Advances and Applications in Mathematical Sciences
Volume 17, Issue 1, November 2017, Pages 31-50

IMPLEMENTATION OF MICRO AVALANCHE EFFECT - DES OPERATIONS

K. RAJA SEKHAR, A. NAGESHWAR REDDY, K. V. ASHISH PRASAD, RATNA SRI, G. S. S. VINEETH and J. AMUDHAVEL

K L University, India
E-mails: rajasekhar_cse@kluniversity.in
nageshwarreddy1234@gmail.com
ash21411@gmail.com
mrs7.sweety@gmail.com
gssvineeth@gmail.com
info.amudhavel@gmail.com

Abstract

Avalanche effect is the number of cipher text bits changing with respect to bit by bit in plain text and key values. The advantage of Avalanche effect can be made applicable in securing embedded applications wherever DES and AES algorithms are used. The attackers are trying a lot to smuggle the data stored in the databases. Many algorithms are prevailed to protect the communication channel. Most of the existed algorithms procure the secured key for encryption and decryption. The theme of the paper is to augment the security in the communication by observing the performance of the Avalanche Effect over multiple operations executed on DES cipher.

1. Introduction

The binary data which is sent through a channel has to be more secured and should not get altered by any cryptanalytic attacks and many principles are added to the transmission channel before and after the data transmission. All the security principles such as authentication, data integrity, privacy, and confidentiality have to be considered. If they can't be achieved, the application's data is not safeguarded. [1] Cryptography yields a strategy for providing security and authentication over a communication channel in order

2010 Mathematics Subject Classification: 68N15, 97N50.
Keywords: Cipher-text (Encoded message), Plain-text (Decoded message), Avalanche Effect, Data Encryption Standard (DES), Communication channel.
Received March 10, 2017; Accepted July 20, 2017

to prevent the unauthorized access to the other users' accounts. It also allows us to communicate over insecure channels without effecting to lose our data and ceases unusual entry. In cryptography, the data security is achieved through scrambling the data by applying substitution or transposition operations. The digital data can be in the form of image, text, video and audio. With the cryptography techniques, any form of data is encoded called the cipher. The data before encoding called the plaintext that is easily understandable is altered by applying different substitution and transposition operations to make it to a cipher text. This strategy of transformation is known as the encryption and the vice versa called the decryption. The strength of the encryption depends on the algorithm that is procured and the core of it is the key by which the security of the data is boosted. [2] A bunch of algorithms are available, but the key principle is to deliver the message by encoding with the secured key and likewise for decoding too. The proposed method enhances the security by increasing the avalanche effect through significant changes in Data Encryption Standard (DES). [3]

The analysis of DES can be done in following ways:

1. Avalanche effect: The minute modification in plaintext effects the cipher text to a large extent.

2. Completeness: Every bit of cipher text will be dependent on most of the bits of plaintext.

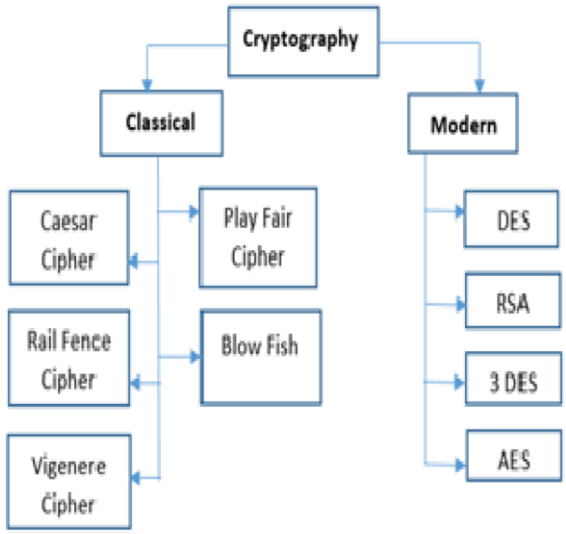

Figure 1. Classification of Cryptography.

This paper is organized as: Module II and III give the related work required for this proposed work. Module IV gives proposed work's methodology. Module V gives the results of it. Module VI is about concluding the paper with future work.

2. Secured Algorithmic Types

The contrast of algorithms is based on the number of keys used to transmit the binary data. The algorithms are categorized as Symmetric and Asymmetric. The keys are either identical or a slight transformation between them. The two or more parties share a secret key which establishes a private communication among them. To encrypt and decrypt the message, identical key is used, named Symmetric key algorithm and the examples of it are DES, AES and the converse of it is asymmetric algorithm which uses pair of keys and the instances of it are RSA, Merkle's Puzzles, ElGamal. [4] The public keys are disseminated but the private keys are known only to the trusted persons. It satisfies authentication. Whenever the message is sent to the receiver using the secret key by the sender, firstly the sender has to be authenticated and then the message has to be decrypted by the private key to attain security. The symmetric cryptographic analysis will mainly have the following components:

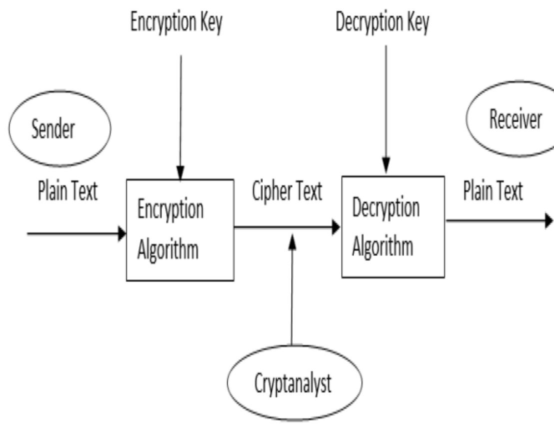

Figure 2. Elements of cryptographic analysis.

2.1. Components of Cryptographic System

2.1.1. Plain Text or Clear Text

It is the actual sender message which is the input to the algorithm.

2.1.2. Encryption (Algorithm)

With the help of this algorithm, data is encoded called cipher text and this encoding process is secured using the key that is used.

2.1.3. Decryption (Algorithm)

This is required to transform the incoherent cipher text into plain text.

2.1.4. Encryption Key

The secret key is supplied to encryption algorithm to provide security to the plain text.

2.1.5. Decryption Key

The secret key is provided to the decryption algorithm in order to decode the ciphertext.

2.2 People involved in Communication

2.2.1. Sender

The person who sends the original message or plain text to the communicating party or receiver.

2.2.2. Receiver

The person who receives the message from the sender in the secure manner.

2.2.3. Crypt-analyst

The person who intrudes into the communication channel and tries to capture the sensitive data.

3. Existing Procedure

3.1. Data Encryption Standard (DES)

The well known block cipher of DES called the Feistel block cipher was first designed by the cryptographic researcher named Hort Feistel in IBM. It contains many rounds where each one has substitution boxes, bit-shuffling and the XOR operations. In the present-day, many of the symmentric data encryption strategies are built on this feistel system to transpose the bits. The inputs to the DES algorithm are plaintext and the secret key. In order to interpret the kind of cipher, firstly observe whether the input message is received or not and secondly, observe the key that is utilized [5]. Hence DES is symmentric where the 64 bit block cipher uses the identical key for encrypting and decrypting. At a time, the DES could operate a 64 bit blocks of data and the size of a key is a 56 bit, but in the proposed method the input size of a key is a 64 bit. The last bit of individual byte is utilized for parity as it doesn't boost the security. [11] These blocks of data are computed in an anti-clockwise manner where the eight bit of a byte turns into a parity bit. When a plaintext is sent as input to the DES algorithm, it is gathered to a 64 bit. If these bits are indivisible by 64, then the end block is padded. The various substitutions and permutations are combined to enhance the security.

3.2. General Design

Initially DES contains permutation for an input of 64 bit. Later it is divided into a two equal blocks of a 32 bit. They are left round (Li) and right round (Ri). These blocks of data are proceeded to the 16 number of rounds. All the rounds are alike but the results are differed. So at the last 16^{th} round of DES, the resultant of a Li (32 bit) and Ri (32-bit) are interchanged in order to notice the pre-outcome. Now the 16th rounds i.e. (R16, L16) are permutated with a function i.e.; the reverse of initial permutation. Hence the 64 bit ciphertext are obtained with the final permutation. [6]

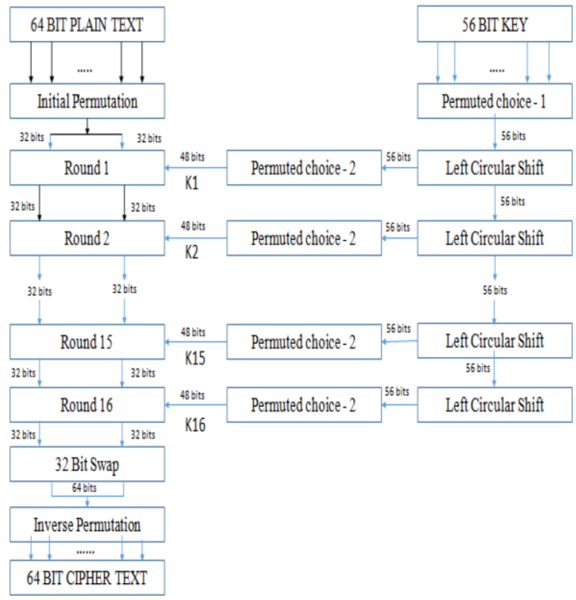

Figure 3. DES algorithm to encrypt the data.

4. Proposed Module

In the proposed module, the applied operations for the original DES are permutations to the shifts of DES. [7] There are two shifts namely single and double. On application of permutation tables, these have been complicated in order to modify the original DES to raise the [12] Avalanche effect.

Therefore more security can be achieved as follows:

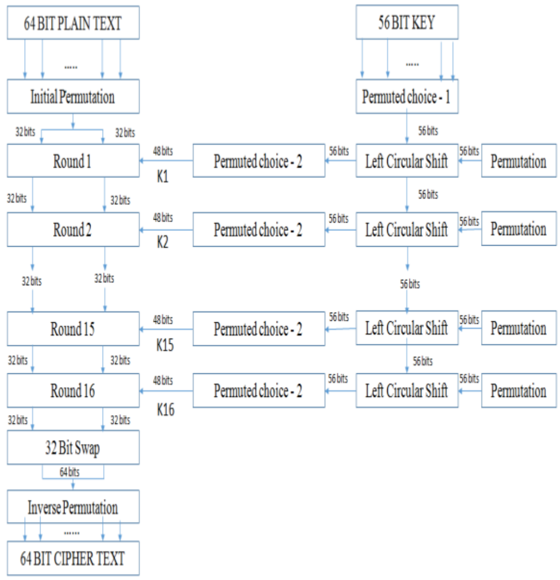

Figure 4. Encryption of modified DES.

The figure 4 is depicting the increasing nature of Avalanche effect for more inputs when compared to the original DES based on the permutations that are applied for each and every shift. [8] The main phenomenon is that the permutation table is to be altered before the communication in order to have much Avalanche effect as it is very much necessary to achieve the strength in security. [9] The modified system increases the security by raising the Avalanche effect. Based on the alteration of permutations that are applied to the shift operations before communication, the Avalanche effect can be boosted up to achieve more security. [10]

5. Observations and Results

5.1. Original DES

Plain Text: 4F4F4F4F4F4F4F4F

Key: 4F4F4F4F4F4F4F4F

Cipher Text: 6C69E720F5105518

Table 1. Original DES.

Bit Changed	Cipher Text	Avalanche Effect
1	DB0558B50D244970	35
2	ED04B0694472D31D	27
3	1494062EF1ED438F	34
4	2FE3526CCC1F1CAF	31
5	6D4B566EDB38E936	26
6	DEBD3791F1B7A4BC	29
7	77503AFDFC1058EF	32
8	DC42447EA6252349	32
9	2914BED69C295688	31
10	E96FAF6F2022F8FB	30
11	33F72F605428240C	27
12	0FA2A5B426DB3385	33
13	1F1772B6627F8F47	41
14	9645ABA16345E19B	29
15	BDD0817D7A07C7EA	35
16	F94F2FD2CA3B8227	37
17	A4D80C6B7DA11627	32
18	8BCD8EB1E16DDB32	31
19	1A658C0C691A8854	30
20	9AB869D0DD194A2D	31
21	F7A28D201E8F3A28	34
22	8EC9F35E9EA5FA9F	34
23	A6BE32C13D4C9535	32
24	128E67C1423E3694	34
25	B23CD7023F9B88C6	34
26	4831A03F05227C0F	28
27	7906D6ADCB2E51A2	32
28	224F6087B18DD6B3	31

29	2BA6EE317D7F6AA3	34
30	2C58D8A020AD94D8	27
31	2210FE1151D206AA	29
32	2ABAE9A474662D74	28
33	42446CBDA5B831EA	30
34	26E3F870C9D2B905	29
35	1F207329AF1BE1A6	30
36	1910CD7644C032CE	34
37	1AF4878C4BC462AC	35
38	0A8D47EB53DE578D	29
39	C0AC32A49922346C	29
40	722F837CEFACCA2D	32
41	468ECE813BE7D30F	34
42	7236BB3088C2E13B	32
43	F1A891B122E05D71	31
44	7174A0780AE0DB85	36
45	103071C9F7E33ECA	34
46	4A27BB809040F827	30
47	57C29486E1DD2D83	35
48	4FCDFAE56E3FB25B	33
49	040338C45C584CD7	33
50	D50E718AE98D23C7	38
51	D79DBEB8BFCDCF8B	35
52	D7952CA7D01EBCC5	38
53	7EF621C4092C0E2C	34
54	0492FFDC9AA181F1	37
55	D797E6F541B98713	34
56	9E01BE09258C353F	28
57	5BB0FEFF193890F3	37

58	677635459957721A	29
59	448FC617E2537E2E	29
60	6D4A3B42EFDC9269	28
61	F2DD8075421D7F7D	34
62	CE27C8B95EBF7FB4	34
63	D9F94490F954C98F	27
64	5F77EB910CC78C79	34

5.2. Modified DES (Proposed System)

Plain Text: 4F4F4F4F4F4F4F4F

Key: 4F4F4F4F4F4F4F4F

Cipher Text: 82604D0426645371

Table 2. Modified DES.

Bit Changed	Cipher Text	Avalanche Effect
1	58BEDE1EA042B3D9	30
2	C37CE3CDB4ED48B6	29
3	3887850670145BE1	25
4	D5DDC93B411D06F2	36
5	1E7CD7F4727E6375	24
6	5012266EDDA7BDB6	39
7	D92E602B9A5EE68F	39
8	CFFB8932295A265D	33
9	72418681AB411E08	30
10	A74BECF885F77593	31
11	63AFA896D1412220	35
12	D8CC7359B0AA047B	34
13	D7AE661DF0794435	31
14	108F14D49EE4AE1C	34
15	0B26BE6BD417A2C9	37
16	48C521B325B92A18	35

17	1D21273E59F6638B	34
18	B1903A5C2A8AE6D1	32
19	DA072EC08D81B039	32
20	77F2E173482F91D5	34
21	E0DB7087A9124781	33
22	922207BB85E1D294	27
23	A90B2E7002DA7257	30
24	ABE5859B955D4E4F	33
25	D2EF98231A92B21C	35
26	D630F5CAA2694CF3	26
27	267AB4E2AB052223	31
28	216C7E1855D717EC	30
29	9B4CF7C73CAC3B47	28
30	53A391FF622E0C88	37
31	C310E5F9E92A56F2	30
32	C9BE333AFC036FCB	40
33	3F9E769470CE1EA9	36
34	C11CB60AF7A5D944	32
35	06A02D89727571E9	20
36	1372A9A3568C6F1F	30
37	E6133457FEE1EEFF	34
38	7CB6630D3097A1DC	37
39	C6516A346550F4BA	27
40	CC1B66D27165B9CB	35
41	FF4672A7C5BFABA2	40
42	2E4F7B7A1B25A5C0	36
43	046566A8E9EBC06D	31
44	D3EA2F9759DA1A60	31
45	E8C6CDC53205ACFC	29

46	0ADE00F74F899110	34
47	03B842EB31B3D26F	33
48	36304A3A6B6AA62E	33
49	485C680352A22D47	32
50	650456C0F7D71A12	32
51	8804BEF00F86A043	32
52	6C774A349AF3E6FE	35
53	6386E8FB1A0FB0B6	40
54	8729433DFEA973B3	25
55	9012A94F8B9BA5B7	37
56	F631038B9DF799F4	33
57	09961D945AACA0FC	32
58	8C8823B830CC659D	32
59	3BFD8612AEE2795C	30
60	5AE6706AEC4827DF	33
61	320C0572C7A9C1F4	29
62	826AF57609EAD267	24
63	7E3951E3C46C6669	30
64	DD2B8C7E412F0E24	36

5.3. Original DES Roundwise Analysis (left Round)

Plain Text. 4F4F4F4F4F4F4F4F

Key. 4F4F4F4F4F4F4F4F

Table 3. Left Round Analysis.

Round No	Li (Left Round)
L1	00000000000000001111111111111111
L2	10101001000111001011010100100010
L3	10000101001000110101000001111010
L4	01111011101101101010010010111000

L5	11101001110000100011000101100110
L6	01100001110110100010101010100110
L7	11001011101010110001100011010101
L8	11111111110110110110111111100011
L9	10100000010010100010001110011100
L10	11011010011100110011100011000011
L11	01110111100111101010111101101010
L12	11010101111100101010011010111101
L13	10111000101110110101000110000010
L14	11010000100100101110100110000100
L15	01010111100111011000001110001111
L16	00010100000111111000001100000100

Table 4. Right Round Analysis.

Round No	Ri (Right Round)
R1	10101001000111001011010100100010
R2	10000101001000110101000001111010
R3	01111011101101101010010010111000
R4	11101001110000100011000101100110
R5	01100001110110100010101010100110
R6	11001011101010110001100011010101
R7	11111111110110110110111111100011
R8	10100000010010100010001110011100
R9	11011010011100110011100011000011
R10	01110111100111101010111101101010
R11	11010101111100101010011010111101

R12	10111000101110110101000110000010
R13	11010000100100101110100110000100
R14	01010111100111011000001110001111
R15	00010100000111111000001100000100
R16	01010111111100000101010101010110

5.4. Change in Permutation bit by bit analysis

Table 5. Left Round (bit changed = 16).

Round No	Li (Left Round)
L1	00000000000000001111111111111111
L2	10101001000111001011010100100000
L3	10001101001000010101100001111010
L4	10101011011110100111010000110010
L5	01100010001001001101001110100110
L6	00001101000111110101110111010001
L7	11010000011101101001001101001101
L8	10100100000100000001111110010110
L9	11010111101110011001010010011100
L10	00110010011110010110101100111111
L11	10101100101101011110100100111110
L12	0000110110100010000110001010110
L13	01000011110011001110011000101001
L14	00001101101011101000110011111000
L15	01110000101011100101101111000010
L16	01011001101001010011011111111110

Table 6. Right Round Analysis (bit changed = 16).

Round No	Ri (Right Round)
R1	10101001000111001011010100100000
R2	10001101001000010101100001111010
R3	10101011011110100111010000110010
R4	01100010001001001101001110100110
R5	00001101000111110101110111010001
R6	11010000011101101001001101001101
R7	10100100000100000001111110010110
R8	11010111101110011001010010011100
R9	00110010011110010110101100111111
R10	10101100101101011110100100111110
R11	00000110110100010000110001010110
R12	01000011110011001110011000101001
R13	00001101101011101000110011111000
R14	01110000101011100101101111000010
R15	01011001101001010011011111111110
R16	00011011001010011000011010100111

5.5. DES output observations with respect to *S*-BOX

Plain Text. 4F4F4F4F4F4F4F4F

Key. 4F4F4F4F4F4F4F4F

Table 7. Bits altered for each round.

L1	0	R1	1
L2	1	R2	3
L3	3	R3	13
L4	13	R4	15

L5	15	R5	20
L6	20	R6	17
L7	17	R7	18
L8	18	R8	18
L9	18	R9	16
L10	16	R10	16
L11	16	R11	18
L12	18	R12	24
L13	24	R13	19
L14	19	R14	16
L15	16	R15	19
L16	19	R16	18

Table 8. Round Function.

Round Function
0010000001001001001000001111000100000000000001000001
1111010100111010101010111110010111010110101111010
0110010011111011010101100001010101111000000000010
0011100100101100111111011010011101111111000000111
0111101101111111010101011110110110000100110100010
0011111101111111111111101111010101110001011010011
1110111001111100110111111111100001110000101000100
1110011011110110011111111100101100000001011101000
0100100100001010110111000110111010000000100000110
1111111101101011001010100111000011101011111111100
0010101011010000111110011011101000010100000101111
1010101010010011100000001101011110000101010000101
1001111110110001110100100111011111100011111111011
1010101010010010100001111010100011000010111110100
0100101001101110110110010011101110000001010100011
1010101000010010110111010011101110010110101101111

Table 9. S-BOX output with respect to round function.

S-Box Output (after each round)
00101111001110000110110011010001
01100000111100011010001110100011
10011110100001010010111000010010
10000111111100100001011100001000
01111100001000100100111000111001
00011100110000100011001110010101
00000001111110010000100010111000
10100010110011101001011110011001
10101001100101001001001000100100
11010110001110111110100001110101
11110100011111001000010000111101
01100011010111011110011100111101
00100101100100101000011010100101
01100011001101011101011010011010
10101011010100010110000100000001
01101101010011100110011010100000

Table 10. Alteration of Bits.

Round Function	Bits Changed
001000000100100100100001111000100000000000001000001	0
111101010010111010101010101001011001011010111110	6
010100010100010001010001111000000110111010100010	25
010111000010110001010101101100100101011000001110	14
010010110000000010101010001011011110111000101110	29
0000111100101011000000010000010100101110010001111	26
100011100011110110001011111011110101110100011001	22

0101100101110110011110010101101110111110101111110	21
1010100101001010011100010010111001101011000010 01	19
0011010100111110110100110001101001101000111100 11	25
0110010110010000111101100110010111101000110110 10	29
1101010111101101011110011000000011011100100010 00	31
0001010101011001001010001100110100100011000000 01	26
0100111100101111110101000101110100101011010111 11	31
0011111100000111011110010011101101111100010100 10	23
1010111111111010110110101100000100010100111000 11	20

Table 11. Box Output.

Round Function Bit Altered	S-BOX Output (After Every Round)	Change in Bits
0	11011111001110000110110011010001	4
3	10100111101010110001000010101000	17
6	01101100001001000110111101011011	13
9	10111110010000100111001001010001	15
12	10100101011010110111101100000010	17
15	11111000100011011100001011110100	17
18	11001000110000010100000101100000	14
21	11001010110000101111000001011000	13
24	01100010011010011110010101111010	23
27	11010000010101111110001011011 1100	14
30	10010110011111100011011010110000	12
33	00111010111011000100011111110110	15
36	01110001010011001111110101110001	19
39	01101000110010001010110100010010	18
42	00010101111111000110011110010110	18
45	10011001100111001111011000110111	16

The proposed system has shown the strict Avalanche effect when compared to the original DES. With the alteration of every single bit by bit (size 64 bit), the results of Avalanche effect of proposed system which is greater than or equal to 50% is 54.32% but original DES resulted in 53.73% and also observed the changes in DES with respect to round wise (left and right rounds' average percentage change of bits are 45.50% and 49.02% respectively. This round wise analysis is done by changing single bit by bit of permutation box of size 32 bits of original and also observed the S-Box outputs after every round with the percentage change of bits is 47.85%.

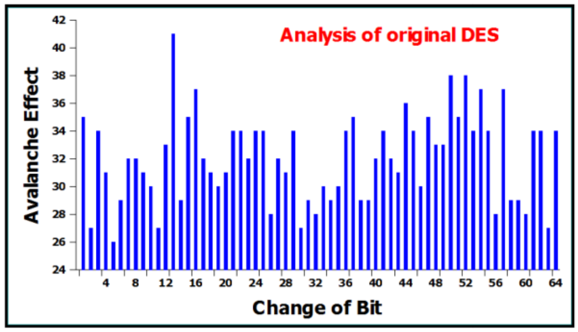

Figure 5. Analysis of original DES.

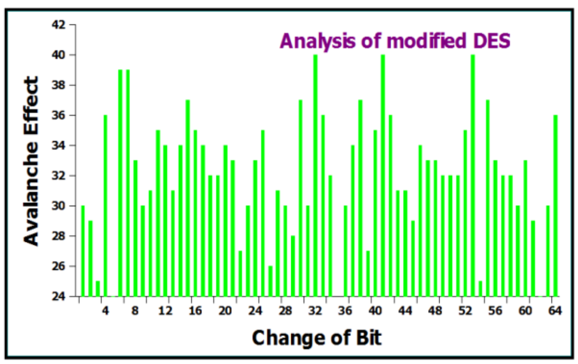

Figure 6. Analysis of original DES with respect to addition permutations for shifts.

6. Conclusion and Future Scope

This concludes that the proposed system gives more security with increased performance of Avalanche Effect. The proposed system can be extended on the same domain for further increase of the Avalanche Effect in future.

References

[1] S. Ramanujam and M. Karuppiah, Designing an algorithm with high Avalanche effect, International Journal of Computer Science and Network Security 11(1) (2011), 106-111.

[2] E. Biham and A. Shamir, Differential Cryptanalysis of DES-like Cryptosystems, J. Cryptology 4(1) (1991), 3-72.

[3] G. Singh and Supriya, A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security, International Journal of Computer Applications (0975-8887) 67(19) (2013).

[4] F. Saeed, A. B. Abdul Qadir, M. Yar Mughal and M. Rashid, A Novel Key Generation for FMET, International Journal of Computer Science and Network Security 11(6) (2011), 197-202.

[5] D. Coppersmith, The Data Encryption Standard (DES) and its strength against attacks, IBM Journal of Research and Development (1994), 243-250.

[6] Amish Kumar and T. Namita, Effective Emplementation and Avalanche Effect of AES International Journal of Security, Privacy and Trust Management (IJSPTM) 1(3/4) (2012).

[7] M. Akash Kumar and T. Archana, Analysis of Avalanche Effect in Plaintext of DES using Binary Codes, IJJETTCS 1(3), ISSN 2278-6856. (2012),

[8] C. Parikh and P. Parimal, Performance Evaluation of AES Algorithm on Various Development Platforms, Consumer Electronics, 2007. ISCE 2007. IEEE International Symposium on.

[9] C. Rajdeep, A. Sonam, M. Sridipta, Kr Vineet A. Sunit and J. K. Mandal, Triple SV: A Bit Level Symmetric Block Cipher Having High Avalanche Effect, (IJACSA) International Journal of Advanced Computer Science and Applications 2(7) (2011).

[10] P. Ganesh, A. Nitin and T. Sitendra, A block based encryption model to improve Avalanche effect for data security, International Journal of Scientific and Research Publications 3(1) January (2013).

[11] S. Gopikrishna, K. Vijaya Sree, K. Raja Sekhar, Sankeerthanareddy and D. Priyanka, Implementation of Parallelism on Block Cipher Modes Using DES In Lab View, International Journal of Applied Engineering Research (IJAER), Volume 10, (2015).

[12] K. Raja Sekhar and P. Jetty, Analysis of avalanche effect in modified des algorithm, International Journal of Applied Engineering Research (IJAER), 2015.