IOSR Journal of Mobile Computing & Application (IOSR-JMCA)
e- ISSN: 2394-0050, P-ISSN: 2394-0042.Volume 12, Issue 2 (Mar. – Apr. 2025), PP 10-17
www.iosrjournals.org

Hybrid Cryptosystem for Preserving Data Privacy in IoT Application

Nahla F.Ibrahim[1] and Johnson I. Agbinya[2]
[1]*Faculty of Computer Science, King Khalid University, 00966Abha, Saudi Arabia*
[2]*School of Information Technology and Engineering, Melbourne Institute of Technology, 0061Melbourne Victoria, Australia*
Corresponding Author: Nahla F.Ibrahim

***Abstract**: Over the recent years, several smart applications like RFID"s, sensor networks, including industrial systems, critical infrastructures, private and public spaces as well as portable and wearable applications in which highly constrained devices are interconnected, typically communicating wirelessly with one another, working in concert to accomplish some task. Advanced safety and security mechanisms can be very important in all of these areas. Light weight cryptography enables secure and efficient communication between networked smart objects. On the stand feistel table, proposed algorithm is a suitable lightweight cryptographic algorithm used in medium security systems. It is a 64-bit block cipher and requires 16-bit key to encrypt the data. Simulations result showed the proposed algorithm provides substantial security in just five encryption rounds.From simulation result, we concluded that our proposed algorithm gave a good performance when compared with DES and showed a good alternative to proposed as network security and privacy on Internet of Things environments.*

***Key word**: Internet of Things (IoT); lightweight cryptography; Feistel Networks; KHAZAD*

I. Introduction

The Internet of Things (loT) promises to be the next big revolution of the World Wide Web. It has a very wide range of applications, ranging from smart cities, smart homes, monitoring radiation levels in nuclear plants, animal tracking, health surveillance and a lot more.When objects, people or animals are provided withunique identifiers and are able to communicate with each otherwithout human intervention, it is referred as the Internet ofThings or Internet of Objects. Four major challenges in loT are powermanagement, the deployment of IPv6, standardization andsecurity [8]. Data Security is a primary issue in any wireless cryptographic protocol, a cryptographic algorithm is an essential part in network security. One of the state-of-the-art techniques is “Lightweight Cryptography (LWC)”. Lightweight cryptography is a cryptographic algorithm or protocol tailored for implementation in constrained environments like RFID"s, sensor networks, healthcare, the Internet of Things, cyber-physical systems, distributed control systems, indicators, measuring devices, custom controllers, smart power system etc.[9].

The rest of the paper is organized as follows, in Section 1is the introduction parts,in Section 2presents the related work of this research, in Section 3, architecture and functioning of the proposed algorithm is presented, in Section 4, Evaluation of proposed algorithm is discussed, in Section 5, Simulation Result and finally some conclusions are given in Section 6.

II. Related work

This section shows some other works from related fields. A number of studies of the eminent researchers are done in literature to improve the security and privacy in IoT. We discussed more relevant and recent available solutions for security, privacy and hence improve small cryptographic algorithms for IoT.

In [10], authors proposed of a secure data transmission using AES in IoT. The main idea for this work, proposed mechanism increase throughput and execution time by enhanced AES algorithm in which number of rounds or generation of private key increases that will help in generation of more secure encrypted key through which devices can transmit data in a secure manner.

In [11], authors proposed an ultra-lightweight cipher ANU. ANU is a balanced Feistel-based network. The main idea for this solution Algorithm is designed to generate the good S-box according to lemma and also to find the minimum number of active S-boxes.

From [12] author designed RECTANGLE block cipher based on the bit-slice technique in a lightweight manner, hence to achieve not only a very low-cost in hardware but also a very competitive performance in software. As a result, RECTANGLE adopts the SP-network structure. The substitution layer (S-layer) consists of 16 4 × 4 S-boxes in parallel. The permutation layer (P-layer) is composed of 3 rotations.

In [13], it was the study of the modified blowfish algorithm implemented on FPGA. There are two changes proposed which are round of feistel, the number of rounds was reduced to 8 rounds and 4 rounds., and The key size was changed from 448 bit to 384 bit, 320 bit, 256 bit, 192 bit, 128 bit and 64 bit.. The result showed that FPGA implementation of the modified blowfish algorithm provides a reducing the rounds of feistelreduce total encryption time, give greater throughput and not affect the avalanche effect significantly. It also showed that larger key length needs more resources to implement in FPGA. However, traditional cryptography focus on the solutions in providing high levels of security, ignoring the requirements of constrained devices.

III. Proposed algorithm

The proposed algorithm is a symmetric block cipher thatcan be effectively used for encryption and safeguardingof data. The objective is to reduce execution time. In the symmetrickey algorithm, theencryption process consists of encryption rounds; each roundis based on some mathematical functions to create confusionand diffusion. Increase in a number of rounds ensures bettersecurity and privacy but eventually results in the increase in the consumptionof constrained energy [1]. The cryptographic algorithms areusually designed to take on an average 10 to 20 rounds to keepthe encryption process strong enough that suits the requirement of the system. However the proposed algorithm is restricted tojust five rounds only, to further improve the energy efficiency, each encryption round includes mathematical operations that operate on 4- bits of data. The details of the proposed algorithm design are discussed in section 3.1,3.2and 3.3.

Another vital process in symmetric key algorithms isthe generation of the key. The key generation process involvescomplex mathematical operations. In WSN environment theseoperations can be performed wholly on decoder [6],[2],[3], on the contrary in IoT the node themselves happens toserve as the Internet node, therefore, computations involvedin the process of key generation must also be reduced to theextent that it ensures necessary security. In the sub-sections, the process of key expansion and encryption are discussed in detail. Some notations used in the explanation are shown inTable1

Table1. Notations

Notation	Function
$\oplus$	XOR
$\Vert$,||	Concatenation

3.1 Key Expansion

The most fundamental component in the processes ofencryption and decryption is the key. It is this key on whichthe entire security and privacy of the data is dependent, should this key beknown to an attacker, the secrecy of the data is lost. Thereforenecessary measures must be taken into account to make therevelation of the key as difficult as possible. The feistel basedencryption algorithms are composed of several rounds, eachround requiring a separate key. The encryption/decryption ofthe proposed algorithm is composed of five rounds; therefore,we require five unique keys for the said purpose. To do so,we introduce a key expansion block which is described in this section.To maintain the security and privacy against exhaustive search attackthe length of the true key $k_t$must be large so that it becomesbeyond the capability of the enemy to perform 2^{k_t-1}encryptions for key searching attacks. The proposed algorithm is a 64-bits block cipher, which means it requires the 16-bits key to encrypt64-bits of data. A cipher key (K_c) of 64-bits is taken as an input from the user. This key shall serve as the input to thekey expansion block. The block upon performing substantialoperations to create confusion and diffusion in the input key will generate five unique keys. These keys shall be used in theencryption/decryption process and are strong enough to remainindistinct during an attack.

The architecture of the key expansion block is adopted fromMuhammad Usman et al. [7 Figure1] with 16-bit modification. The block uses an $\boldsymbol{f}$- function which is influenced bytweaked Khazad block cipher [4]. Khazad is not a feistelcipher and it follows wide trial strategy. The wide trial strategyis composed of several linear and non-linear transformationsthat ensure the dependency of output bits on input bits in acomplex manner [5]. Detailed explanations of the components of key expansion are discussed below:

a. In the first step, the 64-bits cipher key (K_c) divided into 4-bits segments.

b. The $\boldsymbol{f}$-function on 16-bits data. Then, four$\boldsymbol{f}$-functions are used, 16-bits for each $\boldsymbol{f}$-function obtained by performing initial substation of segments of cipher key(K_c) as shown in equation (1).

$$K_{bi}f = \Vert_{j=1}^{4} Kc_{4(i-1)+i} \qquad (1)$$

where$i = 1 \ to \ 4$ for first four rounds keys.

c. The next step is to get$(K_{a_i} f)$ by passing the 16-bits of $(K_{b_i} f)$ to the $\boldsymbol{f}$-function as shown in equation (2).

$$K_{a_i} f = \boldsymbol{f}(K_{b_i} f) \qquad (2)$$

d. $\boldsymbol{f}$-function is comprised of P and Q table. This table performs linear and non-linear transformations resulting inconfusion and diffusion as illustrated in Figure1.

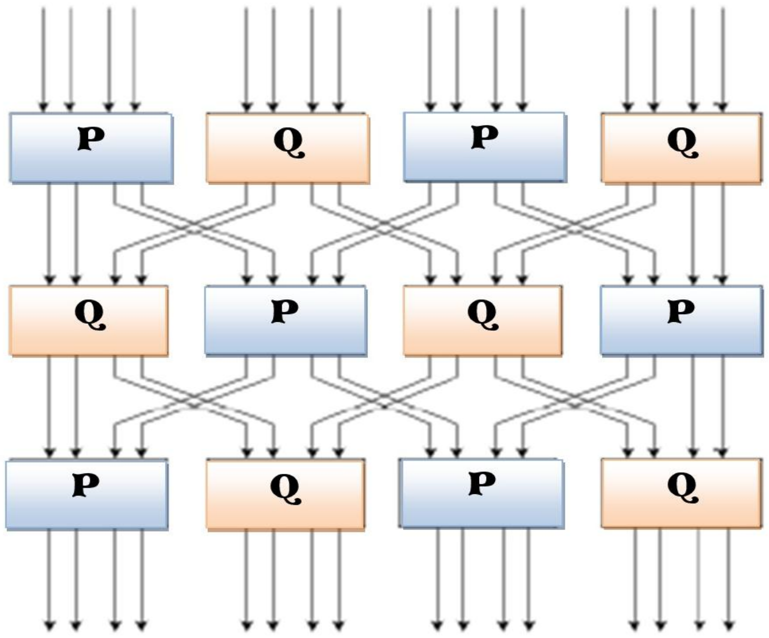

Figure1. KHAZAD F-Function

e. The transformations made by P and Q are shown in theTables 2.

Table2. P TABLE & Q TABLE

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

f. The output of each f-function is arranged in 4 ×4 matrices named K_m shown below:

g. To obtain round keys, K1, K2, K3, and K4 the matricesare transformed into four arrays of 16-bits that we callround keys (K_r). The arrangements of these bits are shownin equations (7), (8), (9) and (10).

$$Km_1 = \begin{bmatrix} Ka_1f_1 & Ka_1f_2 & Ka_1f_3 & Ka_1f_4 \\ Ka_1f_5 & Ka_1f_6 & Ka_1f_7 & Ka_1f_8 \\ Ka_1f_9 & Ka_1f_{10} & Ka_1f_{11} & Ka_1f_{12} \\ Ka_1f_{13} & Ka_1f_{14} & Ka_1f_{15} & Ka_1f_{16} \end{bmatrix} \quad (3)$$

$$Km_2 = \begin{bmatrix} Ka_2f_1 & Ka_2f_2 & Ka_2f_3 & Ka_2f_4 \\ Ka_2f_5 & Ka_2f_6 & Ka_2f_7 & Ka_2f_8 \\ Ka_2f_9 & Ka_2f_{10} & Ka_2f_{11} & Ka_2f_{12} \\ Ka_2f_{13} & Ka_2f_{14} & Ka_2f_{15} & Ka_2f_{16} \end{bmatrix} \quad (4)$$

$$Km_3 = \begin{bmatrix} Ka_3f_1 & Ka_3f_2 & Ka_3f_3 & Ka_3f_4 \\ Ka_3f_5 & Ka_3f_6 & Ka_3f_7 & Ka_3f_8 \\ Ka_3f_9 & Ka_3f_{10} & Ka_3f_{11} & Ka_3f_{12} \\ Ka_3f_{13} & Ka_3f_{14} & Ka_3f_{15} & Ka_3f_{16} \end{bmatrix} \quad (5)$$

$$Km_4 = \begin{bmatrix} Ka_4f_1 & Ka_4f_2 & Ka_4f_3 & Ka_4f_4 \\ Ka_4f_5 & Ka_4f_6 & Ka_4f_7 & Ka_4f_8 \\ Ka_4f_9 & Ka_4f_{10} & Ka_4f_{11} & Ka_4f_{12} \\ Ka_4f_{13} & Ka_4f_{14} & Ka_4f_{15} & Ka_4f_{16} \end{bmatrix} \quad (6)$$

$$K_1 = a_4 \ddagger a_3 \ddagger a_2 \ddagger a_1 \ddagger a_5 \ddagger a_6 \ddagger a_7 \ddagger a_8 \ddagger a_{12} \ddagger a_{11} \ddagger a_{10} \ddagger a_9 \ddagger a_{13} \ddagger a_{14} \ddagger a_{15} \ddagger a_{16}$$
(7)

$K_2 = b_1 \# b_5 \# b_9 \# b_{13} \# b_{14} \# b_{10} \# b_6 \# b_2 \# b_3 \# b_7 \# b_{11} \# b_{15} \# b_{16} \# b_{12} \# b_8 \# b_4$ (8)

$K_3 = c_1 \# c_2 \# c_3 \# c_4 \# c_8 \# c_7 \# c_6 \# c_5 \# c_9 \# c_{10} \# c_{11} \# c_{12} \# c_{16} \# c_{15} \# c_{14} \# c_{13}$ (9)

$K_4 = d_{13} \# d_9 \# d_5 \# d_1 \# d_2 \# d_6 \# d_{10} \# d_{14} \# d_{15} \# d_{11} \# d_7 \# d_3 \# a_4 \# d_8 \# d_{12} \# d_{16}$ (10)

$K_5=[K_1 \oplus K_2] \oplus [K_3 \oplus K_4]$(11)

3.2 Encryption

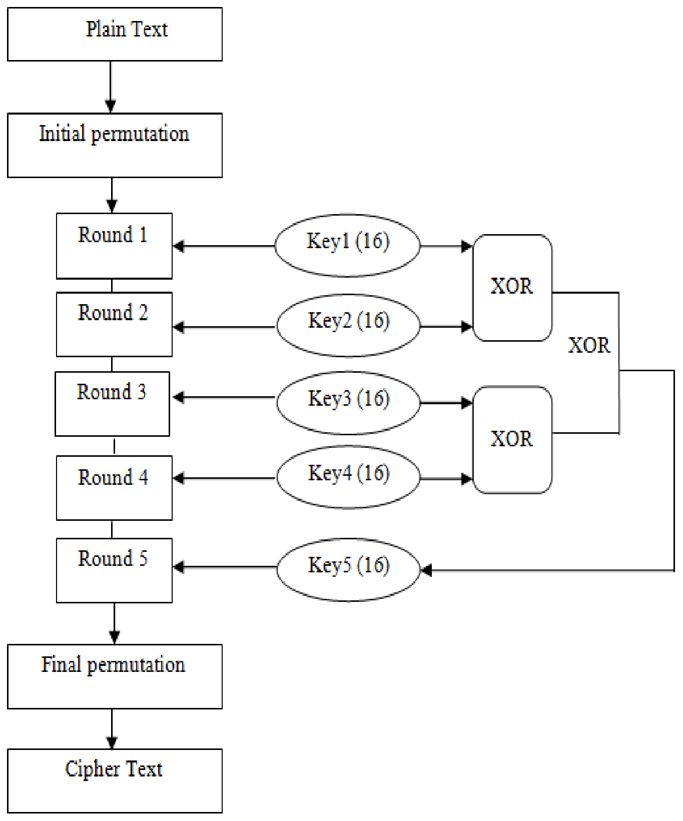

Figure2. The overall feistel structure of the proposed algorithm (Encryption)

3.3 F- function

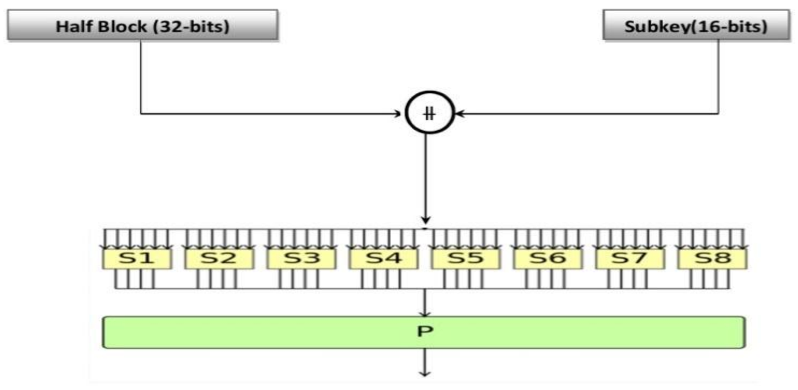

Figure3. F-Function of feistel table

1. Performance Evaluation Criteria

For the investigation, the parameter was used to quantify the information required for a comparison between the existing algorithm and the proposed algorithm and these parameters are as under:

2. Simulation Result

1) *Execution Time:* One of the fundamental parameter forthe evaluation of the algorithm is the amount of time it takes toencode and decode a particular data. The proposed algorithmis designed for the IoT environment must consume minimaltime and offer considerable security.

Table3. Simulation Execution time analysis

Size in KB	Execution time in seconds			
	DES		Proposed	
	Encryption	Decryption	Encryption	Decryption
1kb	0.03174 µs	0.02409 µs	0.01755 µs	0.02155 µs
2kb	0.06553 µs	0.04646 µs	0.03821 µs	0.04531 µs
3kb	0.09278 µs	0.06993 µs	0.05854 µs	0.06285 µs
4kb	0.15349 µs	0.13883 µs	0.13919 µs	0.12309 µs
5kb	0.25106 µs	0.11734 µs	0.17007 µs	0.13222 µs
6kb	0.20692 µs	0.37106 µs	0.10848 µs	0.1203 µs
7kb	0.22737 µs	0.16718 µs	0.16186 µs	0.14466 µs
8kb	0.29727 µs	0.1975 µs	0.15017 µs	0.1602 µs
9kb	0.3885 µs	0.419 µs	0.3396 µs	0.21626 µs
10kb	0.51769 µs	0.22768 µs	0.20074 µs	0.22584 µs
11kb	0.61275 µs	0.29221 µs	0.20505 µs	0.2321 µs
12kb	0.43877 µs	0.31564 µs	0.25016 µs	0.42116 µs
13kb	0.74024 µs	0.40837 µs	0.31113 µs	0.47666 µs
14kb	0.57657 µs	0.48971 µs	0.28846 µs	0.44239 µs
15kb	0.5557 µs	0.36846 µs	0.38238 µs	0.51531 µs
16kb	0.62726 µs	0.82712 µs	0.43352 µs	0.61591 µs
17kb	0.6557 µs	0.58424 µs	0.52655 µs	0.47965 µs
18kb	0.92108 µs	0.42427 µs	0.3411 µs	0.4537 µs
19kb	0.98009 µs	0.50696 µs	0.39018 µs	0.60169 µs
20kb	0.92426 µs	0.51126 µs	0.48906 µs	0.62268 µs
21kb	0.72864 µs	0.65158 µs	0.38049 µs	0.64932 µs
22kb	1.00189 µs	0.51178 µs	0.41083 µs	0.47379 µs
23kb	0.82872 µs	0.55041 µs	0.40814 µs	0.55816 µs
24kb	1.03691 µs	0.57684 µs	0.5619 µs	0.7565 µs
25kb	1.0916 µs	0.57624 µs	0.45512 µs	0.72021 µs
26kb	1.10969 µs	0.59566 µs	0.49943 µs	0.55834 µs
27kb	1.23345 µs	0.62056 µs	0.49599 µs	0.62417 µs
28kb	1.10219 µs	0.6362 µs	0.57352 µs	0.86583 µs
29kb	1.24727 µs	0.7159 µs	0.72167 µs	0.69988 µs
30kb	1.23211 µs	0.67718 µs	0.73301 µs	0.79049 µs
31kb	1.18384 µs	0.71916 µs	0.70869 µs	0.76343 µs
32kb	1.33618 µs	0.76203 µs	0.7814 µs	0.83589 µs
33kb	1.42411 µs	0.77853 µs	0.98596 µs	0.83514 µs
34kb	1.54723 µs	0.77563 µs	0.74604 µs	0.72639 µs
35kb	1.52819 µs	0.86422 µs	0.69782 µs	0.70533 µs
36kb	1.55052 µs	0.8603 µs	0.86441 µs	0.94902 µs
37kb	1.5211 µs	0.8793 µs	0.8862 µs	1.0228 µs
38kb	1.83828 µs	0.85912 µs	1.01302 µs	0.93109 µs
39kb	3.48261 µs	1.17661 µs	1.84727 µs	1.9179 µs
40kb	2.0692 µs	0.91521 µs	0.96106 µs	1.33887 µs
41kb	1.63982 µs	1.30434 µs	1.05304 µs	0.90455 µs
42kb	1.71927 µs	0.98342 µs	1.06949 µs	1.18973 µs
43kb	1.74491 µs	1.07152 µs	1.1184 µs	1.16339 µs
44kb	1.69389 µs	1.403 µs	1.16538 µs	1.07883 µs
45kb	1.8437 µs	1.14866 µs	1.21717 µs	0.97487 µs
46kb	1.9135 µs	1.15587 µs	1.19667 µs	1.05892 µs
47kb	1.88982 µs	1.149 µs	1.18878 µs	0.99888 µs
48kb	1.96608 µs	1.20581 µs	1.26077 µs	1.11909 µs
49kb	1.97174 µs	1.15027 µs	1.40752 µs	1.15778 µs
50kb	1.97742 µs	1.21187 µs	1.31672 µs	1.14579 µs

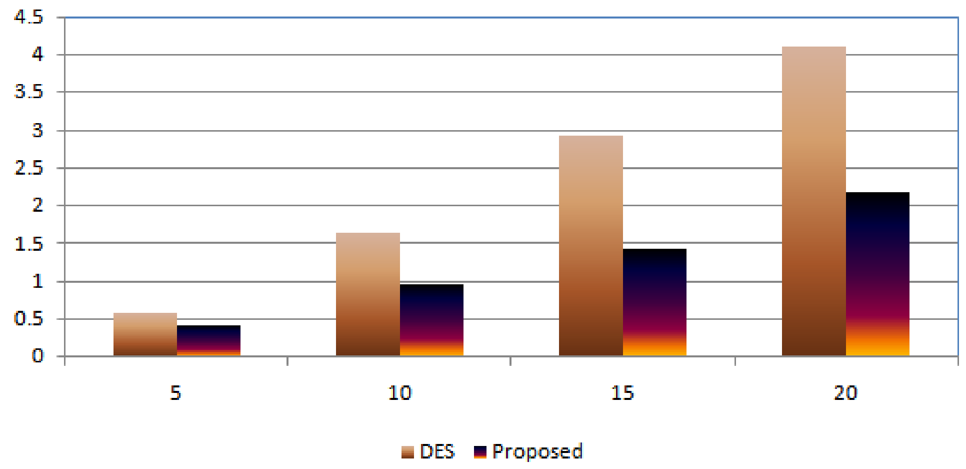

Figure4. Column of Execution time (Encryption)

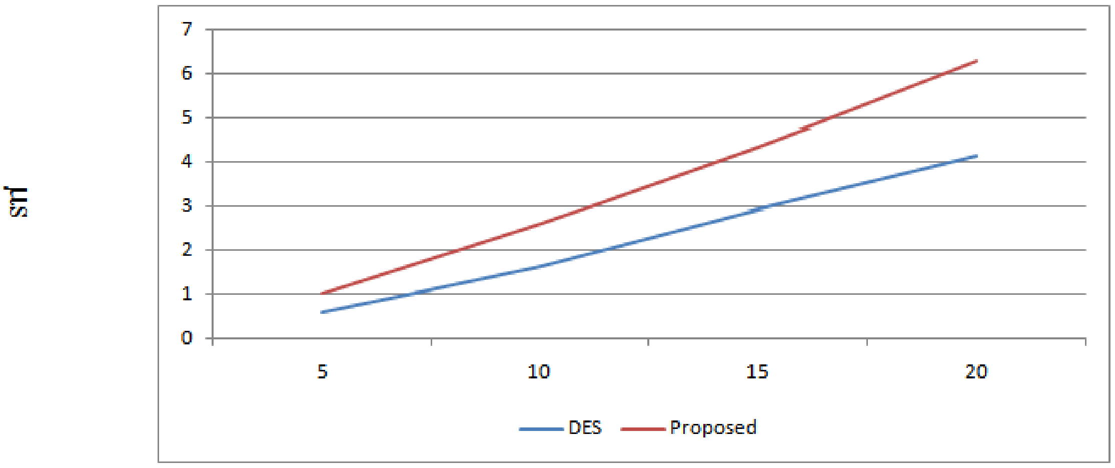

Figure5. Line of Execution time (Encryption)

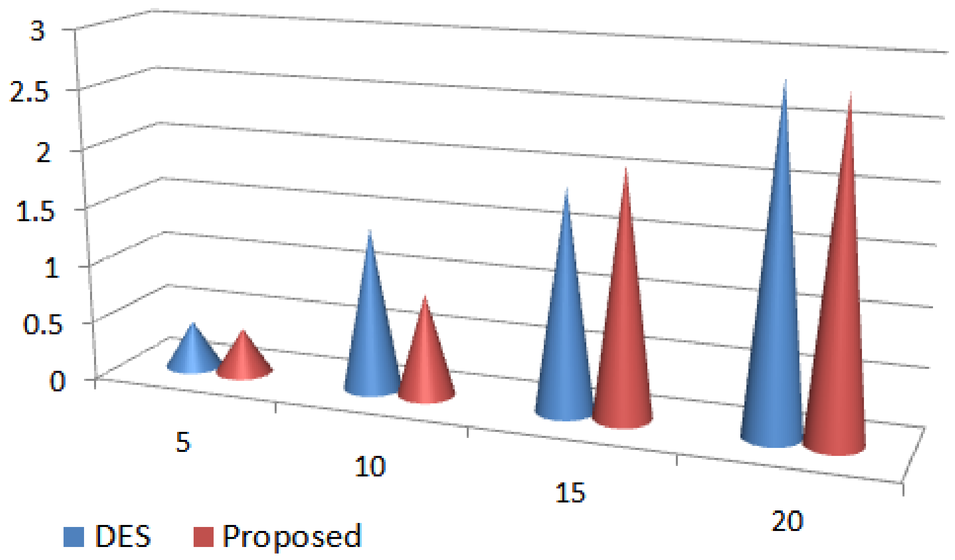

Figure6. Column of Execution time (Decryption)

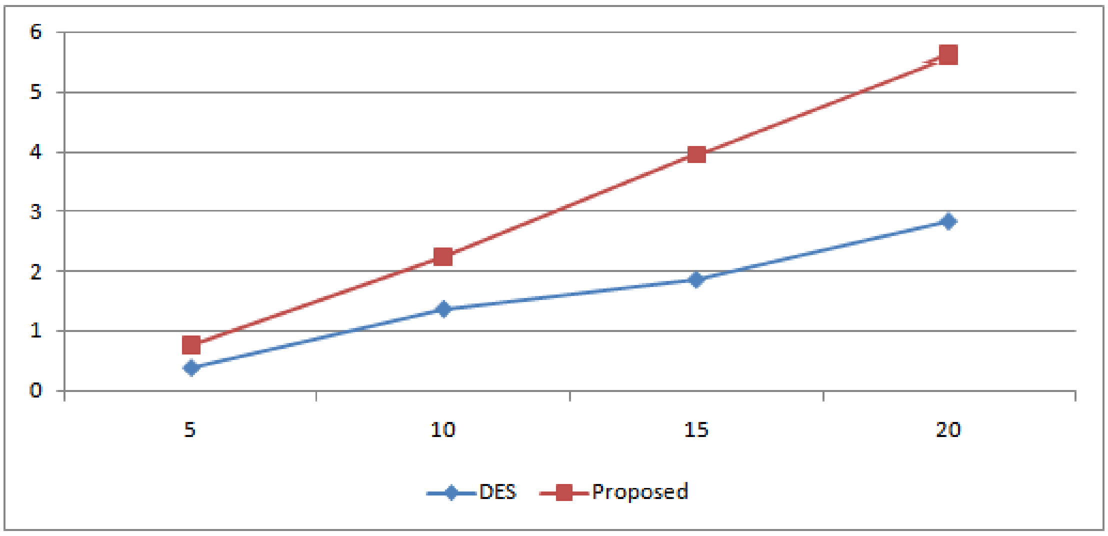

Figure7. Line of Execution time (Decryption)

2) *Avalanche effect:* the "avalanche" quantifies the effect on the cipher of the change of one bit in the text, for instance, the Strict Avalanche Criterion states that with the change of any one input bit, every output bit shall change with a probability of exactly ½.

Avalanche effect Formula:

$$\text{Avalanche effect} = \frac{\text{Number of flipped bits in cipher text}}{\text{Number of bits in the ciphertext}} \qquad (12)$$

Key used: standard key

Table4. Avalanche effect comparison

	Avalanche Effect(Proposed)			**Avalanche Effect(DES)**		
Plain Text	**Original Hex**	**Hex after one letter Modification**	**AE**	**Original Hex**	**Hex after one letter Modification**	**AE**
Proposed	0xc0f7e08aa7182bc9	0xafc459fafec5e7d6	83.3333	0x9a1ee7d180d92caa	0x8038c68e84b529ca	72.222222
Original	0xdd366f55417f2b65	0x7f2d914712b01bf5	72.2222	0xaa0e4b69f0520bf0	0xa813b777ebbc8640	77.777778
DES	0xff77d6784136a934	0x9d2e1b997aa46791	88.8889	0x586dbff69ef0e779f	0x2f286ff635e2d9af1	77.777778
Effect	0x43b23d42d251cd57	0x543261153c66c305	66.6667	0x77781cf74b766dec	0xdb88ebb76e65ec50	77.777778
Boolean	0x4638973b3c6409a9	0xac4fbaf5773d67dd	83.3333	0x69cc499ca73419a6	0x69e702c76a23c386	72.222222

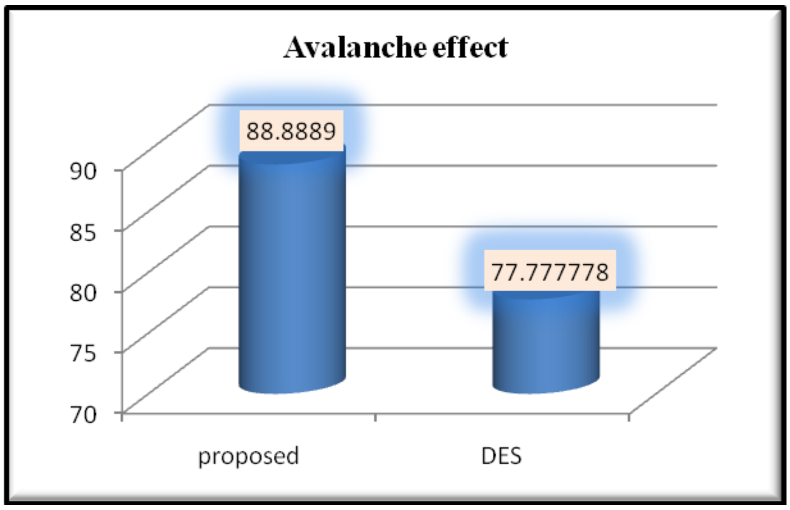

Figure8. Avalanche effect

Table5. Comparison of average proposed with DES

Algorithm	Execution time		Avalanche effect
	Encryption	Decryption	
Proposed	0.01755 µs	0.02155 µs	88.89
DES	0.03174 µs	0.02409 µs	77.7

IV. Conclusion and Remarks

On an average the parameters proving the best algorithm considered are execution timeand Avalanche effect. From the observations made it is clear that the proposed algorithm excels the performance of DES in both execution time and Avalanche effect to a maximum of 257 proving the suitability in preserving security and privacy in any IoT based application.

Author contributions

Nahla F. and Johnson I. have contributed todesign lightweight cryptographic algorithmfor resource constraint that are typically used in the IoT based application. Johnson I. contributed with reviewing the whole paper.

Funding

This research received no external funding.

Acknowledgments

The authors would like to express their thanks the infinite grace of the Almighty God is of essential importance and I solemnly offer my regards for His grace which enabled peace and harmony for this work.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1]. R. Chandramouli;S. Bapatla; K. Subbalakshmi; and R. Uma. Battery Power-Aware Encryption. ACM Transactions on Information and System Security (TISSEC) 2006, vol. 9, no. 2, (pp. 162–180).

[2]. S. Khan; M. S. Ibrahim; H. Amjad; K. A. Khan; and M. Ebrahim.FPGA Implementation of 64 bit secureForce Algorithm Using Full Loop UnrollArchitecture, IEEE International Conference on ControlSystem, Computing and Engineering (ICCSCE) 2006, (pp. 1–6).

[3]. S. Khan; M. S. Ibrahim; M. Ebrahim; and H. Amjad.FPGA Implementation of Secure Force (64-bit) Low Complexity Encryption Algorithm.International Journal of Computer Network and Information Security2015,vol. 7, no. 12, p. 60.

[4]. P. Barreto and V. Rijmen. The KHAZAD Legacy-Level Block Cipher. Primitive submitted to NESSIE 2000, vol. 97.

[5]. J. Daemen. Cipher and Hash Function Design Strategies Based on Linearand Differential Cryptanalysis," Ph.D. dissertation, Doctoral Dissertation,March 1995, KU Leuven, 1995.[Google Scholar

[6]. M. Ebrahim and C. W. Chong. Secure Force: A Low-ComplexityCryptographic Algorithm for Wireless Sensor Network (WSN).in ControlSystem, Computing and Engineering (ICCSCE) 2013, IEEE, (pp. 557–562).

[7]. Muhammad Usman;Irfan Ahmed; M. Imran Aslam; Shujaat Khan and Usman Ali Shah. SIT: A Lightweight Encryption Algorithm for Secure Internet of Things. (IJACSA) International Journal of Advanced Computer Science and Applications2017, Vol. 8, No. 1,(pp. 402-411)

[8]. B.VinayagaSundaram;Ramnath.M;Prasanth.M and VarshaSundaram.J. Encryption and Hash based Security in Internet of Things. 3rd International Conference on Signal Processing Communication and Networking (ICSCN) 2015, IEEE, (pp.1-6).

[9]. KiranKumar.V.G; SudeshJeevanMascarenhas;Sanath Kumar and VivenRakesh J Pais.Design And Implementation Of Tiny Encryption Algorithm.Int. Journal of Engineering Research and Applications2015, Vol. 5, Issue 6, (Part -2) (pp.94-97).

[10]. Deepikakhambra; PoonamDabas. Secure Data Transmission using AES in IoT. International Journal of Application or Innovation in Engineering & Management (IJAIEM)2017. Web Site: www.ijaiem.org Email: editor@ijaiem.org. Volume 6, Issue 6, June. ISSN 2319 - 4847. (pp. 283- 289).

[11]. GauravBansod;AbhijitPatil;SwapnilSutar and Narayan Pisharoty. ANU: an ultra-lightweight cipher design for security in IoT. SECURITY AND COMMUNICATION NETWORKS Security Comm. Networks 2016 Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1692.

[12]. Wentao Zhang; ZhenzhenBao;Dongdai Lin; Vincent Rijmen, Bohan Yang and Ingrid Verbauwhede. RECTANGLE: A Bit-slice Lightweight Block Cipher Suitable for Multiple Platforms, 2015, Vol. 58: 122103(15). (pp. 1-22).

[13]. KurniawanNurPrasetyo ST.; YudhaPurwanto, ST.; MT. and Denny Darlis;S.Si.; MT. AN IMPLEMENTATION OF DATA ENCRYPTION FOR INTERNET OF THINGS USING BLOWFISH ALGORITHM ON FPGA. 2nd (ICoICT) 2014, IEEE, (pp.75-79).