INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 5)
Available online at: www.ijariit.com

Comparison between image encryption algorithms for wireless sensor

Kakshak Porwal
kakshak@gmail.com
Vellore Institute of Technology, Vellore, Tamil Nadu

ABSTRACT

Devices used in the Wireless Sensor Network are very small and compact in size, and this technology is expected to connect billions of devices in the future. Wireless Sensor Network used in IoT is expected to generate heaps of data and security can be a great threat. Due to its compactness, resource constraints and low powered it is difficult to use conventional encryption algorithms because of expensive computation, complexity and power used in encryption and it also requires many rounds to encrypt, leads to wasting of constrained energy of the WSN devices. In this paper, a comparison is proposed between two lightweight encryption algorithms based on Feistel structure named Secure Force (SF) and Secure Internet of Things (SIT) implemented by using a simple architecture that only consists of simple mathematical operation (AND, OR, XOR, XNOR, Shifting, Swapping). This can reduce the load on encoder because complex key expansion process is only carried at the decoder. Confusion and diffusion of the image matrix, and different substitution and permutation techniques are also included to increase the complexity of the cipher. The proposed comparison helps us to evaluate which algorithm is better in terms of architecture, time complexity, flexibility, and security.

Keywords— *IoT (Internet of Things), WSN (Wireless Sensor Networks), SIT (Secure IoT), SF (Secure Force), Feistel Networks*

1. INTRODUCTION

Wireless Sensor Network is gaining popularity because of its low-power, low cost and compact in size and can communicate wirelessly. Security plays an important role as WSNs are deployed in large quantities [1]. As the broadband Internet is now generally accessible and its cost of connectivity is reduced, more gadgets and sensors are getting connected to it. The data is shared in large amount through this device so communication with the IOT must be secure. The nodes of the sensors are vulnerable to attacks, such as tampering, eavesdropping, interception and modification of data. Hence there is a need of strong encryption technique in order to secure data transmission [2, 3]. The IOT is taking the conventional internet, sensor network and mobile network to another level as everything is connected to internet. The matter of concern is that it must be kept confidential, data integrity must be preserved and authenticity that will emerge on account of security and privacy. Conventional cryptographic algorithms are not suitable for encryption because of high power and complexity [4, 5]. The issue in designing an algorithm is security, memory, power, cost and performance. In general, cryptographic algorithm can be designed into two categories symmetric and asymmetric key algorithm. Asymmetric algorithm requires large amount of processing power, memory and bandwidth which limits its implementation for WSN. So, there is a need symmetric algorithm with low power consumption such as Secure Force and Secure Internet of Things [6].

2. SECURE FORCE ALGORITHM

The Secure Force algorithm is a low-complex algorithm implemented in WSN. To increase the energy efficiency, the algorithm consists of five encryption rounds. Less the number of encryptions round less is the power consumption. To enhance the security encryption round consist of six simple mathematical operations on only 4-bit data (it is also compatible with 8-bit computing devices for WSNs). This type of strategy is followed to create enough confusion and diffusion of data to withstand different types of attacks [1].

Figure 1 shows the key expansion process, it includes complex mathematical operations (multiplication, permutation, transposition, substitution and rotation) to generate different keys for various encryption rounds. The computational burden is shifted to decoder this will help to increase the lifespan of sensor nodes. The generated keys are transmitted to encoder securely for encryption process. It is energy efficient, robust and secure designed for WSNs. Overall Secure force comprises of different block: A) Key Expansion Block B) Key Management Protocol C) Encryption Block D) Decryption Block [1]. The following important notations are below shown in table:

Table 1: Notation used in the proposed algorithm

Notation	Function	Notation	Function
∧	AND	Θ	XNOR
∨	OR	<<<	Left Shift
⊕	XOR	X	Multiplication

2.1 Key Expansion

Key expansion is the main process that is used to generate different keys for different rounds of encryption and decryption used for securing images. Different operations are performed to create confusion and diffusion, this is done to reduce weak keys as well as to make the algorithm more efficient. The five different round keys are derived from the input cipher key. The method of key generation has two different parts: key expansion and round key schedule. The process comprises of two components: key expansion and round key selection. The key expansion consist of logical operation which includes logical operations (XOR, XNOR), Left Shifting (LS), matrix multiplication using Fix Matrix (FM), permutation using P-table and transposition using T-table [7].

The cipher key (K) is a linear array of 64 bit, which divided consist of 4 half's of 16 bits. Each matrix of 16 bit is arranged in to a 4*4 matrix row-wise on which Left Shift (LS) operation is applied. The result is arranged in 4*4 matrix column wise and logical operations (XOR, XNOR) are then performed. The resultants of these operations are combined to form 64 bit linear array. The obtained bits are passed to P-table and are arranged in 4*4 matrix row-wise on which Left Shift (LS) operation is performed. After the left shift the resultant matrix is multiplied with a Fix Matrix (FM) that transforms the 16 bit data into 64 bits and left shift is performed. The shifted bits are divided into four column wise 16 bits blocks on which AND & XOR operations are performed to transform to a single 16 bit block. These keys are used by substitution and transposition techniques on the 16 bit blocks to produce 4 sub keys (K1, K2, K3, K4) of 16 bits each, which are used for the first four rounds of the encryption for the fifth round the key is generated with XOR operations with the four keys [8].

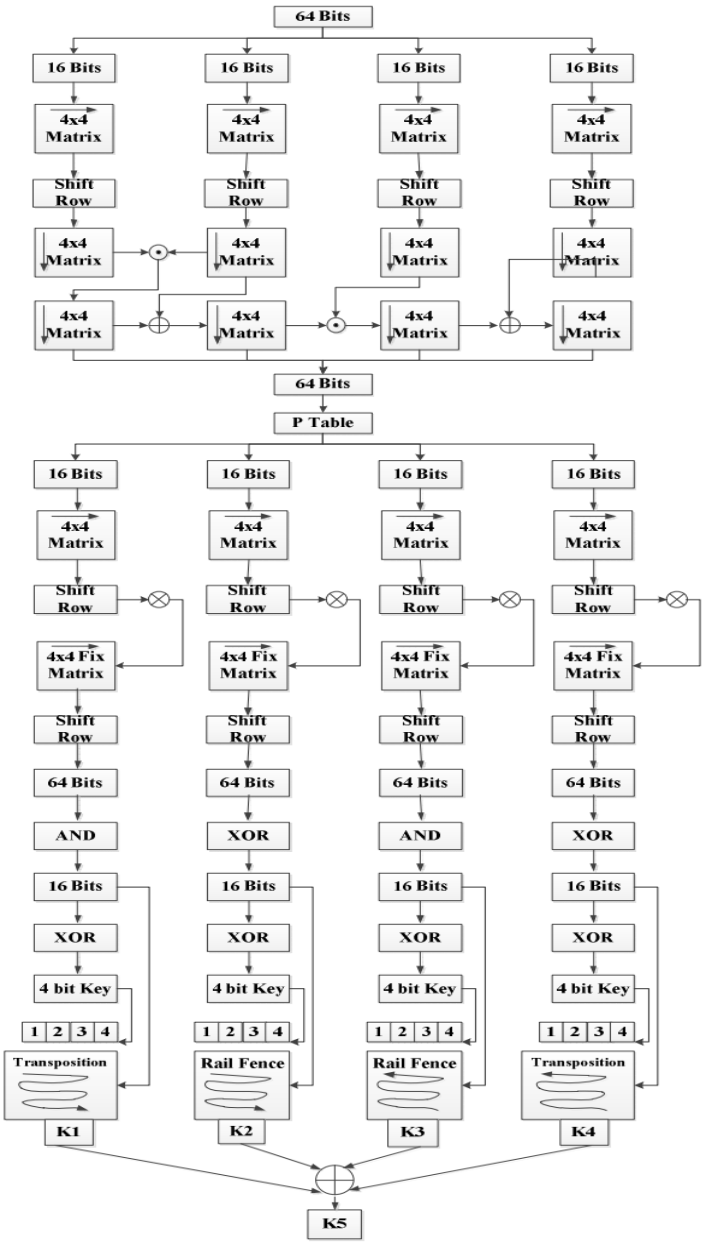

Fig. 1: Key Expansion

2.2 Encryption

The encryption process is described in figure 2 after the key is generated using key expansion block is securely received by the encoder through the secure communication channel created using LEAP protocol. The encryption process simply involves operation like AND, OR, XOR, XNOR, left shift (LS), substitution (S boxes) and swapping operations are performed.to create diffusion and confusion. The plain text is of 64 bit which is divided into two half's each of 32 bits and each 32 bits are further divided into two halves of 16 bits each. In each round swapping of16 bit blocks are performed [9]. The major purpose of this is to change the original positions of data to get more complex cipher. Sub keys generated using key expansion is XNOR with the left and right half of each round respectively. The output of each round is input for the next round as well it is mapped with F-function. F-function box involves substitution (S boxes), AND, OR and left shift operation. The output from the F function is then XOR with the swapped 16 bits of the same round resulting in confusion of data. This brings the end to the encryption process. The decryption process is just the reserved of the procedure described above [10].

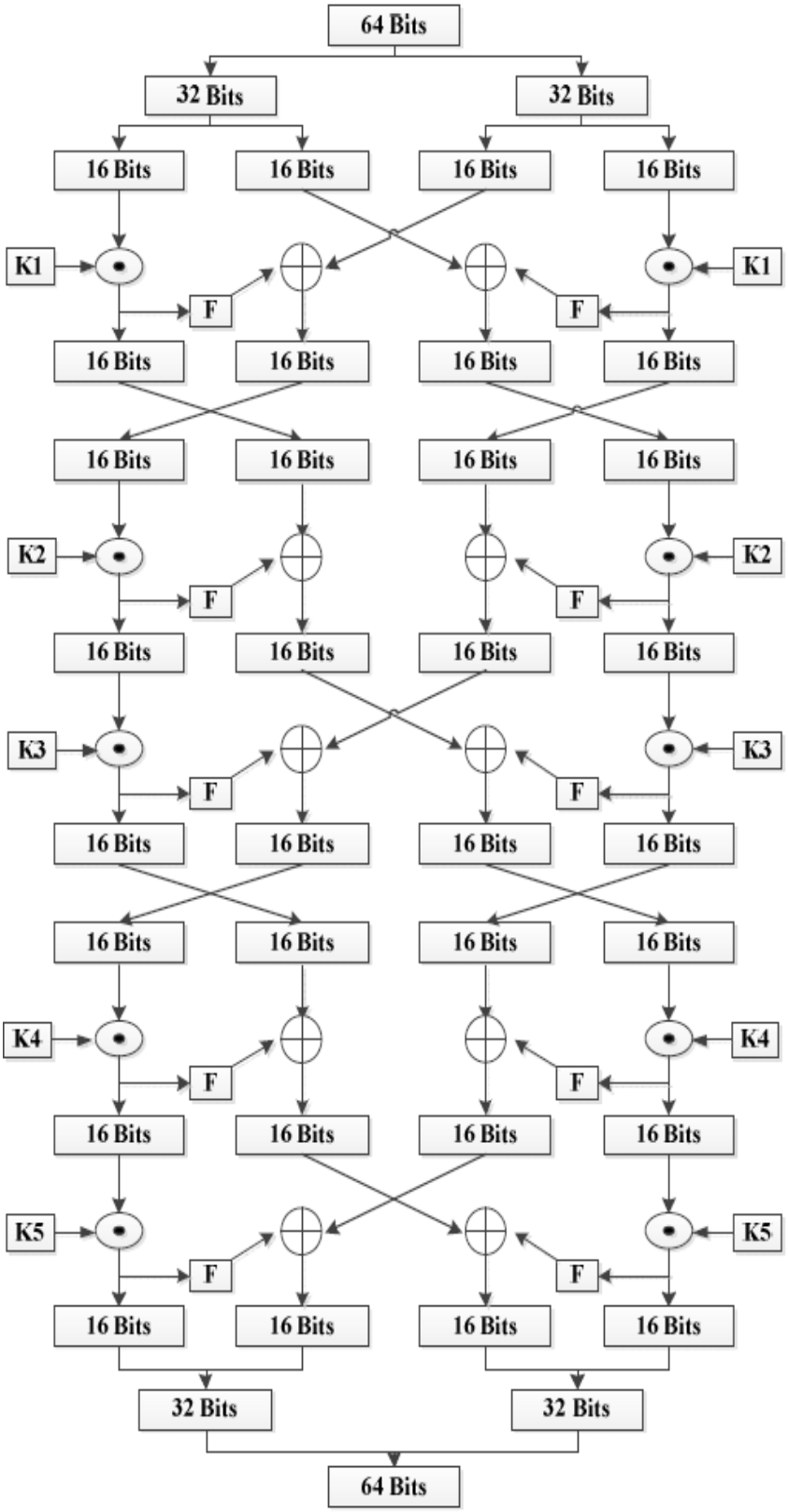

Fig. 2: Block Diagram of Encryption process

2.3 Round Transformation

The round transformation of Secure Force algorithm consists of different operation that includes F-function, XOR, XNOR and swapping.

- **Swapping Operation:** The left 16 bits of input is swapped with right 16 bits and right is swapped with left 16 bits. The aim is to change the original position of data to get more complex cipher.
- **F-function:** It is one of the most important part of the encryption algorithm that induces diffusion of data. The figure for the f-function is displayed below:

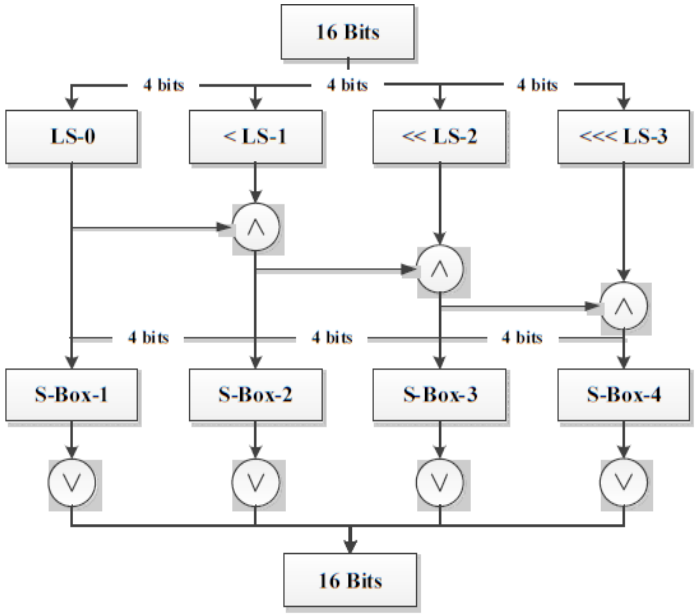

Fig. 3: F-function

- **Left-Shift:** In left shift operation 16 bits of data is divided in 4*4 block and left shifting is performed on each block. The operation results in complete mixing of data.
- **Substitution (S Boxes):** F-function 4 different types of S boxes constructed by using different operations to mix data and make result more complex. S boxes results are generated in such a way that middle part of each 4-bit data is considered as the column and the corner bits are considered as rows.

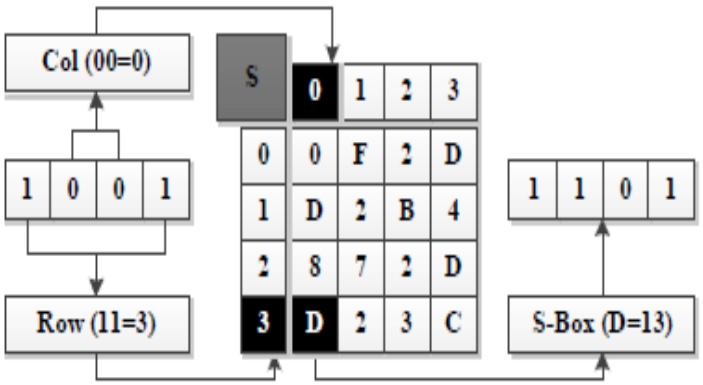

Fig. 4: S-box

3. SECURE INTERNET OF THINGS

The architecture of the SIT provides a simple architecture suitable for implementing in IoT environment. It is based on fiestel architecture and SP networks; the major advantage of this architecture is that encryption and decryption operations are almost same. It is a light weight algorithm and computational complexity is at moderate level [11]. It consists of 64-bit key and plain text. It consists of encryption rounds to create confusion and diffusion. More the number of rounds better is the security. The cryptographic algorithms are usually designed to take an average 10 to 20 rounds to keep the encryption process strong but due to resource constraints it is restricted to five rounds, each encryption round include mathematical operation that operate on 4 bits of data, to create sufficient confusion and diffusion of data [12]. Another vital process in symmetric algorithm is key generation, which includes complex operations as discussed below.

3.1 Key Expansion

The most fundamental component in the process of encryption and decryption is the key generation. The key determines the security of the whole algorithm. Therefore, necessary measures must be taken be in order to increase strength of key. Encryption consist of five different rounds so five keys is required. To do so key expansion is shown below.

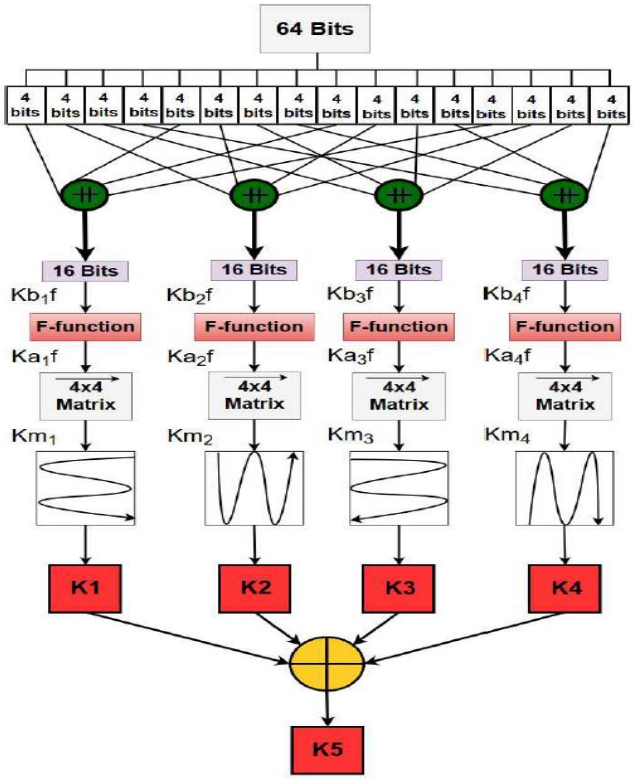

Fig. 5: Key Expansion for SIT

In the first step 64 bit key is divided into 4 segments [13, 14].

- **F-function:** F-function operates on 16-bits data. Therefore four f-function blocks are used.

$$Kb_if=||^4_{j=1}Kc_{4(j-1)+1} \quad (1)$$

- The next step is to get values of Ka_if by passing 16 bits of Kb_if to the f function.

$$Ka_if=f(Kb_if) \quad (2)$$

F-function comprises of P and Q table. These tables perform linear and non-linear resulting to confusion and diffusion of data [15, 16].

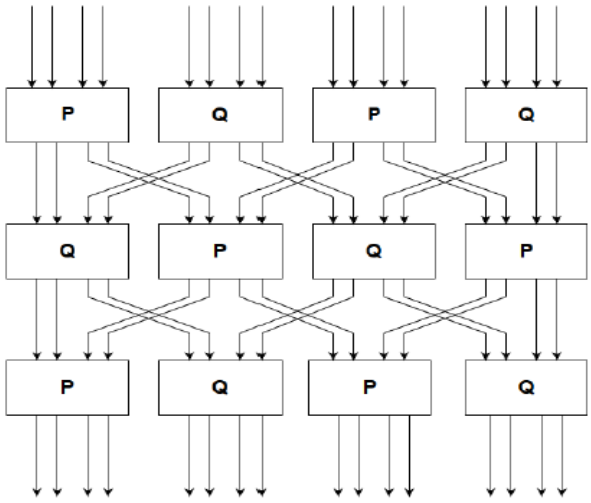

Fig. 6: F-function

Table 2: P-Table

Kci	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
P(Kci)	3	F	5	6	A	2	3	C	F	0	4	D	7	B	1	8

Table 3: Q-Table

Kci	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Q(Kci)	9	E	5	6	A	2	3	C	F	0	4	D	7	B	1	8

- To obtain different keys for encryption rounds K1, K2, K3, K4 the matrices are transformed into arrays of 16 bits that is called round keys.

$$K1=a_4 ++ a_3++ a_2++ a_1++ a_5++ a_6++ a_7++ a_8++ a_{12}++ a_{11}++ a_{10}++ a_9++ a_{13}++ a_{14}++ a_{15}++ a_{16} \quad (3)$$

$$K2=b_1 ++ b_5++ b_9++ b_{13}++ b_{14}++ b_{10}++ b_6++ b_2++ b_3++ b_7++ b_{11}++ b_{15}++ b_{16}++ b_{12}++ b_8++ b_4 \quad (4)$$

$$K3=c_4 ++ c_3++ c_2++ c_1++ c_5++ c_6++ c_7++ c_8++ c_{12}++ c_{11}++ c_{10}++ c_9++ c_{13}++ c_{14}++ c_{15}++ c_{16} \quad (5)$$

$$K4=d_{13} ++ d_9++ d_5++ d_1++ d_2++ d_6++ d_{10}++ d_{14}++ d_{15}++ d_{11}++ d_7++ d_3++ d_4++ d_8++ d_{12}++ d_{16} \quad (6)$$

3.1 Encryption

The process of encryption consists of five rounds to create enough confusion and diffusion of data. It consist of logical operation, left shift, swapping and substitution [2, 17, 18]. The figure is shown below.

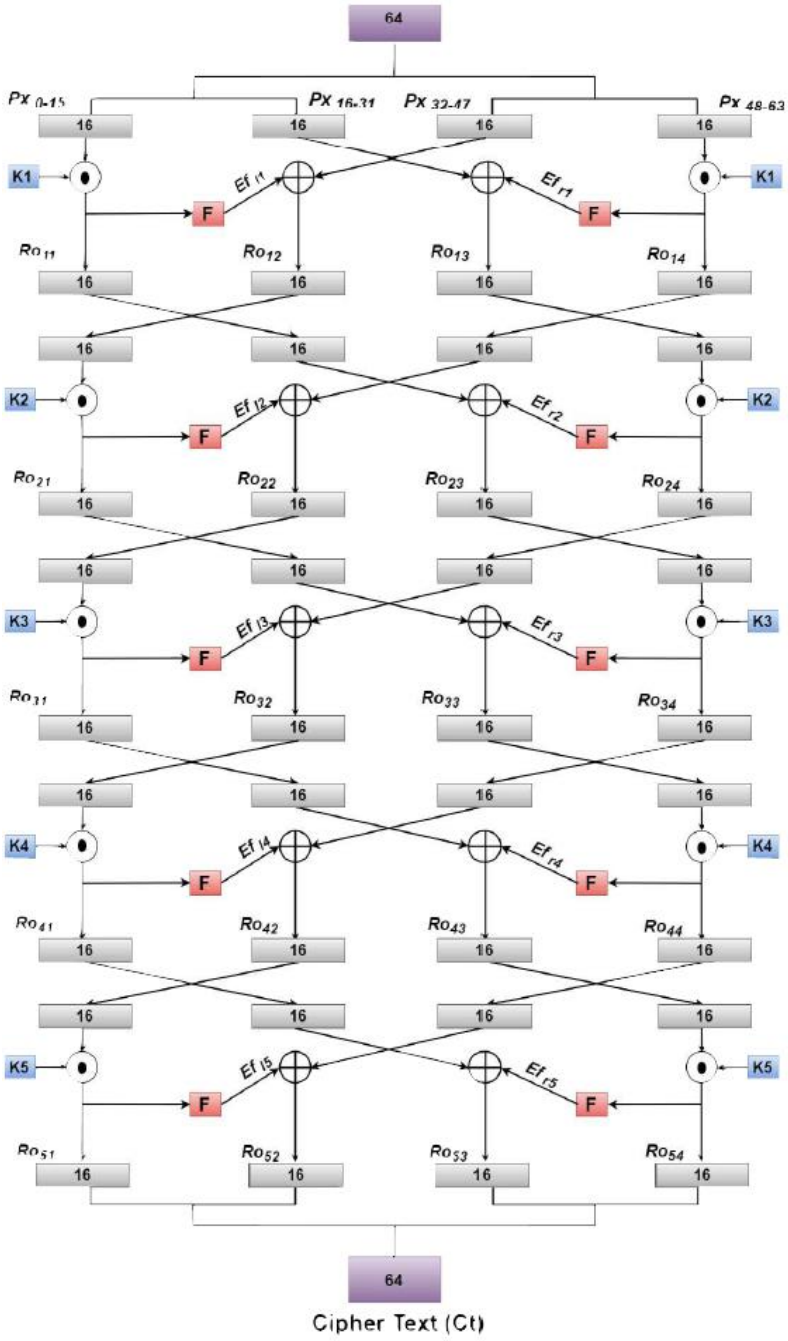

Fig. 7: Encryption for SIT

4. COMPARATIVE ANALYSIS FOR SF AND SIT

4.1 Evaluation parameters

To test and analyse the strength between the two proposed algorithms the following parameters are needed. Comparisons must be done on the basis of security and efficiency. The evaluation of the image encryption algorithms is carried out on certain well know parameters used by various authors in order to assess the performance.

- **Execution Time:** The execution time is one of the essential parameters that needs to consider along with security in the development of encryption algorithm defined as total time required for the encoding/decoding of a particular data.
- **Image Entropy:** In information theory, entropy is a measure of the uncertainty associated with a random variable which quantifies the expected value of the information contained in a message. In statistical analysis, an image with higher entropy value is considered to be more secure than an image with lesser entropy value and hence is used as an evaluating parameter while checking the robustness and security of an algorithm against statistical crypt-attack. Image entropy is a quantity which is used to describe the `business' of an image, i.e. the amount of information which must be coded for by a compression algorithm. An image that is perfectly flat will have an entropy of zero. Consequently, they can be compressed to a relatively small size. On the other hand, high entropy images such as an image of heavily cratered areas on the moon have a great deal of contrast from one pixel to the next and consequently cannot be compressed as much as low entropy images. Image entropy as used in my compression tests is calculated with the formula:

$$\text{Entropy}= -\sum_{i=1}^{2^8} P(Ii)\log_b P(Ii) \tag{7}$$

The more the value of entropy of the image, the better it is encrypted.

- **Correlation:** All correlations have two properties: strength and direction. The strength of a correlation is determined by its numerical value. The direction of the correlation is determined by whether the correlation is positive or negative. Positive correlation: Both variables move in the same direction. In other words, as one variable increases, the other variable also increases. As one variable decreases, the other variable also decreases. Negative correlation: The variables move in opposite directions. As one variable increases, the other variable decreases. As one variable decreases, the other variable increases. The graph in the experiment is positive correlation.
- **Histogram Analysis:** A graphical representation, similar to a bar figure in structure that organizes a group of data points into user-specified ranges, a histogram is the most commonly used graph to show frequency distributions. The histogram of the original image taken is usually unique with peaks and troughs corresponding to the pixel value at each point of the image while the histogram of an encrypted image is more or less level showing that the pixels are uniformly scattered.
- **NPCR and UACI:** The number of changing pixel rate (NPCR) and the unified averaged changed intensity (UACI) are two most common quantities used to evaluate the strength of image encryption algorithms/ciphers with respect to differential attacks. Conventionally, a high NPCR/UACI score is usually interpreted as a high resistance to differential attacks. The NCPR computes the ratio of different pixels between the plaintext and the cipher text images as follows.

$$\text{NPCR}=\frac{\sum_{i,j} D(i,j)}{M\times N}\times 100\% \tag{8}$$

Where

$$\text{D(i,j)}=\begin{cases}0 & if\ I(i,j)=K(i,j)\\ 1 & otherwise\end{cases}$$

UACI computes the number of averaged changed intensity between cipher text images. UACI can be calculated as:

$$\text{UACI}=\frac{1}{M\times N}\left[\sum_{i,j}\left[\frac{I(i,j)-K(i,j)}{255}\right]\right]\times 100\% \tag{9}$$

4.1 Results

Simulation of both the algorithm is done to perform the standard tests image entropy, histogram analysis and execution time of images for different keys is calculated. For applying encryption algorithm the 64 bit key used K1-AAAAAAAAAAAAAAAA for the SIT encryption algorithm on lena, baboon, panda image and encrypted image is shown in figure 8, 10, 12. For the secure force algorithm with the same key is shown in figure 9, 11, 13 implemented in MATLAB.

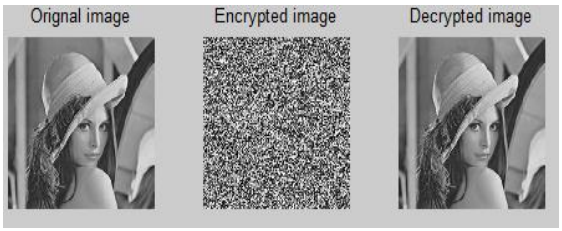

Fig. 8: Encryption and decryption with SIT (lena)

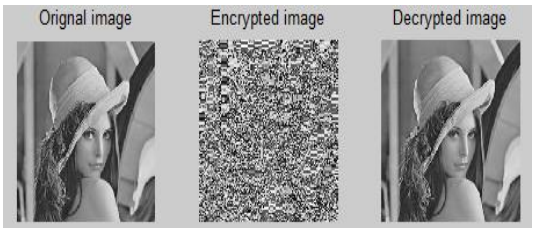

Fig. 9: Encryption and decryption with SF algorithm (lena)

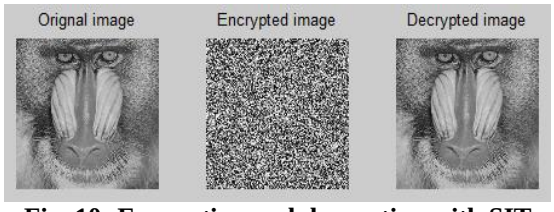

Fig. 10: Encryption and decryption with SIT (baboon)

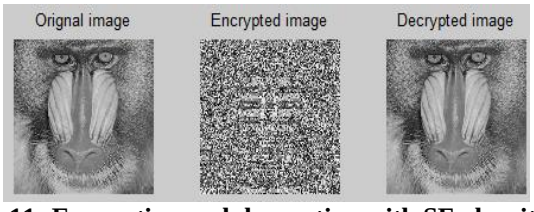

Fig. 11: Encryption and decryption with SF algorithm (baboon)

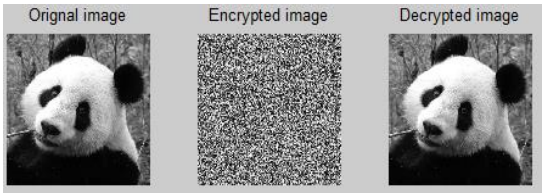

Fig. 12: Encryption and decryption with SIT (Panda)

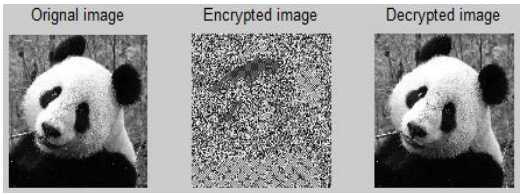

Fig. 13: Encryption and decryption with SF algorithm (Panda)

For the same images histogram of original and encrypted images is also plotted in the figure 14, 15, 16, 17, 18, 19 using both the algorithms. It can be noted that histogram plot of the encrypted images are uniformly distributed as compared to original image.

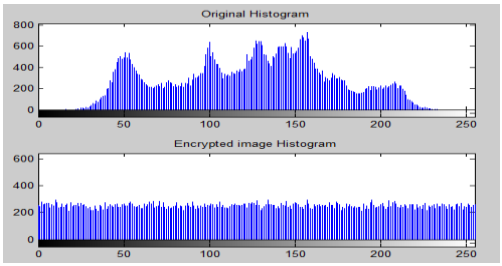

Fig. 14: Histogram of Lena image obtained from SIT

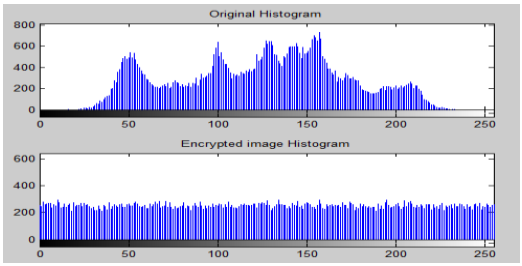

Fig. 15: Histogram of Lena image obtained from SF

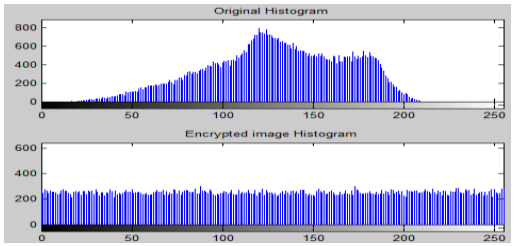

Fig. 16: Histogram of Baboon image obtained from SIT

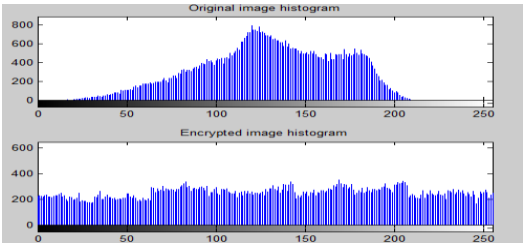

Fig. 17: Histogram of Baboon image obtained from SF

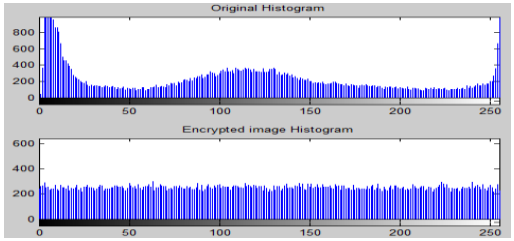

Fig. 18: Histogram of Panda image obtained from SIT

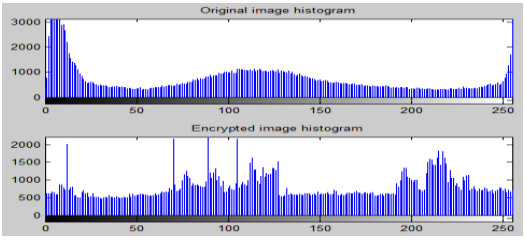

Fig. 19: Histogram of Panda image obtained from SF

Images results obtained after correlation is shown in figures 20, 21, 22, 23, 24, 25

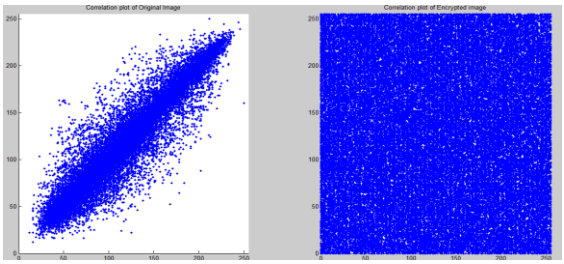

Fig. 20: Correlation plot of Lena image obtained from SIT

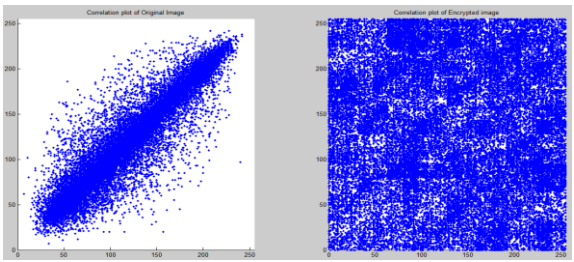

Fig. 21: Correlation plot of Lena image obtained from SF

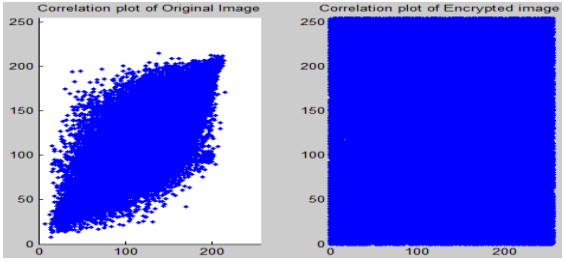

Fig. 22: Correlation plot of Baboon image obtained from SIT

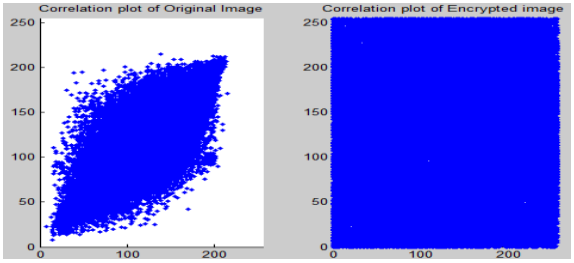

Fig. 23: Correlation plot of Baboon image obtained from SF

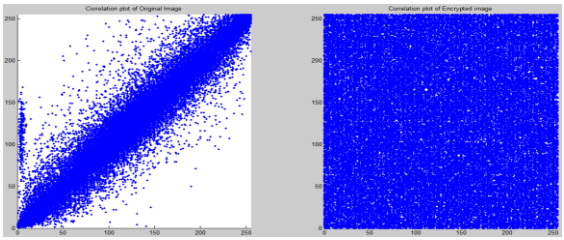

Fig. 24: Correlation plot of Panda image obtained from SIT

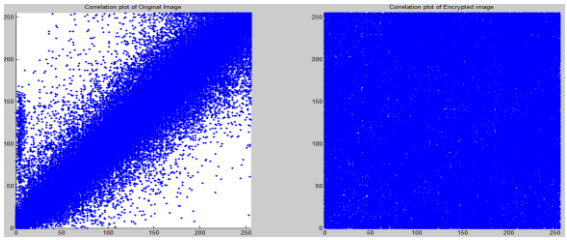

Fig. 25: Correlation plot of Panda image obtained from SF

Execution Time for both the algorithms with all three images is shown in table 4.

Table 4: Execution Time

Key	Image Name	Algorithm	Time
KEY 1	Lena	SIT	16.830409
		SF	87.177946
	Baboon	SIT	20.88466
		SF	134.31148
	Panda	SIT	17.209347
		SF	358.765815

Table 5: Average Execution Time

Key	Algorithm	Average time
KEY 1	SIT	18.308138666
	SF	193.418413666

Entropy Tables

Table 5: Sit – Key 1

Image	Entropy(Original)	Entropy(Encrypted)	Percentage Change
Lena	7.45092	7.996871	7.327296495
Baboon	7.231567	7.997284	10.588534961
Panda	7.4938	7.9972	6.717553177

Table 6: Secure Force – Key 1

Image	Entropy(Original)	Entropy(Encrypted)	Percentage Change
Lena	7.46184	7.949874	6.540397543
Baboon	7.231567	7.984318	10.409237721
Panda	7.5085	7.9002	5.216754345

NPCR and UACI Table

Table 7: Key – 1

Algorithm	Image	NPCR	UACI
Secure Force Algorithm	Lena	99.8471	16.1493
	Baboon	99.7696	13.5689
	Panda	99.84	25.9999
Sit Algorithm	Lena	99.5972	14.8698
	Baboon	99.6262	13.3172
	Panda	99.6292	22.6543

Average of NPCR Value for Key 1 For
Secure Force Algorithm: 99.8189
Sit Algorithm: 99.624633333

Average of UACI value for key 1
Secure Force: 16.5094
Sit Algorithm: 16.941233333

5. CONCLUSION

- **Visual Assessment:** In this assessment the encrypted image is compared with the original image and in both the algorithms taken for comparison (SIT, Secure Force), the encrypted image is highly distorted and bears no resemblance to the original image.
- **Entropy Values:** The entropy of the encrypted image taken in both the algorithms show a significant difference from the entropy of the original image. From the above values it can be concluded that the encrypted image which comes out of the SIT algorithm has higher entropy change when compared to the Encrypted image of the Secure Force Algorithm.

Hence SIT algorithm is better than Secure Force in this regard.

- **UACI AND NPCR VALUES:** From the above it can be concluded that the value of NPCR is close to 99 which is ideal for an encryption algorithm for both in the two cases taken (Key 1) and (Key 2). The UACI values are also more or less same for both the algorithms. NPCR and UACI test is performed by changing the pixel values and the resulting values are acceptable. Thus it can be deduced that the schemes have high sensitivity to change therefore resisting different differential attacks.
- **Histogram Analysis:** With respect to the experimental data collected, the histogram of the encrypted image produced as a result of the SIT algorithm has lesser peaks and is more level with respect to the same case taken for Secure Force Algorithm. Hence SIT is better in the case of distribution of pixels than Secure Force Algorithm.
- **Correlation Plot:** Correlation plots of the encrypted image taken for both the algorithms are very scattered and are uniformly distributed.
- **Correlation Coefficient:** The correlation coefficient is calculated for both the algorithms and are tabulated. It is noticed that all of them had a low correlation value and the correlation plot of the encrypted image was scattered and distributed. This proves that there is no information leakage and that both the algorithms can withstand statistical attacks.
- **Avalanche Test:** SIT: The Avalanche test of the algorithm shows that a single bit change in key or plain text brings around 49% change in the cipher bits, which is close to the ideal 50% change.
- **Secure Force**: The Avalanche test on the algorithm shows that the SF 64-bit can change 58.2% of cipher bits due to the change of one bit in text or key both of them are close to the ideal avalanche percentage of 50% with secure force generating more changes than SIT in the encrypted image with key change.
- **Execution Time:** From the average execution time table given above, it can be concluded that the time for the SIT algorithm to complete the process is almost twice as fast as the Secure Force Algorithm.

6. REFERENCES

[1] Mansoor Ebrahim, Chai Wai Chong "Secure Force: A Low-Complexity Cryptographic Algorithm for Wireless Sensor Network (WSN)", 29 Nov. - 1 Dec. 2013.

[2] Muhammad Usman, Irfan Ahmed, M. Imran Aslam, Shujaat Khan, Usman Ali Shah, "SIT: A Lightweight Encryption Algorithm for Secure Internet of Things", 2017.

[3] J. Luo, P. Papadimitratos, and J.P. Hubaux, "Gossicrypt: WSN data confidentiality against parasitic adversaries", SECON '08. 5th Annual IEEE Communications Society, June 2008.

[4] M. Ebrahim, S. Khan and U.B. Khalid," Symmetric algorithm survey: a comparative analysis". International Journal of Computer Applications 61(20), January 2013, pp. 12-19. USA.

[5] S.T.F. Al-Janabi, "Nahrainfish: A green cryptographic block cipher", IEEE, SIECPC 2011, April 2011, pp. 1-5, Riyad, Saudia.

[6] National Institute of Standards and Technology (NIST), "Advanced encryption standard (AES)," Federal Information Processing Standard (FIPS) 197, Nov. 2001.

[7] S. Misra, M. Maheswaran, and S. Hashmi, "Security challenges and approaches in internet of things," 2016.

[8] B. Karakostas, "A dns architecture for the internet of things: A case study in transport logistics," Procedia Computer Science, vol. 19, pp. 594–601, 2013.

[9] J. Wang, G. Yang, Y. Sun, and S. Chen, "Sybil attack detection based on rssi for wireless sensor network," in 2007 International Conference on Wireless Communications, Networking and Mobile Computing. IEEE, 2007, pp. 2684–2687.

[10] B. Ray, S. Douglas, S. Jason, T. Stefan, W. Bryan, and W. Louis, "The simon and speck families of lightweight block ciphers," Cryptology ePrint Archive, Report./404, Tech. Rep., 2013.

[11] Color Image Encryption in YCbCr Space, Published: 2016 IEEE, Xin Jin, Sui Yin, Xiaodong Li, Geng Zhao, Zhaohui Tian, Nan Sun1 , Shuyun Zhu

[12] Coloured Image Encryption Algorithm using DNA Code and Chaos Theory, Published: 5th International Conference on Computer & Communication Engineering, Samesh n Gobran, El-Sayed, M. Amr Mokhtar

[13] Improving for Chaotic Image Encryption Algorithm Based on Logistic Map, 2010 2nd Conference on Environmental Science and Information Application Technology, Ai-hongZhu, Lian Li

[14] Image Encryption using Various Transforms-A Brief Comparative Analysis, International Conference on Magnetics, Machines & Drives (AICERA-2014 iCMMD), Hemlata Agrawal, Dimple Kalot, Ankita Jain

[15] An Efficient and Effective Lossless symmetric Key cryptography algorithm for an Image, Niraj Kumar, Prof. Sanjay Agrawal

[16] A Unique Approach to Multimedia Based Dynamic Symmetric Key Cryptography, International Journal of Computer Science and Mobile Computing

[17] Data Embedding into Image Encryption using the Symmetric Key for RDH in Cloud Storage, International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 6, G. Preethi and N.P.Gopalan

[18] Memristor-Based Chaotic Circuit for Text/Image Encryption and Decryption, 2015 8th International Symposium on Computational Intelligence and Design Chenyu Yang , Qingqing Hu , Yongbin Yu , Rongquan Zhang , Yuanzhe Yao , Jingye Cai.

