Improved Lightweight SAFER Encryption by Using S-Boxes at Diffusion Layer for IoT Devices

Hamza Sajjad, Muhammad Junaid Arshad, Muhammad Sohail Akram
Department of Computer sciences, University of Engineering Technology
Lahore, Pakistan
imhamzasajjad@gmail.com, m.junaiduet@gmail.com, miansohailakram@gmail.com

***Abstract*—Advancements of communication increase the growth of IoT, where devices are interconnected with each other withthe help of the Internet. IoT is the combination of heterogeneous devices and each device has different hardwareand features. We have to make sure that communication is secured within the network. The physical layer of IoT hasbelonged to the devices and sensors which are the data generating part of the network. To send data over the networksecurely, a lightweight reliable encryption method for IoT devices over the network is required. IoT devices have lowcost and having a small circuit with low computation power. We required an encryption technique that needs to bevery suitable for this type of device, which requires low memory and low computation power for the encryption anddecryption of data. IoT network consists of heterogeneous devices with different security levels. During transmission,data encryption is the most important thing because of the heterogeneous nature of devices. We have different deviceswith different power, cost and hardware properties. There must be an encryption algorithm that fulfills our requirementsand suitable for low-cost devices as well. In this paper, we propose an encryption scheme that is designed for IoT on thebases of low memory and low computation power. This technique requires less memory for S-Boxes storage and hasmore security. It is based on the SAFER encryption algorithm which is improved with the help of S-Box implementation.**

I. Introduction

The concept of the Internet of Things is the new era of communication and which is growing very rapidly. Term IoT gives the idea of an environment where devices can communicate with other devices and also with peoples regardless of time and place just with the help of the Internet. This technology is based on the idea of creating a word were devices act on the bases of our demand [1]. This network consists of a lot of devices that are responsible for data generation. Each device has different sensors that are used for the collection of data from the environment. After processing that data information is sent to the server [2]. As this is the network of millions and millions of devices which are increasing every day and every device is sharing data. So, the security and the privacy of that data is the most important part of this network. End to end communication have to secure with the help of encryption from IoT devices to till the end of the network which is IoT application [3]. Through this, we have to make sure that our data is fully secure and this requires an encryption technology that ensures the security of our data and it should also need to be lightweight. As we know that IoT devices have very low computation power and low battery life so, we should have to go for a lightweight encryption algorithm [4]. Block encryption algorithms are very suitable for information security; this type of encryption is also useful for the encryption for wireless [5]. One of the most famous block algorithms is DES [6] [7] and after some security reasons triple, DES and AES [8] are introduced.
Block cipher algorithms from secure and fast encryption routine (SAFER) family are very suitable for this type of lightweight encryption and lightweight devices like Bluetooth. The SAFER have different Variants on the bases of Keys and numbers of rounds of encryption. SAFER K-64, SAFER K-128, SAFER SK-64, SAFER SK-128, SAFER+, and SAFFER++ have belonged to this series of the algorithm [9] [10].

- This algorithm is from block cipher so it performs encryption and decryption operations byte by byte. That is why it is suitable for embedded applications likes IoT.
- Suitable linear substitution transformation is performed on each round of the encryption.
- Each round of encryption has a different key which is generated by the addition of constant in the key. This will increase the security of our key [11].

With the security issues of IoT, we have to consider the power consumption of devices, computation power, memory space and performance cost. So we are considering different operations that a low power device can easily perform. Different operations for encryption are applied to the confusion and diffusion layer. We are using SAFER++ at the confusion layer and S-Boxes are used for the diffusion layer.

II. Literature Survey:

DES is belonging from block cipher family, which encrypts the block size of 64bit of plain text into the 64bit of ciphertext. The key size use is also 64bit which is converted into 56bit and for each round, we use 48bits from this key to operate on plain text. We convert our plain text into two equal parts of 32bits and apply the function on the right side of the block with the key. The operations that we perform in this encryption are initial and final permutation, XOR, bit shift for the key, S-BOX, and P-BOX. This encryption algorithm is failed due to the less secure and much easy to break this encryption with the help of high power computers. To overcome the security

of this algorithm we extend it to triple-DES which shows more resistance toward the attack. Later this DES algorithm is replaced with the AEC algorithm which has high security but also requires a lot of computational power which is not suitable for IoT devices [6].

Vijay Dahiphale et al .in [12] propose a new encryption method which is the lighter version of ANU and name as ANU-II. This method uses the block of 64bits for encryption and a key size of 80-128bits. This version is more effective than the previous version because of less memory, low latency, and less power. These factors make this encryption lightweight. ANU-II requires 20 percent less memory than the ANU. The operation that we have to perform are S-BOX, Shift operator and XOR. As compare to the ANU, ANU-II use only one S-BOX rather than multiple S-Boxes, less shift operation and Couple of XOR. So this is the very lightweight encryption method as is consuming less memory, less execution time due to less operation and consume less power due to less computation [12]. But it is more lightweight so the security of the encryption can also be compromised.

Seddiq Q. Abd Al-Rahman et al in [13] present an algorithm that is low cost and also a lightweight encryption scheme. A concept of whitening is introduced through which we start the encryption. Whitening is coming with the bitwise XOR between cipher key and plain text before performing the encryption round and after the last round. Through this, we increase the strength of our key. The second thing which is involved in this scheme is the Mixing column name as Mixcolayer. After column mixing the next step is to use 4-bit S-Box for the data. They are not using 8-bit S-Box or more than this because of the energy consumption. Then the shift layer is come with the concept of shifting the column and make the output different then input and this process are known as the diffusion process of this process. Then 64bit key is used which is generated with the help of a key generator which takes 80bit to 128bit key set and generates the 64bit key for each round of data. This encryption involves the 20 round of this encryption. This scheme involves the MixCoLayer which need high computation and also involve 20 round for signal encryption which is not suitable for the small IoT devices just because of high consumption power.

Musaria K. et al .in [10] show the MATBL implementation of SAFER+ which is from the family of Secure and Fast Encryption Routine (SAFER). This is a very simple algorithm with high throughput and low memory requirements. SAFER+ is also belonged from block cipher family and encrypt/decrypt the block of 128bits and the key length is 128bits, 192bits, and 256bits on the bases of a number of the round. We use 2 keys for each round and keys are generated with the help of key scheduler. For each round bytes 1,4,5,8,9,12,13 and 16 need to XOR with key and modulo 256. Reaming bytes need to add with key and modulo 256. After that, the result of 1,4,5,8,9,12,13, and 16 need to be 45e mod 257 and other bytes need to be log45(x) mod257. In the end, we will perform the matrix multiplication. The 16bytes are multiplied with the 16x16 matrix and modulo 256 and give the output [10]. But at the decryption end, we need to find the inverse of this matrix which is a very complex task and we need to replace this to make it as simple as we can.

XIAXIA GUO et al in [14] comes up with the SAFERFermat. He implements the S-LP structure on this, in which nonlinear transformation (S) on the confusion layer. This is the same as we see in Musaria K. et al .in [10], but linear transformation(LP) is applied to the diffusion layer. At the diffusion layer, FNTT is used in SAFER-Fermat. The rest of the part is the same as SAFER [14]. But FNTT is also not a simple theory to apply for IoT devices. We need a simple, more secure and require less computational power.

III. Key Parameters for encryption Scheme

We have three parameters that we need to consider for the best solution. Cost (memory, processing, hardware upgradation), complexity and security are the key factors need to be considered for the design of the encryption scheme. They all are directly proportional to each other. If you make an encryption method more complex than its security will also increase and with the increase in complexity of encryption scheme we also need to increase the computational power or we may require some hardware changing. So if you want to design the encryption method for IoT these will be the key parameters that must be in your mind. It is very important to take all the parameters in a very balanced way and its very important for reliable encryption methods.

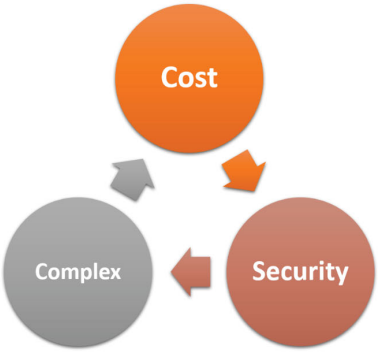

Fig. 1. Key Factors for the encryption scheme

These parameters are observed on the basses of the literature review that we have done and understood different encryption techniques. Here we are going to present an improved and better form of encryption method that was presented by Xia Xia Guo et al in [14]. During this we consider these parameters, and make our encryption more reliable by having high security and it is less complex and it will also cost you less.

IV. Methodology

To understand the principle of the proposed encryption method we have to follow some of the basic principles of

cryptography. We are using a block cipher encryption method so we have to understand the methods behind this. After that, we have to understand the importance of the confusion and diffusion layer and which type of operation we can perform. In the end, we also have an understanding of S-Box and how we can use them.

A. Block Cipher

Block cipher is the family member of symmetric key cryptography and the most important encryption mechanism. To understand our proposed method, we have to understand the basic principle of a block cipher and why we are using this. As we know, IoT devices are low computational and low power devices so stream cipher or asymmetric encryption are not good choices for this because they are very hardware hungry encryption mechanisms. In-stream cipher bit by bit data is processed which needs very high speed. That's why we are considering block cipher form symmetric encryption and decryption.

Block cipher performs byte by byte operation for encryption and decryption. It takes byte or bytes from the plain text and performs some operations on it with the help of key and gives the encrypted data. While decryption it will also perform decryption method byte by byte on encrypted data. Some famous cipher forms this family is DES, triple DES and AES [6]. It's easy to operate on a block of fix length so this mod of encryption is easy and has low computation boxes. Block cipher shows in Fig. 2:

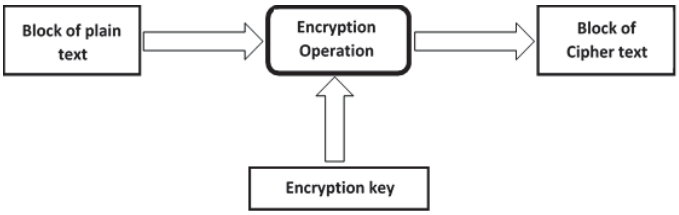

Fig. 2. Block cipher basic structure

V. Basic SAFER S-Box Encryption Algorithm

This simple encryption algorithm consists of two layers, confusion layer which helps us to convert our plain text into the clueless ciphertext and second is the diffusion layer which increases the redundancy of that encrypted data. In this given algorithm we are using SAFER ++ at the confusion layer and at the diffusion layer we are going to consider S-Box. Byte by byte function is performed because this algorithm is under the category of block chipper. The basic structure of SAFER S-Box given below in Fig. 3:

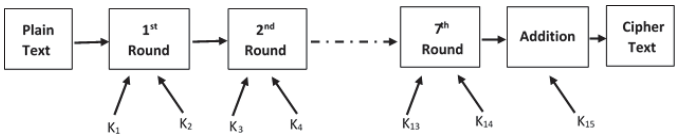

Fig. 3. SAFER S-Box Round Structure

VI. SAFER S-Box Structure for Encryption

On the bases of Shanon's proposed methodology, a single round of encryption process is subdividing into the confusion and diffusion layers [15].

A. Confusion Layer

The proposed algorithm involved the implementation of different operations. According to the SAFER ++ four operations are performed; addition, XOR, logarithm and exponential and at the decryption end inverse of every operation will be performed. We will take 16 bytes for plain text and performs 7 round of encryption and give the 16bytes of ciphertext. For each round, we are using 2 keys through which we are making this more secure but in the last round only a single key which is simply added, the total number of keys are 15 used for one-time encryption. XOR is performed on 1st, 4th, 5th, 8th, 9th, 12th, 13th, and 16th block of bytes and addition is performed on remaining bytes 2nd, 3rd, 6th, 7th, 10th, 11th, 14th, and 15th bytes. We will use modulo-256 for Addition and XOR and this function has especially known as hybrid XOR/modulo addition HXMD [14].

The K2i-1 key is used for this step. After this, we will perform exponential on bytes on which we already performed XOR function and logarithm function is performed on the bytes on which we performed addition. The bytes coming after the exponential are now treated with addition and bytes coming after logarithm are now treated with XOR. The key for this step is changed, K2i is the key for this step and this is the 2nd key for this step [16]. Musaria K. Mahmood et al. [10] explain the Key generation algorithm is explained.

Two nonlinear functions "log" and "ex" are also used for each block of the byte in each round. on different bytes of this layer. These operations are also under the confusion layer. we are defining X and Y for these functions and "n" is representing the byte on which these functions are performed.

$$X(n) = 45^n Mod257$$

$$Y(n) = log_{45}(n) Mod257$$

We will perform X(a) on 1, 4, 5, 8, 9, 12, 13 and 16 bytes position and Y(n) will be for 2, 3, 6, 7, 10, 11, 14, and 15 bytes position. Before performing this we have to perform XOR on the bits that are entering in X(a), plus function on the bytes that are entering in Y(n). After performing X(n) and Y(n) we performed plus on bytes that are exiting from X(n) and XOR on bytes that are exiting from Y(n). For XOR, plus Mod 257 is used.

$$XOR = N \oplus K_n Mod257$$

$$Plus = N + K_n Mod257$$

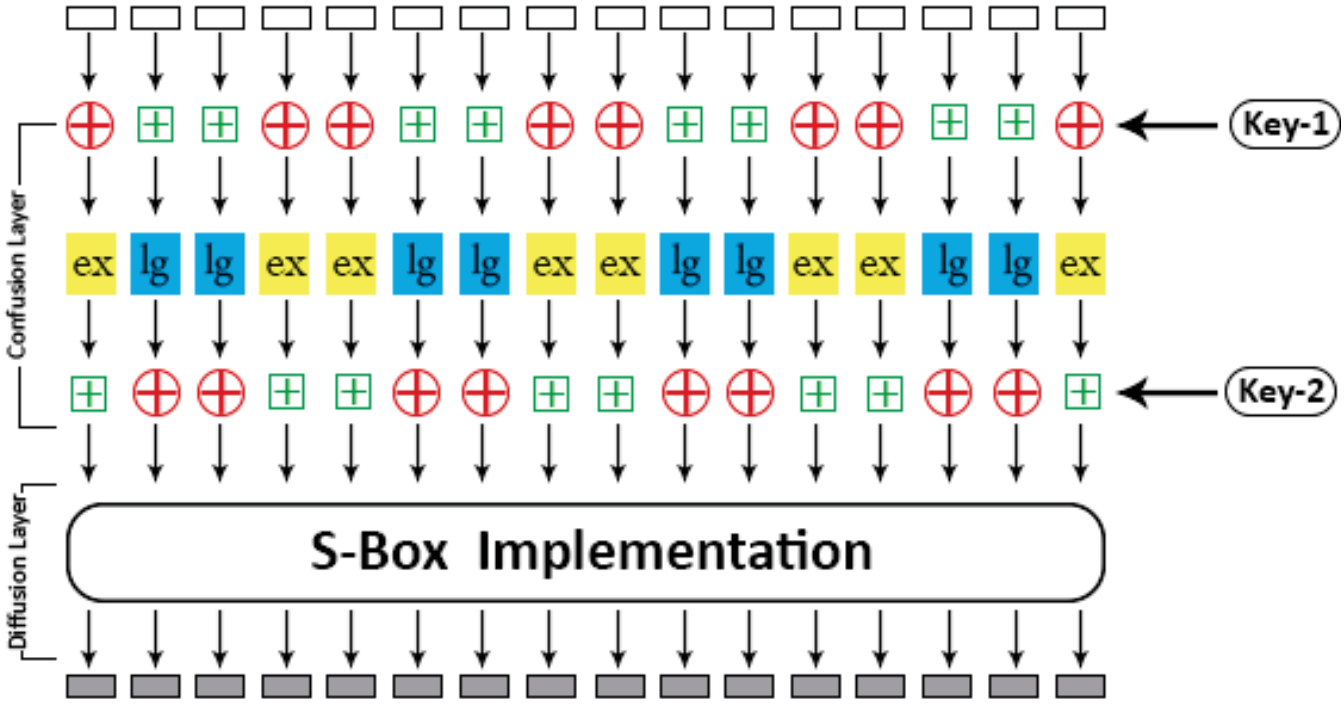

Fig. 4. SAFER S-Box Encryption Block Diagram

Here

$$K_n$$

is the key number. 15 different keys are used and this is telling us which number of the key we are using. These four are very simple to function that we have to perform for the confusion layer [16].

B. Diffusion layer

All these functions are performed on the confusion layer where we performed functions to turn our plain text into the clueless ciphertext. This layer is introducing as a diffusion layer which will take the strength of encryption of ciphertext one step up and it increases the redundancy of our ciphertext. As we are improving the cipher encryption for IoT devices, we have to make it simpler and secure. Musaria K. Mahmood et al. [10] use the 16x16 matrix and Xiaxia Guo et al. [14] use FNTT for the diffusion layer. These type of operations require more computation power and that increase the cost and power consumption for devices. By keeping all the restrictions about IoT devices in mind we chose S-Boxes for the diffusion layer. There are some rules for the selection of S-Box for each byte.
You can use these rules for the selection of S-Boxes.

Algorithm 1 Algorithm for the Selection of S-Boxes at Diffusion layer

1) We have used 4 S-Boxes for seven rounds.
2) Each byte coming from the confusion layer is split-up into two equally divided chunks. The first chunk of bits is used for row and the second chunk is used for column number.
3) Corresponding to that row and column number we have a value stored which we are using and considering as new data.
4) For the 1st round of encryption, we change S-Box after each byte.
5) For the 2nd round of encryption, S-Box is changing after 2-bytes.
6) For the 3rd round of encryption, S-Box is changing after 3-bytes.
7) For the 4th round, S-Box is changing after 4-bytes.
8) The process for the selection of S-Box for remaining rounds is in reverse order till the last round. Like for 5th round we are changing after 3-bytes, for 6th round, we are changing after 2-bytes and for 7th round, it is changed for each byte.

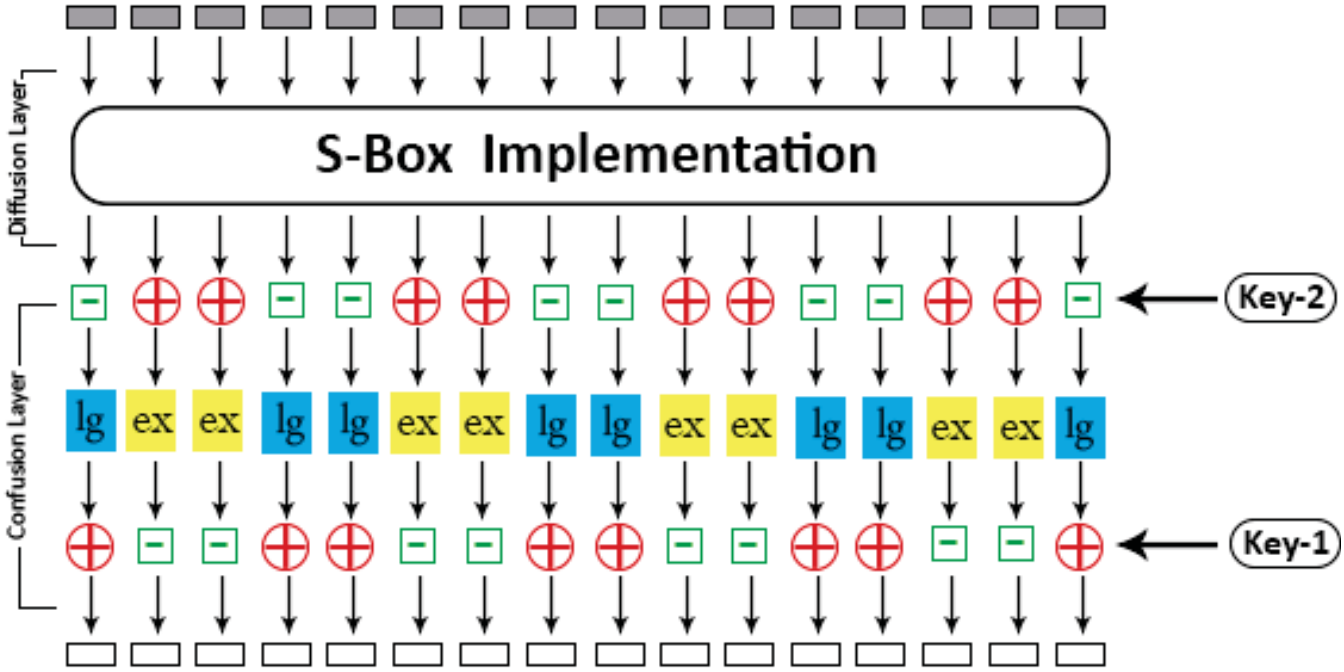

Fig. 5. SAFER S-Box Decryption Block Diagram

We are using 4 S-Boxes just because they require less memory as compared to use a new box for each round of encryption.By using this algorithm we are manipulating the places of S-Boxes for each round in this way we are making the diffusion layer strong. The manipulation will be secreted between the sender and receiver and we can also change the manipulation arrangement of S-Boxes. Bypassing each byte with different S-Boxes for each round make the text more secure and if an attacker gets the text and also the S-Boxes, then he must have to know the manipulating sequence for the S-Boxes. We are making data secure with the help of S-Boxes and making S-Boxes secure with the help of manipulating the positions of S-Boxes. If an attacker needs to encrypt the data he must have the S-Boxes and also the sequence of manipulation of S-Boxes.To can get an understanding of the encryption with the help of the Block diagram of SAFER S-Box that is shown in Fig. 4.

This is the single-cycle for encryption 16 bytes. This block diagram is clearly showing each process that is involved in the encryption. In this diagram, we diffusion layer consists of S-Box implementation and for the implementation process, you have the algorithm that we just presented. To construct good S-Boxes there must be some certain criteria for that. Amjad Hussain Zahid et al. [17] proposed a technique for the construction of substitution Boxes (S-Box) and have good results as compared to the previously proposed S-Boxes construction methods. We can consider this technique for the construction of S-Boxes for our technique [17].

VII. SAFER S-Box Structure for Decryption

For decryption, we have to know the inverse process of each operation which we have used for encryption. In this method, we have used Plus function, XOR, logarithm, exponential. At the diffusion layer, S-Boxes are implemented. First, we get the first chunk bit from selected byte value and replaced with the row and the second chunk is replaced with column number form the S-box. Now we are on the confusion layer and here we performed four different operations. First, we have two functions XOR and Plus. For XOR function we are taking XOR again and for Plus, subtract is used as an inverse of the function and this is done with the help of the first key. For now, two more functions known as logarithm and exponential are remaining. We just need to replace the logarithm with exponential and exponential for logarithm [10] [14]. After that, we performed XOR and subtract again with the 2nd key. After completing all the rounds, the result is our plain text. Method of encryption and decryption involves simple operations that's why it's not much complex for any kind of IoT devices. The process of decryption is also described with the help of the figure in Fig. 5.

VIII. Analysis of SAFER S-Box Algorithm

This algorithm is covering both sides which an algorithm requires for IoT devices, it is lightweight and also more secure. This encryption algorithm involves SAFFER with S-Box. SAFER is the most outstanding algorithm for low power devices and Bluetooth devices [10]. So that's why we chose SAFER on the confusion layer, which has simple and fast operations. On the diffusion layer, Musaria K. et al .in [10] are using a 16x16 matrix which makes this system complex because for decryption we have to find t he i nverse o f the matrix which is not a suitable option for low computational devices. XIAXIA GUO et al in [14] are also using SAFER for the confusion layer but at the diffusion layer, FNTT [13] is used which also requires very computation. So we need some simple method for the diffusion layer and we chose S-Boxes for that. But like DES and other algorithms, we are not using new S-box for each round which consumes more memory to store S-Boxes. Vijay Dahiphale et al .in [12] are just using one S-Box which is also not a good option because it compromises the security of the encryption. As if we use only one S-Box its very easy for an attacker to compromise your security because he only needs one S-Box to encrypt the whole encrypted text into plain text and if we use a new S-Box for each round we need more number of S-Boxes which is not good for low memory devices, so that's why we chose 4 S-Boxes. They require less memory as compared to use a new S-Box for each round like we were doing in DES and it's also good that we are using multiple S-Boxes not relying on a single S-Box. As I already mention that if you follow this encryption algorithm the attacker needs S-Boxes and also the sequence S-Boxes because they don't know at which position which S-Box is used. Low cost and the lightweight algorithm is presented bySeddiq Q. Abd Al-Rahman et al in [18] which is involved the bitwise XOR and MixCoLayer type of complex functions which are not much suitable. After understanding these issues we used 4 S-Boxes which are half than the number of rounds and we are not changing S-Box for each round. We using all the boxes for each round by just manipulation its positions. By using fewer S-Boxes, we are saving device memory and required less storage capacity for S-Boxes and by manipulating the positions of S-Boxes we can achieve high security at the diffusion layer. This is the main advantage of our scheme on other schemes that were previously presented. A comparison with the already proposed algorithm is also presented in Table I.

TABLE I. Comparison with already proposed Algorithms

Algorithm	Complexity	Time	Cost	Security
ANU-II [12]	Less	Less	Low	Low
XTEA with AES [18]	Less	High	High	High
SAFER+ 128-Key [10]	High	High	High	High
SAFER-Fermat [14]	Less	High	High	High
SAFER With S-BOX (Proposed)	Less	Less	Less	High

IX. Conclusion

This paper presents an improved form of the SAFER++ encryption algorithm by using the S-Boxes. which involved less complex mathematical implementations and make this encryption algorithm more simple and secure. We design an encryption algorithm that gives us more security, less cost and less complex. This scheme is developed by keeping in mind the three key parameters of IoT devices. Encryption has divided into two-part confusion and diffusion layer, at confusion layer SAFER++ block cipher has used for encryption, which has simple and very less complex operations which a device can handle very easily.At the diffusion layer, we have used less number of S-Boxes and manipulate the positions so that each byte is processed with the help of different Box at different round. This scheme requires low memory to store S-Boxes because of fewer S-Boxes and we are not using different S-Box for each round of encryption, in this way we are saving our resources of IoT devices. This encryption algorithm is suitable for low memory devices. As we know IoT devices have low memory and cannot handle complex computations. So, this encryption technique fulfills all the requirements for the encryption of data for IoT devices.

References

[1] G. T. anthoshš Krishna B V, "Optimum information transmission through a channel with unknown parameters," pp. 107–111, 2017.

[2] P. P. Marioš Frustaci, "Evaluating critical security issues of the iot world: Present and future challenges," *IEEE INTERNET OF THINGS JOURNAL*, vol. 5, pp. 2483–24 958, 2018.

[3] D. S. S. s. Sridharš, "Intelligent security framework for iot devices," pp. 1–5, 2017.

[4] S. Y. J. H. P. s. Singhš, P. K. Sharma, "Advanced lightweight encryption algorithms for iot devices: Survey challenges and solutions," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–18, 2017.

[5] N. M. R. K. J. Patelš, "Secure end to end data aggregation using public key encryption in wireless sensor network," *Int. J. Comput. Appl.*, vol. 122.

[6] williamš, "Cryptography and network security principle and practices, chapter 3, block cipher and the data encryption standard."

[7] D. Coppersmithš, "The data encryption standard (des) and its strength against attacks," 1992.

[8] R. V. Daemenš J., "The design of rijndael. "aes - the advanced encryption standard," p. 238.

[9] J. L. Masseyš, "Safer k-64: A byte-oriented block-ciphering algorithm," *Proc. Fast Softw. Encryption*, pp. 1–17, 1993.

[10] M. K. M. et al., "Matlab implementation of 128-key length safer+ cipher system," *Int. Journal of Engineering Research and Application*, vol. 7.

[11] A. A. et al., "A novel symmetric cryptography algorithm for fast and secure encryption," pp. 1–6, 2015.

[12] G. B. Vijayš Dahiphale, "Anu–ii: A fast and efficient lightweight encryption design for security in iot," pp. 131–137, 2017.

[13] M. F. A.-G. S. Boussakta and J. A. Neasham, "Fermat number transform diffusion's analysis," pp. 237–240, 2011.

[14] J. H. IAXIAš GUO, "A complexity-reduced block encryption algorithm suitable for internet of things," *IEEE Access*, pp. 54 760–54 769, 2019.

[15] C. E. Shannonš, "Communication theory of secrecy systems," *Bell Labs Tech. J.*

[16] A. B. D. Cannière and G. Dellkrantz, "Crypt analysis of safer++," pp. 195–211, 2003.

[17] M. J. A. Amjadš Hussain Zahid, "A novel construction of efficient substitution-boxes using cubic fractional transformation," vol. 21.

[18] A. M. S. Seddiqš Q, Abd Al-Rahman, "Nvlc: New variant lightweight cryptography algorithm for internet of things," pp. 176–181, 2019.