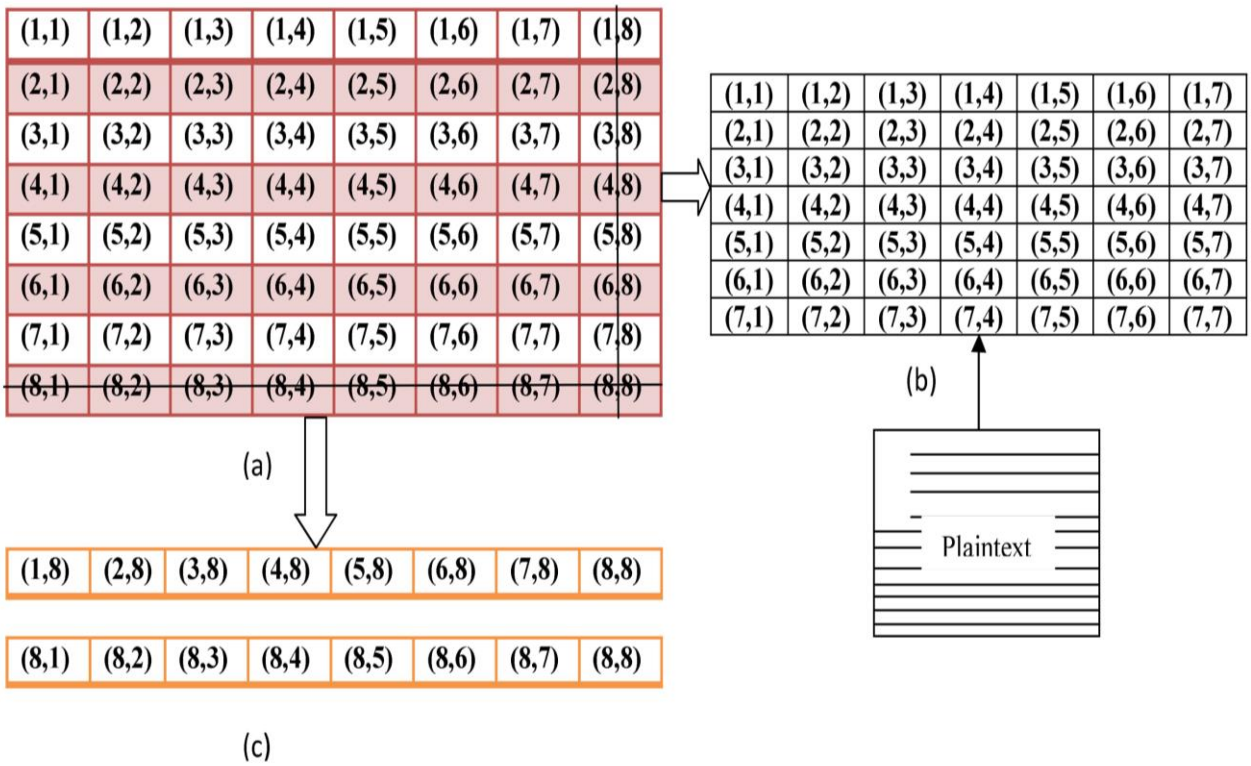
With an aim to strengthen user security in public cloud a novel method has been proposed. The original image (IM_n) is considered as $N \, X \, N$ matrix at the outset. In the first step, the content owner splits the image into two non-overlapping image blocks IM_{n1} and IM_{n2} of sizes $(N-1) \, X \, (N-1)$ as shown in Fig: 2(a)-(c). The diagonal pixels of IM_n are partitioned to form the 8-bit gray scale image block represented as IM_{n2} and the rest is considered as IM_{n1}.

A. Key Generation Schema

A good encryption key generation algorithm has to be sensitive to the initial values of the parameters concerned and the proposed method uses the logistic function to generate the initial values for specific threshold conditions. This function covers the mask of the image IMG_{n2} as described below:

$$L_{n+1} = r \times L_n (1 - L_n) \tag{1}$$

In (1), the radius r and the initial value L_n are supplied as input for logistic function which reflects a chaotic behavior when $0 < L_n < 1$ and $3.99 < r < 4$. After initiating the values for the key, the threshold function is used to store the binary values into IMG$_{n2}$ with the following conditions:



(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)	(2,8)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)	(3,8)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)	(4,8)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)	(5,8)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)	(6,8)
(7,1)	(7,2)	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)	(7,8)
~~(8,1)~~	~~(8,2)~~	~~(8,3)~~	~~(8,4)~~	~~(8,5)~~	~~(8,6)~~	~~(8,7)~~	~~(8,8)~~

(1,8)	(2,8)	(3,8)	(4,8)	(5,8)	(6,8)	(7,8)	(8,8)

(8,1)	(8,2)	(8,3)	(8,4)	(8,5)	(8,6)	(8,7)	(8,8)

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)
(7,1)	(7,2)	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)

Fig.2. Partition of image block (a) Original image IMG$_n$, (b) User's plaintext image IMG$_{n1}$, (c) Cellular automata rule image IMG$_{n2}$

$$f(L_n) = \begin{cases} 1, & L_n \geq 0.5 \\ 0, & L_n < 0.\,5 \end{cases} \tag{2}$$

Cellular automata (CA) is a regular grid of cells, which depends on its neighbors to the left, neighbors to the right and cells itself. The major drawbacks of CA in cryptography along with the minimum number of reversible rules and failure to produce the long sequence of numbers. To overcome these drawbacks, we use eight best CA rules (i.e., R153, R86, R101, R30, R105, R90, R165, R150) in the proposed TSS method which have the best results such as entropy [20].

First, we choose the eight best selected CA rules with 2^3 blocks in IMG$_{n2}$. Consequently, each block will contain different CA rules in IMG$_{n2}$. For example, let us take a block from IMG$_{n2}$. Suppose the block contains rule30, then the key length will be 128 bits. Then, before ciphering the text in IMG$_n$, we need to generate a master key in IMG$_{n2}$. Usually, the traditional techniques used for generating the master key suffer from various attacks such as chosen plain text attack, statistical cryptanalysis and brute force attacks. Therefore, the TSS method to generate the required master key using IMG$_{n2}$ which is robust to mitigate these attacks has been proposed in this paper.

The proposed model contains three stages: user (U$_i$), cloud service provider (CSP$_i$) and cloud service provider machine (CSPM$_i$). After initiating the values, a CA rule from IMG$_{n2}$ block will be selected using SL$_i$ = $|L_n\ X\ 7|$+1. The value of L_n can be obtained from logistic functions. In the first stage, U$_i$ needs to select a CA rule from IMG$_{n2}$ denoted as K$_{a1}$. In the second stage, CSPM$_i$ selects a CA rule K$_{a2}$ from IMG$_{n2}$. Both K$_{a1}$ and K$_{a2}$ are added. The resulting value K$_{a12}$ is temporarily stored in a memory buffer. In the third stage, CSP$_i$ selects another CA rule K$_{a4}$ from IMG$_{n2}$ block and it is added with previously result K$_{a12}$. Finally, it is the robust master key K$_{a124}$ in the form of binary values. The detailed process of the generation of master key is illustrated as shown in Fig:3.

B. *Block Encryption*

After generating K$_{a124}$, the IMG$_{n1}$ block is XORed with IMG$_{n2}$ block. Then, IMG$_{n1}$ contains U$_i$'s plaintext (PT$_i$) information in the form of binary values. The result will be produced at the end of the encrypted block IMG$_{n3}$.

Finally, the IMG$_{n3}$ will be sent to the registered U$_i$'s via smart phone. If U$_i$ enters the correct code, then they can access the cloud services, otherwise the request will be denied.

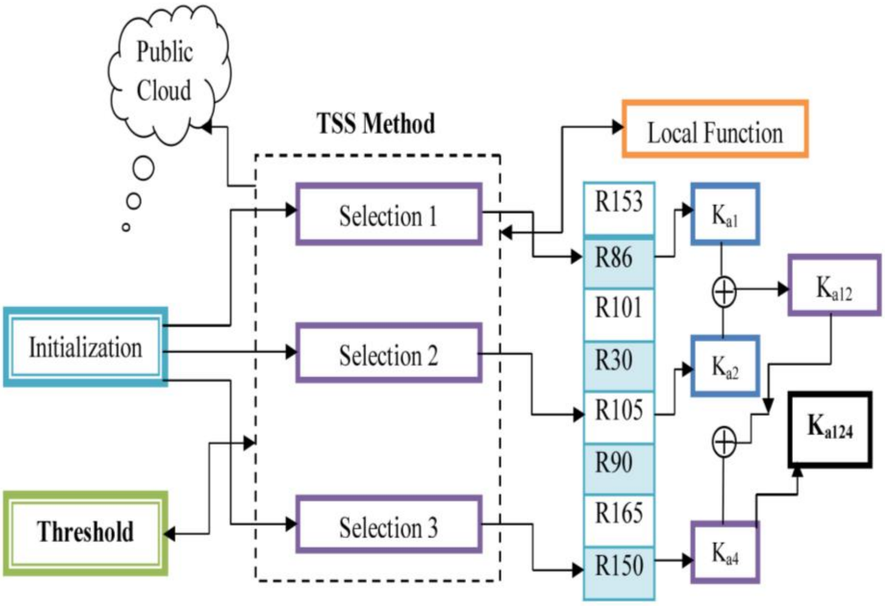


Fig.3. Detailed Process of Generating Image Key

IV. Security Analysis

Theorem 1: The proposed method is correct in verifying the correctness of computation results.

Proof: From section 3, let us consider the following logistic map function:

$$L_{n+1} = r \times L_n(1 - L_n)$$

with the interval from when $0 < L_n < 1$ and $3.99 < r < 4$. Where r represents radius. After satisfying interval, we can choose the binary values based on the following conditions:

For $f(L_n) \geq 0.5$, then stored 0 and
$f(L_n) < 0.5$, then stored 1 in the matrix.

That is,

$$\mathbf{K}_{N\times N} = \begin{bmatrix} 0.232 & 0.243 & 0.632 & \cdots & x_{1n} \\ 0.372 & 0.285 & 0.821 & \cdots & x_{2n} \\ 0.328 & 0.234 & 0.366 & \cdots & x_{3n} \\ \multicolumn{5}{c}{\dotfill} \\ x_{d1} & x_{d2} & x_{d3} & & x_{dn} \end{bmatrix}$$

Hence, each number is used to encrypt each pixel in the matrix. Therefore,

$$\mathbf{K}_{N\times N} = \begin{bmatrix} 0.232 & 0.243 & 0.632 & \cdots & x_{1n} \\ 0.372 & 0.285 & 0.821 & \cdots & x_{2n} \\ 0.328 & 0.234 & 0.366 & \cdots & x_{3n} \\ \multicolumn{5}{c}{\dotfill} \\ x_{d1} & x_{d2} & x_{d3} & & x_{dn} \end{bmatrix} \times 255$$

So, we have the following matrix

$$\mathbf{K}_{N\times N} = \begin{bmatrix} 59 & 62 & 161 & \cdots & x_{1n} \\ 95 & 73 & 209 & \cdots & x_{2n} \\ 84 & 60 & 93 & \cdots & x_{3n} \\ . & & & & \\ \multicolumn{5}{c}{\dotfill} \\ x_{d1} & x_{d2} & x_{d3} & \cdots & x_{dn} \end{bmatrix}$$

Then, the obtained matrix converted into 8-bit binary number and it becomes:

$$\mathbf{K}_{N\times N} = \begin{bmatrix} \mathbf{00111011} & 00111110 & 10100001 & \cdots & x_{1n} \\ 01011111 & 01001001 & 11010001 & \cdots & x_{2n} \\ 01010100 & 00111100 & 01011101 & \cdots & x_{3n} \\ . & & & & \\ \multicolumn{5}{c}{\dotfill} \\ x_{d1} & x_{d2} & x_{d3} & \cdots & x_{dn} \end{bmatrix}$$

Hence, the proposed method takes first 8-bit number of $\mathbf{K}_{NXN}$ as input of the cellular automata selected rules from SL_i. Now assuming,

$$\mathbf{CA} = 0\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 1$$

In case the first stage, the $\mathrm{U_i}$ selects Rule30 from $\mathrm{IMG_{n2}}$, we obtained:

Rule30 binary expression: $\mathrm{U_i^{t+1} = U_{i\text{-}1}^t xor\left[U_i^t or U_{i+1}^t\right]}$

$$0\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 1$$
$$\text{CA'= 11101011} \tag{3}$$

where CA' represents first stage first iteration state result.

$$1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 1$$

$$CA'' = 00100010 \tag{4}$$

where CA" represents first stage second iteration state result. Then, equation (3) added with equation (4) and it becomes:

$$11101011 + 00100010 = 100001101$$

Therefore, the first stage result is: **100001101**

Similarly in the second stage, the $CSPM_i$ selects Rule153 from IMG_{n2} and it becomes:

Rule153 binary expression: $U_i^{t+1} = U_i^t \text{xnor}\left[U_{i+1}^t\right]$

1 0 0 0 0 1 1 0 1

$$CA' = 01110100 \tag{5}$$

where CA' represents second stage first iteration state result.

0 1 1 1 0 1 0 0

$$CA'' = 01100011 \tag{6}$$

where CA" represents second stage second iteration state result. Then, equation (5) added with equation (6) and it becomes:

$$01110100 + 01100011 = 11010111$$

Therefore, the second stage result is: **11010111**

Similarly in the third stage, the CSP_i selects Rule165 from IMG_{n2} and it becomes:

Rule165 binary expression: $U_i^{t+1} = U_{i-1}^t \text{xnor}\left[U_{i+1}^t\right]$

1 1 0 1 0 1 1 1

$$CA' = 10111011 \tag{7}$$

where CA' represents third stage first iteration state result.

1 0 1 1 1 0 1 1

$$CA'' = 01010101 \tag{8}$$

where CA" represents third stage second iteration state result. Then, equation (7) added with equation (8) and it becomes:

$$10111011 + 01010101 = 100010000 \tag{9}$$

Therefore, the third stage result is: **100010000**

Therefore, the image key values is: 272

Similarly, to perform a decryption process. Hence, the proposed method uses the same rules (i.e., Rule30, Rule153, Rule165) while decrypting the final image key. In the end, it obtained the initial key 59 (i.e., 00111011). The system remembers all the selected rules while encrypting the initial key image.

It is proved that the proposed method computation result is correct in the system.

Theorem 2: The proposed method supports the robustness of the key image against cipher plaintext attack.

Proof: According to the recent proposal on IMG_n the block enciphering schema, the master key should meet all the basic security requirements such as avalanche criteria (AVL_c), confusion (CFS_p) and diffusion (DFS_p) properties. The criteria AVL_c is evident when an input changes slightly and the output changes dramatically. Such a small change in the K_{a124} and PT_i which has to cause a significant change in CT_i.

The TSS method meets these criteria by using a logistic function L_{n+1} from (1) to generate the initial parameter values for a key generation. CFS_p is about the relationship between CT_i and K_{a124} which has to be as much complex as possible and this is achieved through CFS_p with three stages and the selected key values K_{a1}, K_{a2} and K_{a4} from IMG_{n2} block and thus, the computations leads to generate a enough complexity.

Hence, it provides better K_{a124}. DFS_p is about the relationship between PT_i and CT_i which has to be as much complex as possible and this is achieved through DFS_p with both L_{n+1} and K_{a124} techniques. Therefore, the TSS method meets all the basic security requirements of image block encryption and also prevents the chosen plaintext attack in public cloud.

A. *Key Space Analysis*

An encryption scheme has to be sensitive to all the encryption keys. Based on U_i's requirements, a key K_{a124} of 128-bit length (with a possibility of choosing 2^{128} keys) is generated. Here, L_{n+1} from (1) is used as the key initiator and the images IMG_{n3} of sizes $N X N$ used for encryption that results in 2^{8XNXN} choices. These are clearly sufficient to prevent all possible attacks.

B. *Efficiency*

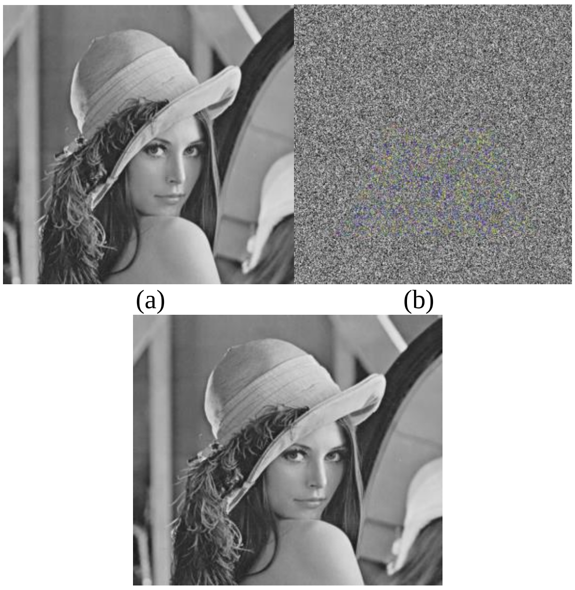
From Theorem 1, the proposed method executed with a low computational cost. It is noticed that many functions are assigned to the cloud environment. Hence, the U_i's needs to generate keys after the data stored in the cloud. Also, the system input is independent of the proposed system efficiency.

V. Experimental Results

In our experiment, the different statistical tests and measurements are implemented with three gray scale (512 X 512) images such as Lena, Baboon and Peppers from [6]. Fig: 4(a) shows the original lena gray scale key image. Fig: 4(b) shows the encrypted lena key image using proposed TSS encryption algorithm. Also, Fig: 4(c) shows the decrypted lena key image which is similar to the Fig: 4(a). Our method meets the requirements on the histogram (HGM) and data entropy (ENV_n). HGM has to be fall on a uniform distribution of the pixel values. Figure 5 shows that the HGM analysis of the IMG_{n3} images is uniform way. Therefore, without the knowledge of K_{a124} values, there is no possibility of statistical attacks and the adversary cannot steal any information about the

PT_i block of IMG_{n1}.

Also, the ENV_n is concerned with randomness test. High ENV_n indicate a high degree of randomness in values for any message that is coded into n bits, as the gray level images are coded into 8 bits and the optimal ENV_n is 8. The ENV_n is calculated using (10).



(a) (b)

(c)

Fig.4. Test image: Lena (a) The original key image (b) The encrypted key image (c) The decrypted key image

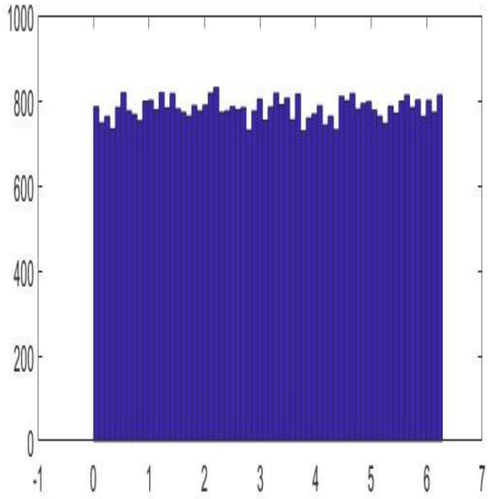


Fig.5. Lena Histogram Key Image

$$H(m) = -\sum_{c=0}^{2^N-1} P_c \log_2(P_c) \qquad (10)$$

where P_c represents different gray scale image values from 0 to 255.

Hence, Table 3 interprets the different ENV_n which are obtained from IMG_n and IMG_{n3}. The results are compared with existing two block based methods proposed in [7, 19]. It has been found that the IMG_{n3} has achieves near optimal ENV_n which is sufficient to prevent statistical cryptanalysis attack. Also, the adversary cannot extract the required information from IMG_{n1} without the knowledge of K_{a124} value.

Table 3. Entropy Value of Plain and Proposed Images for TSS and Existing Methods

Images	Plain	Proposed	[19]	[7]
Lena	7.3401	**7.9988**	7.9926	7.9986
Baboon	7.1125	7.9975	7.9934	7.9920
Peppers	7.0725	7.9988	7.9923	7.9987

Table 4. Illustration of Key Generation Time (sec)

Entry	[11]	[10]	Proposed
19	13.9	15.1	5.5
14	9.0	10.5	4.1
10	8.7	9.9	4.0
6	3.7	5.8	2.4
2	2.6	4.4	1.0

The key generation time has been examined based on number records in the CSP database. It has been found that the proposed method has better results in comparison with existing methods in [10, 11]. Also, in case the number of entries increased the key generation time is decreased this is shown in Table 4 and graphically shown in Fig: 6.

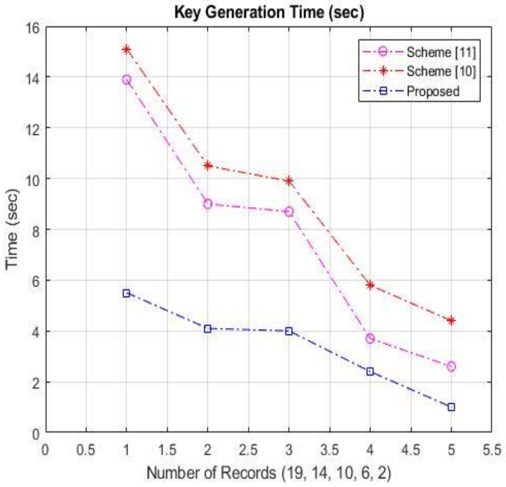


Fig.6. Analysis of Key Generation Time

VI. Conclusions

A novel image block encryption for key generation using TSS method with reversible cellular automata to secure highly sensitive data in the public cloud has been proposed and analyzed in this paper. The encrypted key prevents the statistical cryptanalysis attacks and brute force attack and as it also satisfies the avalanche criterion, confusion and diffusion properties, chosen plain text attack is prevented. Finally, the results obtained show that the proposed method provides better performance and an enhanced ciphered image entropy value of 7.9988 which an ideal one for any image block encryption method for key generation.

ACKNOWLEDGMENT

This work has been supported by the University Grants Commission in India under rajiv gandhi national fellowship (Reference number: F1-17.1/201516/RGNF-2015-17-SC-TAM-4667).

REFERENCES

[1] A. A. Abdo, L. Shiguo, I. A. Ismail, M. Amin and H. Diab. A cryptosystem based on elementary cellular automata. *Commun Nonlinear Sci Number Simulat,* 18:136-147, 2013.

[2] Y. N. Abolfazl, H. M. Mohammed, and N. T. Masood. Color image encryption based on hybrid hyper chaotic system and cellular automata. *Optics and Lasers in Engineering,* 90:225-237, 2017.

[3] P. Anghelascu. Encryption algorithm using programmable cellular automata. *IEEE World Congress in Internet Security,* pages 233-239, 2011.

[4] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, and I. Brandic. Cloud computing and emerging IT platforms vision hype and reality for delivering computing as the 5th utility. *Future generation computer systems,* 25:599-616, 2009.

[5] G. Cheng, L. Ningqi, A. B. Zakrirul, J. Yingmo, C. Yuvanfong, F. Ben, and A. Muhammad. Key aggregate authentication cryptosystem for data sharing in dynamic cloud storage. *Future Generation Computer Systems,* 84:190-199, 2018.

[6] Img512. Miscellaneous gray level images accessed november 2018. *Accessible at http://www.decsai.ugr.es/cvg/dbimagenes/g512.php.,* 2018.

[7] C. Junxin, Z. Yu, Q. Lin, F. Chong, and X. Lisheng. Exploiting chaos based compressed sensing and cryptographic algorithm for image encryption and compression. *Optics Lasers Technology,* 99:238-248, 2018.

[8] A. Kerckhoff's. La cryptographie militaire. *Journal des sciences militaires*, 9:161-191, 1883.

[9] G. Kumaresan, N. P. Gopalan. An analytical study of cellular automata and its applications in cryptography. *International journal of computer network and information security*, 9:45-54, 2017.

[10] G. Kumaresan, N. P. Gopalan. EduCloud: A dynamic three stage authentication framework to enhance security in public cloud. *International journal of engineering and manufacturing,* 6:12-26, 2017.

[11] G. Kumaresan, N. Veeraragavan and L. Arockiam. A dynamic two stage authentication framework to enhance security in public educloud. *International journal of applied engineering research*, 10:126-131, 2015.

[12] B. Lee, B. Robert, C. Shilong, H. Mike, L. Fang, K. Viktor, and Jian. Useful information for cloud adopters. *NIST cloud computing program*, 2:1-73, 2011.

[13] P. Mell T. Grance. The NIST definition of cloud computing version 15 technical report. *Computer and information Sciences*, 53:1-10, 2009.

[14] S. Sen, C. Shaw, D. R. Chowdhuri, N. Ganguly, and P. Chaudhuri. Cellular automata based cryptosystem. *ACRI, Geneva, Switzerland,* pages 303-314, 2002.

[15] A. Seredynski, Bouvry, P. Albert, and Y. Zomaya. Cellular automata computations and secret key cryptography. *Parallel Computing,* 30:753-766, 2004.

[16] N. K. Sreelaja and G. A. Vijayalakshmi. Swarm intelligence based key generation for stream cipher. *Security and Communication Network,* 4:181-194, 2011.

[17] S. Subashini, V. Kavitha. A survery on security issues in service delivery models of cloud computing. *Journal of network and computer applications,* pages 1-11, 2011.

[18] X. Wang and D. Luan. A novel image encryption algorithm using chaos and reversible cellular automata. *Commun Nonlinear Sci Number Simulat,* 18:3075-3085, 2013.

[19] X. Wang, Y. Zhao, H. Zhang, and K. Guo. A novel color image encryption scheme using alternate chaotic mapping structure. *Optics Lasers and Engineering,* 82:79-86, 2016.

[20] S. Wolfram. Random sequence generation by cellular automata. *Advanced Applied Mathematics,* 7:123-169, 1986.

[21] Z. Yunpeng, L. Xin, and S. Manhui. Dna based random key generation and management for otp encryption. *Biosystems,* 159:51-63, 2017.

[22] W. Zhendong, T. Longwei, L. Ping, W. Ting, J. Ming, and W. Chunming. Generating stable biometric keys for flexible cloud computing authentication using finger vein. *Information Sciences,* 433:431-447, 2018.

Authors' Profiles

G. Kumaresan: Research Scholar at Department of Computer Applications, National Institute of Technology Tiruchirappalli. He received MCA from Thiagarajar College of Engineering, Madurai, India. M.Tech from Bharathidasan University, Tiruchirappalli, India and M.Phil. from St.Joseph College, Tiruchirappalli, India. He has four years experience in teaching as an Assistant Professor. His areas of interest include Cellular Automata based Cryptography and Cloud Security.

N. P. Gopalan: is Professor of Computer Applications Department at National Institute of Technology, Tiruchirappalli, Tamil Nadu, and India. He obtained his PhD from the Indian Institute of Science, Bangalore. His research interests lie in Data Mining, Cryptography, Distributed Computing and Theoretical Computer Science.

T. Vetriselvi: Part-Time Research Scholar at Department of Computer Applications, National Institute of Technology Tiruchirappalli. She is currently an assistant professor at the K. Ramakrishnan College of Technology, Tiruchirappalli. She has 11 years of teaching experience. Her areas of interest include Data Mining and Programming Languages.

How to cite this paper: G. Kumaresan, N. P. Gopalan, T. Vetriselvi,"An Efficient Image Block Encryption for Key Generation using Non-Uniform Cellular Automata", International Journal of Computer Network and Information Security(IJCNIS), Vol.11, No.2, pp.28-35, 2019.DOI: 10.5815/ijcnis.2019.02.04