A Novel Simple and Highly Secure Method for Data Encryption-Decryption

Amjad Y. Hendi[1], Majed O. Dwairi[1], Ziad A. Al-Qadi[2], and Mohamed S. Soliman[3,4]

[1]Communication Technology Engineering department, Faculty of Engineering Technology, Al-**B**alqa Applied University, Jordan
[2]Computer &Network Engineering department, Faculty of Engineering Technology, Al-Balqa Applied University, Jordan
[3]Department of Electrical Engineering, Faculty of Energy Engineering, Aswan University, Aswan, Egypt
[4]Department of Electrical Engineering, Taif University, Taif, Kingdom of Saudi Arabia

Abstract: Data with deferent types are now transferring over the internet and the demands for clouding systems are now increasing Data generated, captured, and replicated are increasing in size and expanding applications. So the need for data encryption-decryption methods became a vital necessary issue. The used technique here must very simple, Effective, and accurate. A simple and highly secure encryption decryption (SHSED) algorithm that can be used for cloud computing-based applications, where introduced. It uses simple and efficient logical operations, such as XORing, addition, and subtraction in addition to byte shifting. The introduced, introduced method is also powered by the flexibility of selecting the secret key length and the number of rounds to generate the cypher text. Experimental results for the introduced method were compared with other methods results (LED, AES, DES), and as a result of comparisons the introduced method give a good improvement in encryption-decryption time measurement.

Keywords: AES, DES, LED, encryption time, decryption time, security level.

1. Introduction

Cloud technology is an important facility information capacity without any need of investment in creating new infrastructure or upgrading an existing one. This technology is now considered as a promising business concept to one of the steadfast ontogenesis circular segment of the IT laboriousness. Deferent data types in huge sizes are now transferring over the internet, which considered as unsecure environment, so security is one of the major deliveries which shorten the growth of cloud number and complications with data privacy and data safety continue to plague the market. [1,2]. Security is a needing task because of many reasons such as:

1. The data may be private.
2. The data may be personal.
3. The data may be confidential

Thus the security is becoming a must tool to prevent data from being understood by third party or unauthorized person [3].

The effectiveness and efficiency of traditional existing methods and technique using for data security and protection are being reconsidered, as the characteristics of this innovative deployment model, differ widely from those of traditional architectures [4].

In this research paper, the researchers attempt to demystify the unique security challenges introduced in a cloud and internet environments and clarify issues from a security perspective. The new light weight cryptosystem will be referred to as “Simple and Highly Secure Encryption-Decryption algorithm (SHSED) it is inspired by international data encryption algorithm (IDEA**)**. This introduced cryptosystem is anticipated to provide speed in execution time and powerful level of data security. Moreover, comparison will be conducted with other reported hybrid schemes.

Encryption has a special importance in information security science, which is the heart of information security because of its confidentiality. The use of encryption - through history – was to share messages that cannot be read by anyone other than the person intended to receive the message. Digital encryption technology has expanded beyond simple confidential messages; encryption can be used for more complex purposes, such as verifying the message author or surfing the Internet anonymously. Under certain circumstances, encryption can be automatic and simple. Encryption is the way to protect your private, personal and confidential data, such as documents, images, or electronic transactions on the Internet, from unauthorized people to prevent them from being seen or accessed or changed. Encryption uses a selected method, and a private key to convert readable data "plaintext" to a format that others cannot understand "encrypted text", or “cipher text”.

Cipher is a generic cryptographic recipe, and the encrypting key (secret key) makes encrypted data unique, only those who know this key can decrypt the encrypted massage. Keys are usually a long string of numbers protected by common authentication mechanisms such as passwords, symbols, or biometrics, like fingerprints or palm prints [5].

Today, encryption technology has a prominent place among science. Its practical applications have varied to include the diplomatic, military, security, commercial, economic, media, banking and informatics fields. It should be noted that the Arabs used the term "blindness" as a phase of converting clear text into an incomprehensible text using a specific method.

Hence encryption is the conversion of data from a readable state to a completely opaque state, like a utilitarian puzzle that does not add information to the reader.

There are four main objectives behind the use of cryptography:

1. Confidentiality or Privacy: Confidentiality is a service used to make the information unseen by unauthorized person.
2. Integrity: A service that is used to save information from changes, without being updated (such as delete, add or modify) by unauthorized persons.
3. Authentication: A service used to establish the identity of the customer with the data (authorized).

4. Non-repudiation: a service used to prevent a person from denying the reception of a message or denying the sending of a message, as demonstrated in [6, 7]. Therefore, in cryptography the use of mathematics to convert plaintext message (PT) into an unreadable cipher text (CT), during encryption phase at the sender side, while reconverting CT back to PT by decryption phase. Both phases use the same secret keys [6]. It must be stated the Key in the encryption phase must equal the key in the decryption phase in the case of symmetric systems while they are different in the case of asymmetric systems, as will be explained later on in this paper.

2. Related Work

Many lightweight symmetric encryption-decryption methods had been developed and reported for appropriate applications, such as LED. HIGHT, Block, DESL, CLEFIA, PRESENT, TWINE, RECTANGLE, SIT, etc.

In [8] a lightweight algorithm was proposed which operated with 64 bits block size text and 128 bits key, iterated in 32 rounds, and having two types of operations; This method used an XOR operation combined with left or right rotations. It was designed to suite hardware implementation on ubiquitous devices, such as wireless sensor nodes and RFID tags, having almost the same chip size as AES but works much faster. It was tested for security using differential attack giving results slightly less than the exhaustive search.

In [9] a CLEFIA-128 light weighted symmetric block cipher method was developed, this method was considered to be suitable for both hardware and software implementation, CLEFIA-128 has encrypting data block of 128 bits under 128 bits key length and 28 rounds with Festal structure, many versions of this method were reported with 192- or 256-bits key lengths running 22 and 26 rounds, respectively. CLEFIA implements 2 different S-boxes of 8 bits, followed by a diffusion matrix multiplication inspired from the AES Mix Columns operation. Using the impossible differential attack against CLEFIA reduced to 12 rounds for a 128 bits key with 2119 encryptions. Surely the execution time increases for longer keys.

In [3] two versions of lightweight Data Encryption-Decryption methods were developed, DESL and DESXL. DESL used a single S-Box instead of different ones with no initial and final states, on the other side, in DESXL, a whitening step is used to improve the security level by using a key of 184 bits length. No attack has been exhibited against DESL and DESXL as claimed by them.

In [4] an ultra-light block Encryption-Decryption method named "PRESENT' was proposed. It applies substitution-permutation network (SPN) structure with a block size of 64 bits and 80- or 128-bits key size running 31 rounds with multiple uses of 4 bits S-box. This system is also optimized for hardware implementation.

In [5] an efficient hardware-oriented block ciphers were proposed, it used a phase block size of 32, 48, or 64 bits with 80 bits key length. Two types were suggested, namely KATAN and KTANTAN differing only in the key scheduling, they use two logical functions with no shifting, running for 254 rounds in total. Both algorithms, KATAN and KTANTAN have a serialized structure.

In [6] a lightweight Encryption-Decryption method was proposed, it was called "LBLOCK". This method has SPN structure and efficiently implemented in both software and hardware. LBLOCK is designed with block size of 64 bits having a key of 80 bits and run for 32 rounds.

In [7] a lightweight block Encryption-Decryption method called LED was introduced. It presented reasonable performance efficiency for software implementation. This method has the ability to encrypt 64 bits blocks with different key sizes. They are 64 bits, 80 bits, 96 bits and 128 bits in length. The same S-box used for PRESENT cipher is used here in the execution of LED cipher system.

In [8] a lightweight Encryption-Decryption method which uses SPN technique was proposed. It presented reasonable performance efficiency for software and hardware implementation. This method implements 4 bits/16 S-boxes for substitution, and Rotate-4 bits and Mix-4 bits for permutation. It is called "KLEIN" ciphers with 64 bits block using key of different length, namely 64, 80, or 96 bits running for 12 or 16 or 20 rounds, respectively.

In [9] a generalized Festal structure with multi-platform Encryption-Decryption method was reported and it was called "TWINE". It is claimed that it has extremely-small hardware size, with efficient on embedded software. It is of 64 bits block size running 36 rounds with key of either 80 or 128 bits length, and each round involves a nonlinear substitution layer with 4-bits S-boxes and 4 bits block permutation layer In [4] a cryptosystem was reported and it was called "RECTANGLE" which designed for 64 bits block size with key length of either 80 or 128 bits, running only 25 round. It is an ultra-lightweight bit-slice block cipher, which is found suitable for multiple platforms achieving highly competitive software performance and requires very low area in hardware.

3. Introduced Method

The introduced method suggests the newly designed lightweight encryption algorithm (SHSED) that is based on (IDEA) cryptosystem. It is suggested to be used for data storage on cloud computing. Here we will include the design and implementation of SHSED, then its comparison with other cryptographic algorithms such as the widely used AES and DES algorithms that are used in cloud computing together with lightweight algorithms. Then examples are included, and results were output.

The proposed her in this paper method has the following advantages:

1. Variable length private key.
2. Variable length cypher text blocks.
3. Variable number of rounds.
4. Changeable logical function(S-function).

3.1 Modified Crypto System

The main idea of the SHSED cryptography is to use a 64-bit size block, 256-bit key length and 7-bit constants variables CST, besides the number of rounds ranges from 1 to 31 rounds. The procedure followed in this algorithm implements mixed operations of different algebraic groups, namely XOR and Addition operations. The algorithm accepts the plaintext data and the encryption key prior to produce the cipher text data. The algorithm uses two types of keys; work keys (WK) and sub-keys (SK).

In the following, the encryption phase, decryption phase, and the key generation phase will be outlined.

3.2 Encryption phase

The text to be encrypted is to be divided into equal blocks (each is referred to as M) of 64 bits length, as shown in figure 1. Each block to be encrypted is treated applying the following steps:

1. The input block, M is divided into 8 sub-blocks of 8 bits each, namely P1, P2, P3, P4, P5, P6, P7 and P8.
2. (Initial state steps): Each sub-block is treated by mixing operation from different algebraic groups they are XOR and Addition operations, using work key that are described later.WK1, WK3 Add with P1, P5. Then WK2, WK4 are XORed with P3, P7. As shown in Figure 2.

This step will be repeated also in the final state using different Work keys WK.

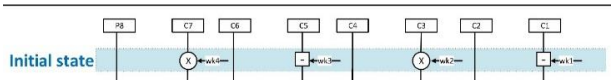

Figure 1. Structure of initial state SHSED encryption introduced

Figure 2.a show the block diagram for operations in round 1, in which the following steps are performed.

3. (round 1): The result of P1 becomes (X1,1), use this result again to have (X1,2) by treating it with F1 function, then Add result to sub key SK1 (also will be described later) then XOR it with P2.
4. The result of P3 becomes (X1,3), use this result again to get (X1,4) by treated with F0 function, then XOR the result with sub key SK2then Add the result with P4.
5. The steps 3 and 4 repeat as they are using the sub-keys SK3, SK4 to get (X1, 5), (X1, 6), (X1, 7) and (X1, 8).
   [It should be noted that in (X1, 1) to (X1,8), the first number represents the round number and the second parameter represents the byte number].
6. The output of the (round 1) are circular shifted to the right, as shown in Figure 2.b. Hence the value of (X1,1) becomes (X1,2), (X1,2) becomes (X1,3), (X1,3) becomes (X1,4), (X1,4) becomes (X1,5), (X1,5) becomes (X1,6), (X1,6) becomes (X1,7), (X1,7) becomes (X1,8). (X1, 8) becomes (X1, 1) for the next round. This step is the end of the round.

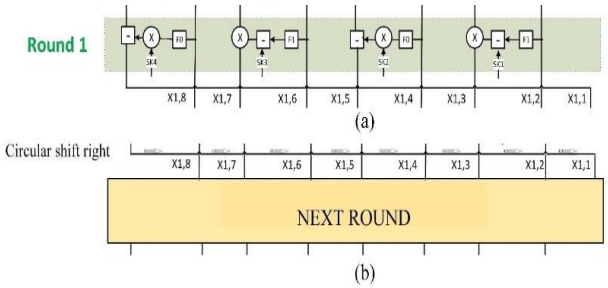

Figure 2. Block diagram for (a) Round 1 operation, (b) circular shift operation.

7. The same operations are performed for (n) of rounds, where n is considered as part of the secret key.
8. The result of the round (n) will be the initial values treated by the steps of initial state using keys (WK5, WK6, WK7 and WK8) this step called final state.
9. The final state output is to be considered as a cypher text.

All phases involved in implementing the introduced SHSED algorithm for n rounds are illustrated in the block diagram of Figure 3. Encryption phase contains more than one round, the number of rounds ranges from 1 to 31 rounds. After each round the value of (X i , j; where I is the round number and j is the byte number in the processed data block) will be circular shifted.

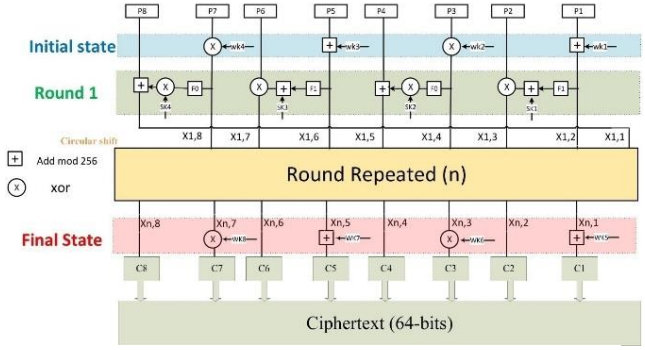

Figure 3. Structure of encryption phase for the introduced SHSED algorithm.

3.3 Decryption

The process of decryption can be implemented in an inverse way of the encryption phase.

4. Implementation

For the implementation and testing of SHSED, the personal computer (PC) is used for installation and execution, with the following technical features:

1. Operating system: Windows 10 home 64-bit.
2. Processor: A10-9600P RADEON R5, 10 COMPUTE CORES 4C+6G GHz.
3. Installed Memory (RAM): 6.00 GB

4.1 Performance analysis of cryptography algorithms

One of the most important goals of the performance analysis for various encryption-decryption methods is the computational cost of the processing time. Hence, the computational cost (execution time) for the introduced SHSED algorithm, together with the other algorithms under consideration, namely AES, DES, and LED are computed for both encryption and decryption and listed in the following sub-sections.

4.1.1 Computational cost for the introduced SHSED method

Table 1 lists out the computational cost or the execution time taken for encryption and decryption for the introduced SHSED algorithm. It included a number of experiments, and the average time is also determined. It must be stated that the same message is used for all the measurements involved.

Table 1. Execution time (seconds) for the introduced SHSED algorithm

Experiment #	Encryption time (Seconds)	Decryption time (Seconds)
1	0.0980	0.0820
2	0.0950	0.0880
3	0.1020	0.0850
4	0.0990	0.0840
5	0.1000	0.0840
6	0.1050	0.0870
7	0.1010	0.0690
8	0.0970	0.0850
9	0.1010	0.0850
10	0.0930	0.0940
Average	0.0991	0.0843

The fluctuations noticed in execution time can be attributed to the time complexity of the operation in the CPU; however, the average time is more meaningful and reflects the efficiency of the considered cryptographic system.

4.1.2 Computational cost for AES algorithm

Table 2 lists out the computational cost or the execution time taken for encryption and decryption for AES algorithm. It must be stated that the same message is used for all the measurements involved.

Table 2. Execution time(seconds) for AES algorithm

Experiment #	AES	AES
1	0.4260	0.5920
2	0.4180	0.5030
3	0.4420	0.5450
4	0.5220	0.4670
5	0.3940	0.4520
6	0.3720	0.5370
7	0.3470	0.6000
8	0.5830	0.5570
9	0.4780	0.5320
10	0.4680	0.6230
Average	0.4450	0.5408
Speed up of SHSED	0.4450/0.0991=4.4904	0.5408/0.0843= 6.4151

4.1.3 Computational cost for DES algorithm

Table 3 lists out the computational cost or the execution time taken for encryption and decryption for DES algorithm.

Table 3. Execution time (seconds) for DES algorithm

Experiment #	DES	DES
1	0.0690	0.0670
2	0.0570	0.0620
3	0.0590	0.0690
4	0.0560	0.0830
5	0.0480	0.0690
6	0.0380	0.0570
7	0.0510	0.0430
8	0.0490	0.0660
9	0.0610	0.0670
10	0.0460	0.0590
Average	0.0534	0.0642
Speed up of SHSE	0.0534/0.0991=0.5388	0.0642/0.0843=0.7615

4.1.4 Computational cost for LED algorithm

Table 4 lists out the computational cost or the execution time taken for encryption and decryption for LED algorithm.

Table 4. Execution time (seconds) for LED algorithm

Experiment #	LED	LED
1	0.1240	0.1980
2	0.1210	0.1830
3	0.1230	0.1880
4	0.1230	0.1910
5	0.1210	0.1810
6	0.1230	0.1900
7	0.1200	0.1820
8	0.1190	0.1850
9	0.1230	0.1910
10	0.1230	0.1910
Average	0.1220	0.1880
Speed up of SHSED	0.1220/0.0991= 1.2311	0.1880/0.0843= 2.2301

It is noted that a significant improvement in the efficiency of the introduced SHSED algorithm as compared with that for light weight algorithms (LED). The speed up gain in the encryption phase is 1.2311 times, while the speed up in the decryption phase is 2.2301 time.

4.2. Analysis and Comparison of SHSED with other algorithms

The Introduced (SHSED) algorithm achieved lower computational cost in both cases of encryption and decryption when compared with the corresponding computational cost for the other cryptographic system, namely AES, and DES algorithms. Figures 4, and 5 illustrate the encryption time for these algorithms, and figure 6 illustrates the decryption time for the same algorithms under consideration. These figures listed the results for ten runs using the same message block.

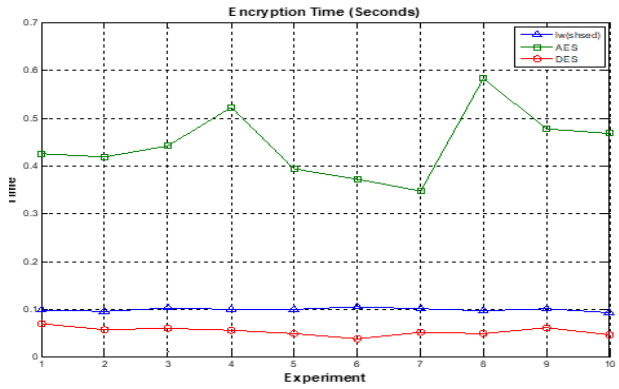

Figure 4. Encryption time comparison of SHSED with AES and DES

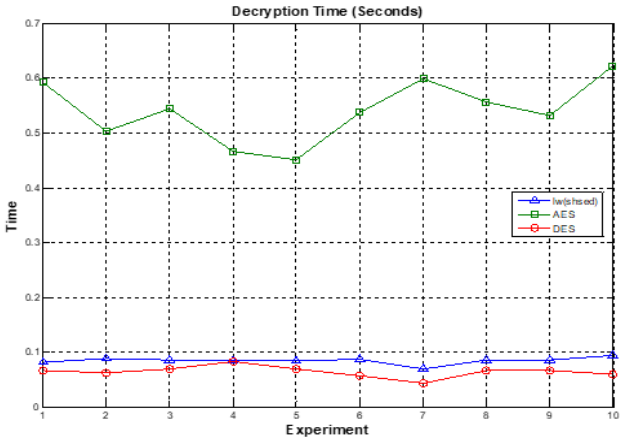

Figure 5. Decryption time comparison of SHSED with AES and DES

It is quite clear that the execution time for SHSED algorithm is much shorter than that for the AES algorithm, but it is slightly longer than those for DES algorithm. This is true for both cases of encryption and decryption phases.

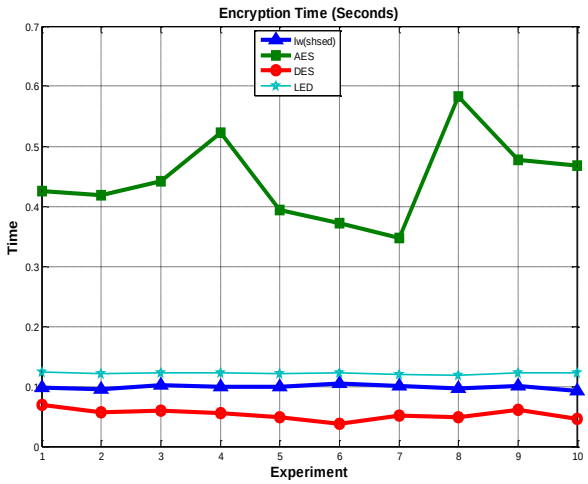

Figure 6. Decryption time comparison of SHSED with AES, DES and LED

When the speed up gains for SHSED algorithm with respect to AES, DES and LED are calculated, they will be as listed in table 5. It is shown for both encryption and decryption phases.

Table 5. Speed gain comparison for SHSED with respect to AES, DES, and LED

	Speed up	
	Encryption	Decryption
AES	4.4904	6.4151
DES	0.5388	0.7615
LED	1.2310	2.2301

4.3. Result analysis of rounds for the Introduced SHSED

The number of rounds for the introduced SHSED algorithm is investigated and the results are listed in Table 6, and they are plotted in Figure 7.

Table 6. rounds computational time for SHSED algorithm

Rounds	Encryption time(Seconds)	Decryption time(seconds)
2	0.0710	0.0258
3	0.0706	0.0265
4	0.0778	0.0277
5	0.0730	0.0316
6	0.0747	0.0328
7	0.0715	0.0288
8	0.0732	0.0307
16	0.0762	0.0362
20	0.0787	0.0385
24	0.0793	0.0453
30	0.0820	0.0441

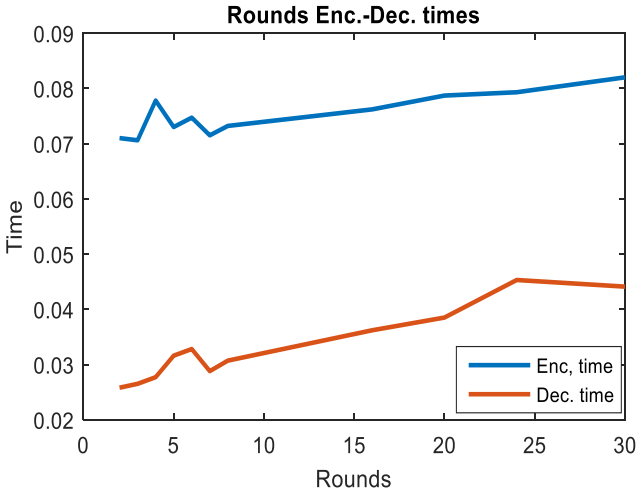

Figure 7. Effect of number on encryption and decryption time for SHSED algorithm

4.4. Measurement SHSED algorithm security strength

The security strength of the SHSED algorithm was measured using an avalanche effect equation:

(number of flipped bits in the ciphertext / number of bits in the ciphertext) ×100%

Table 7 lists out result of avalanche effect with fixed plaintext **(4142434445464748)** and varied key, in the rounds=3.

Table 7. avalanche effect with fixed plaintext

test	key	Ciphertext (Hex)	Number of changed bits	Avalanche effects %
1	ffeeddccbbaa99887766554433221100	89E053EB67653FBF	21	48
2	ff00ddccbbaa9912776655ac33ff1100	9BE0B5CA01FA11F4	34	53
3	1500d7ccb4ac991972665bad33ff11f0	2616B6BC17705499	33	52
4	ab09d7ccafac991c72365bad33af110f	2616B6BC17705499	33	52
5	ac02d9ccb5ac961c72385ba9330f1acf	D2F2AC8BA4BCE3A6	39	61
6	a4c279ccbfac991c72b85ba9330f1a78	1C267EC2D3FDCC96	37	58
7	f4cd795cbf2c9a1c62b84bb9cc0f1a78	62916358451E7A97	27	42
8	f4cd795cbf2cccab62b84bb9cc0f1a21	62916358451E7A97	27	42
9	24cd795cbfccccab6ab84bb95c0f1a2c	F6DDF021A288924B	39	61
10	24cd795cefccccbb6ab84bb95c0f1a7d	8D1717DC8A9808AC	35	55

Table 8 lists out the result of avalanche effect with fixed key (**ffeeddccbbaa99887766554433221100**) and varied plaintext, in the rounds=3

Table 8. avalanche effect with fixed key

test	**Plaintext (Hex)**	**Ciphertext (Hex)**	**Number of changed bits**	**Avalanche effects %**
1	4142434445464748	89E053EB67653FBF	21	48
2	4942434446564748	D2F16D97BC390661	30	47
3	9232554446 5647005	B67B478C4A5937AE	24	38
4	a232c54446 5646261	C3E38A1521F4993E	32	50
5	b232c244d6 5548241	4CB2AC35015C38F4	31	48
6	a53dc254d6 5544221	65F652337740B713	29	45
7	f53cc258d6 5944201	DB0985B0CC897CEE	31	48
8	f63cc258d6 a944241	A626DB7EC5865031	31	48
9	af3cc2b8d6 b9c42f1	D7C33861C7E4CCBB	35	55
10	a93bb2b8d6 b9c42f	D7C3B63AD8002CAB	28	44

Table 9 lists out the result of avalanche effect with fixed key (**ffeeddccbbaa99887766554433221100**) and fixed plaintext, with varied rounds.

Table 9. avalanche effect with fixed plaintext and key

Number of Rounds	Plaintext (Hex)	Ciphertext (Hex)	Number of changed bits	Avalanche effects %
1	4142434445464748	649AA5796C1474E0	31	48
2	4142434445464748	BF96D38EC9657464	30	47
3	4142434445464748	89E053EB67653FBF	29	45
4	4142434445464748	7D607B893C2E3A89	31	48
8	4142434445464748	715F900782E15A54	31	48
16	4142434445464748	CEF255948C174847	31	48
20	4142434445464748	C987B42AE95F22D4	33	52
24	4142434445464748	98CBCBE09AA01F34	33	52
28	4142434445464748	A98FCAC7CE46A4D9	27	42
31	4142434445464748	F836F069128A7884	37	58

From the obtained results shown in the previous table we can see that a small change in the key or the plain text or in the number of rounds should be resulted with a significant change in the cipher text, thus indicate the strength of the introduced algorithm.

5. Conclusions

A simple light weight and highly secure encryption_decryption (SHSED) method was introduced and it can be applicable for various data processing applications including Cloud based applications.

Experimental results have demonstrated powerful security level and a clear improvement in the encryption/decryption execution times compared with other cryptographic systems widely used in cloud computing.

The speed up obtained by the proposed (SHSED) method compared with AES and the lightweight LED algorithms are encouraging for a practical application as the efficiency was 4.4 times for encryption and 6.41 times for decryption in the case of AES algorithm. It is also 1.23 times faster than LED for encryption and 2.23 times for decryption in the case of LED. However, it was slightly slower than DES.

References

[1] ziad alqadi, “Analysis of stream cipher security algorithm” Journal of Information and Computing Science, Vol. 2, No. 4, pp 288-298.2007.

[2] Ziad A Alqadi, Majed Omar Al-Dwairi, Hatim Zaini, Mohamed S. Soliman “ZJICD algorithm for JPEG image compression/decompression,” Elixir International Journal, Elixir Digital Processing 94, pp 40368-40374. 2016.

[3] Randeep Kaur, SupriyaKinger, “Analysis of Security Algorithms in Cloud Computing,” International Journal of Application or Innovation in Engineering & Management (IJAIEM), Vol. 3, Issue 3, pp 171-176, 2014.

[4] Jean Raphael NgnieSighom, Pin Zhang and Lin You, "Security Enhancement for Data Migration in the Cloud," Future Internet, P 1-13, 2017.
[5] Vinita Keer, Dr. Syed Imran Ali, Prof. Neeraj Sharma, "Hybrid Approach of Cryptographic Algorithms in Cloud Computing," International Journal of Emerging Technology and Advanced Engineering, Vol. 6, Issue 7, pp 87-90, 2016.
[6] Akashdeep Bhardwaj, GVB Subrahmanyam, Vinay Avasthi, HanumatSastry, "Security Algorithms for Cloud Computing." Procedia Computer Science, pp 535-542, 2016.
[7] Felicisimo V. Wenceslao, Jr, "Enhancing the Performance of the Advanced Encryption Standard (AES) Algorithm Using Multiple Substitution Boxes," International Journal of Communication Networks and Information Security (IJCNIS), Vol. 10, No. 3, pp 496-501.December 2018.
[8] Hong D., J. Sung, S. Hong, J. Lim, S. Lee, B. Koo, C. Lee, D. Chang, J. Lee, K. Jeong, H. Kim, J. Kim, and S. Chee, "HIGHT: A New Block Cipher Suitable for Low-Resource Device. In Cryptographic Hardware and Embedded Systems - CHES, LNCS," 4249, Springer-Verlag, pp. 46-59,2006.
[9] Shirai T., K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, "The 128-bit block ciphercle_a (extended abstract). In Fast Software Encryption – FSE", vol. 4593 of Lecture Notes in Computer Science, Springer-Verlag, pp. 181-195,2007.
[10] Leander G., C. Paar, A. Poschmann, and K. Schramm, "New lightweight DES variants. In the proceedings of Fast Software Encryption – FSE", Lecture Notes in Computer Science, Springer-Verlag, vol. 4593, pp. 196-210,2007.
[11] Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J. ... &Vikkelsoe, C. Present: "An ultra-lightweight block cipher. International Workshop on Cryptographic Hardware and Embedded Systems", Springer, Berlin, Heidelberg, pp. 450-466September, 2007.
[12] Gholami, A., Laure, E. "Security and privacy of sensitive data in cloud computing", a survey of recent developments. arXiv preprint arXiv:1601.01498,2016.
[13] Wenling Wu and Lei Zhang, Lblock, "A lightweight block cipher". In Javier Lopez and Gene Tsudik, editors, Applied Cryptography and Network Security, vol. 6715 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp 327–344,2011.
[14] Guo J., T. Peyrin, A. Poschmann, and M. J. B. Robshaw, "The LED blocks cipher", In Bart Preened and Tsuyoshi Takagi, editors, Cryptographic Hardware and Embedded Systems CHES, vol. 6917 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp. 326–341,2011.
[15] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matt Robshaw, "The LED Block Cipher", L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," ACM SIGCOMM Computer Communication Review, vol. 39, no. 1, pp. 50-55, 2011.
[16] Suzaki T., K. Minematsu, S. Morioka, and E. Kobayashi, TWINE: "A lightweight block cipher for multiple platforms". In LarsR. Knudsen and HuapengWu, editors, Selected Areas in Cryptography, vol. 7707 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, pp 339–354, 2013.