INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY
WINGS TO YOUR THOUGHTS…..

Design and Evaluation of Performance of Cryptography Technique based on DES

Kavita Sharma[1], Pardeep Tyagi[2], Abhinav Juneja[3]

[1]M.tech Student , DCRUST University,
B.M.I.E.T , Sonepat
[2]Assistant Professor, MDU, Rohtak,
A.I.T.M , Palwal,
[3]Associate Professor, GGSIPU, Delhi

Abstract: *Cryptography is an emerging technology which uses the characteristics of human vision to decrypt encrypted images. It does not require cryptography knowledge and complex computation. It also ensures that hackers cannot perceive any clues about a secret image from individual cover images for security concerns. There are many challenges, which one face when design a security model. The requirements of the security model depend upon the type of data to be encrypted. There is main problem to be considered, is the computational speed of the encryption model. There are many security models, which provide high level of security. Such model increases the execution time for encryption. The effectiveness of the algorithms highly depends upon computational time or computational speed. There is a main problem to Mono-alphabetic Cipher, that it can be broken because same plain letters are encoded to same cipher letter and the underlying letter frequencies remain unchanged. This problem can be solved by assigning various cipher letters or symbols to same plain letters. This can be implemented by a poly-alphabetic ciphering and deciphering technique. The proposed method is based on DES and it has similar properties and structure to DES with much smaller parameters. The proposed encryption algorithm takes an 8-bit block of plaintext (example: 10111101) and a 10-bit key as input and produces an 8-bit block of cipher text as output. The proposed decryption algorithm takes an 8-bit block of cipher text and the same 10-bit key used to produce that cipher text as input and produces the original 8-bit block of plaintext. A time effective symmetric key algorithm for small amount of data has been proposed. The performance of proposed technique will be evaluated through text received and computation time. MATLAB R2013a will be used as an implementation platform.*

Keywords: *Data Encryption Standard, Encryption, Cryptography, Security*

1. INTRODUCTION

Cryptography is the study of methods of sending messages in disguised form so that only intended recipients can remove the disguise and read the message. Cryptography is the practice and study of techniques for secure communication in the presence of third parties [11]. To accomplish this task, the original text, called plaintext, is translated into an encrypted version called cipher text, which is sent to the intended recipient. The recipient decrypts the text to obtain the original message [8]. The process of converting a plaintext to cipher text is called enciphering and the reverse process is called deciphering. Cryptanalysis is the study of methods for obtaining the meaning of encrypted information, without access to the secret information which is normally required to do so. Cryptanalysis is one of the challenging research areas in the discipline of security. Typically, this involves finding the key that is used for disguising the message. An attack on Cipher text may be of various types. . Medical MRI primarily images the MRI signal from the hydrogen nuclei in the body tissues [4]. In cipher text only attack, the encryption algorithm used and the cipher text to be decoded are known to cryptanalyst [9].

Many digital services require consistent security in the storage and transmission of digital images. Due to quick growth of the Internet in the world, nowadays, the safety of digital images has turn into more necessary and much involved attention. In order to fulfil the security requirements of digital images, many image encryption approaches have been used [1].

The Simplified Data Encryption Standard (SDES) is a simplified version of the well known Data Encryption Standard (DES) algorithm. The SDES has been designed for academic purposes and is used as a benchmark for cryptanalysis [2].

It is well known that cryptanalysis of the SDES scheme is an NP-hard problem and that met heuristics are well designed to solve combinatorial and difficult problems. By exploring a large set of solutions that improve over time, evolutionary algorithms have been successful for solving difficult and challenging problems. Even if the SDES is an academic and fairly easy problem that can be solved with an exhaustive search (as the key length is only 10 bits, there are no more than 2^10 = 1024 keys to try) it is used as a starting example for meta-heuristics and evolutionary Cryptanalysis [2]

1.1 SYMMETRIC KEY ALGORITHM

There are two primary types of symmetric algorithms:
(a) Block Cipher
(b) Stream Cipher

A block cipher is used to encrypt a text to produce a cipher text, which transforms a fixed length of block data size into same length block of cipher text in which a secret key and algorithm are applied to the block of data. Data Encryption Standard (DES), Triple-DES, IDEA, Simplified-DES and RC2 are examples of symmetric block cipher [3]. The symmetric key algorithms employ a solitary key for encryption and decryption process [1].

1.2 DES Algorithm

The DES algorithm was issued by the National Bureau of Standards (NBS) in 1977 [3] [6]. DES (Data Encryption Standard) algorithm purpose is to provide a standard method for protecting sensitive commercial and unclassified data. In this same key used for encryption and decryption process

[7].The algorithm enciphers 64-bit data blocks using a 56-bit secret key (not including parity bits which are part of the 64-bit key block). The algorithm employs three different types of operations: permutations, rotations, and substitutions [3]. DES is a block cipher. It encrypts the data in a block of 64 bits. It produces 64 bit cipher text. The key length is 56 bits. Initially the key is consisting of 64 bits. The bit position 8, 16, 24, 32,40,48,56, 64 discarded from the key length [5].

DES is based on two fundamental attributes of cryptography: Substitution (confusion) and transposition (Diffusion). DES consists of 16 steps, each of which called as a Round.

Algorithm:- 1. In the first step, the initial 64-bit plain text block is handed over to in Initial Permutation (IP) function.

2. The Initial permutation is performed on plain text.

3. The initial permutation produce two halves of permuted block: Left Plain text (LPT) and Right Plain Text (RPT).

4. Now, each of LPT and RPT goes through 16 rounds of encryption process, each with its own key:

a. From the 56-bit key, a different 48-bit Sub-key is generated using Key Transformation.

b. Using the Expansion Permutation, the RPT is expended from 32 bits to 48 bits.

c. Now, the 48-bit key is XORed with 48-bit RPT and resulting output is given to the next step.

d. Using the S-box substitution produced the 32-bit from 48-bit.

e. These 32 bits are permuted using P-Box Permutation.

f. The P-Box output 32 bits are XORed with the LPT 32 bits.

g. The result of the XORed 32 bits are become the RPT and old RPT become the LPT. This process is called as Swapping

h. Now the RPT again given to the next round and performed the 15 more rounds. [5] After the completion of 16 rounds the Final Permutation is performed [5].

1.3 SIMPLIFIED DATA ENCRYPTION STANDARD

The SDES algorithm [2] is a simple encryption algorithm; it was devised for pedagogical purposes. It is a symmetric-key algorithm which means that the sender and the recipient share the same key.

The S-DES encryption algorithm takes an 8-bit block of plaintext (example: 10111101) and a 10-bit key as input and produces an 8-bit block of cipher text as output. The S-DES decryption algorithm takes an 8-bit block of cipher text and the same 10-bit key used to produce that cipher text as input and produces the original 8-bit block of plaintext [10].

The algorithm consists in five steps: an initial permutation (IP), a complex function fk, another permutation function (SW), another application of the fk function and eventually a final permutation (IP-1). This final permutation is the inverse of the initial permutation.

The function f*K* takes as input not only the data passing through the encryption algorithm, but also an 8-bit key. The algorithm could have been designed to work with a 16-bit key, consisting of two 8-bit sub keys, one used for each occurrence of f*K*. Alternatively, a single 8-bit key could have been used, with the same key used twice in the algorithm. A compromise is to use a 10-bit key from which two 8-bit sub keys are generated. In this case, the key is first subjected to a permutation (P10). Then a shift operation is performed. The output of the shift operation then passes through a permutation function that produces an 8-bit output (P8) for the first sub key (*K*1). The output of the shift operation also feeds into another shift and another instance of P8 to produce the second sub key (*K*2) [10].

Simplified-Data Encryption Standard (S-DES) is a reduced adaptation of the Data Encryption Standard (DES) algorithm. It was designed as a test block cipher for learning about modern cryptanalytic techniques such as linear cryptanalysis, differential cryptanalysis and linear-differential cryptanalysis. It is a variation of basic DES. In Simplified-DES, the same key is used for encryption and decryption in fig1 [1].

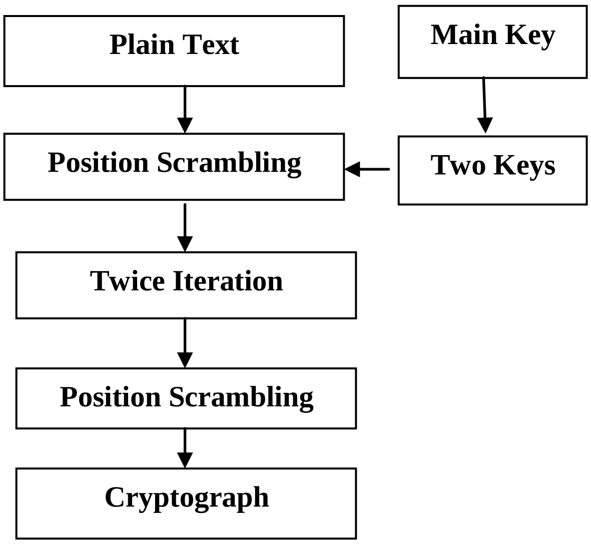


Fig. 1 S-DES Structure [1]

1.3.1 ADVANTAGES OF S-DES [1]

1. It is simpler than Data Encryption Standard.

2. It takes smaller block of plaintext and use small key in encryption than DES.

3. Its execution speed is faster than DES.

1.3.2 LIMITATIONS OF S-DES [1]

1. The key size is low in this algorithm.

2. Due to low key size, the security of S-DES algorithm is reduced.

3. If we use lots of data such as an image, then that algorithm can't satisfy the encryption requirement.

1.4 ENCRYPTION ALGORITHM

The block schematic of the SDES encryption algorithm is shown in Fig. 3. The Encryption process involves the sequential application of five functions:

1.4.1 S-DES KEY GENERATION

S-DES depends on the use of a 10-bit key shared between sender and receiver. From this key, two 8-bit sub keys are produced for use in particular stages of the encryption and decryption algorithm [10]. The block schematic of the S-DES Key generation algorithm is shown in Fig. 2 For key generation, a 10-bit key is considered from which two 8-bit sub-keys are generated. In this case, the key is first subjected to a permutation P10= [3 5 2 7 4 10 1 98 6], then a shift operation is performed. The numbers in the array represent the value of that bit in the original 10-bit key. The output of the shift operation then passes through a permutation function that produces an 8-bit output P8 = [6 3 74 8 5 10 9] for the first sub key (K1). The output of the shift operation also feeds into another shift operation and another instance of P8 to produce the second sub key K2. In all bit strings, the leftmost position corresponds to the first bit [9]

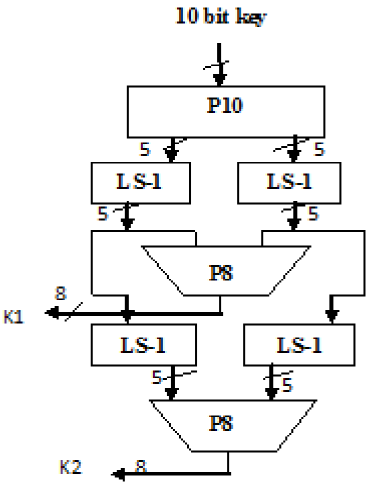


Fig. 2 Key Generation for S-DES

1.4.2 INITIAL AND FINAL PERMUTATIONS

The input to the algorithm is an 8-bit block of plaintext, which is first permuted using the IP function IP = [2 6 3 1 4 8 5 7]. This retains all 8-bits of the plaintext but mixes them up. At the end of the algorithm, the inverse permutation is applied; the inverse permutation is done by applying, IP-1 = [4 1 3 5 7 2 8 6] Where, IP-1(IP(X)) = X [9].

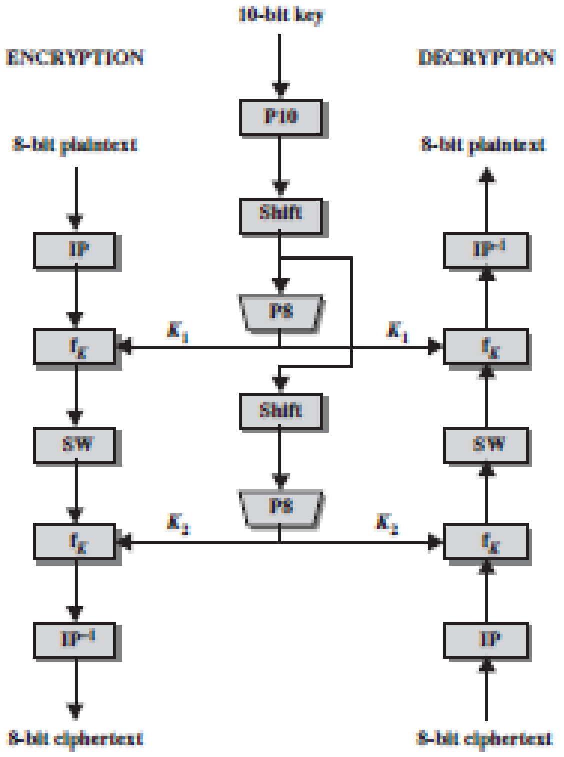


Fig. 3 S-DES Encryption [9]

1.4.2 THE FUNCTION f*K*

The most complex component of S-DES is the function f*K*, which consists of a combination of permutation and substitution functions. The functions can be expressed as follows. Let *L* and *R* be the leftmost 4 bits and rightmost 4 bits of the 8-bit input to f*K*, and let F be a mapping (not necessarily one to one) from 4-bit strings to 4-bit strings [10]. Then we let

$$f_K(L, R) = (L \oplus F(R, SK), R)$$

where *SK* is a sub key and $\oplus$ is the bit-by-bit exclusive-OR function. For example, suppose the output of the IP stage in Figure G.3 is (10111101) and F(1101, *SK*) = (1110) for some key *SK*. Then f*K* (10111101) = (01011101) because (1011) $\oplus$ (1110) = (0101).

THE SWITCH FUNCTION:

The function f*K* only alters the leftmost 4 bits of the input. The switch function (SW) interchanges the left and right 4 bits so that the second instance of f*K* operates on a different 4 bits. In this second instance, the E/P, S0, S1, and P4 functions are the same. The key input is *K*2.

2. PROPOSED METHODOLOGY

Opening the file being ciphered.

1. Opening a new file to store ciphered text.
2. Reading files for the first time to know number of bytes.

Enciphering of input text Data

3. Requesting ciphering key from user.
4. Conversion of decimal key into equivalent 10 bit binary vector.
5. Generating k1 and k2 from given key.
6. This function generates two 8-bit keys: k1, k2 from a given 10-bit key.
7. Declaration of empty array where ciphered text will be stored.
8. Reading one byte at once from original file.
9. Converting ASCII data into binary values.
10. Initial permutation of equivalent logical matrix of text data in input.
11. Ciphering or encoding of left 4 bits of logical matrix of text data in input using key k1.
12. Combining of encoded left and right halves.
13. Ciphering or encoding of left 4 bits of new logical matrix of text data in input using key k2.
14. Again combining of encoded left and right halves.
15. Application of inverse permutation on the resultant matrix.
16. Conversion of ciphered byte and writing in to a special file.
17. Closing of original and ciphered text files.

Deciphering of ciphered data

18. Opening of the file being deciphered.
19. Opening of a new file to store deciphered text.
20. Reading of ciphered file for the first time to know number of bytes.
21. Declaration of empty array where deciphered text will be stored.

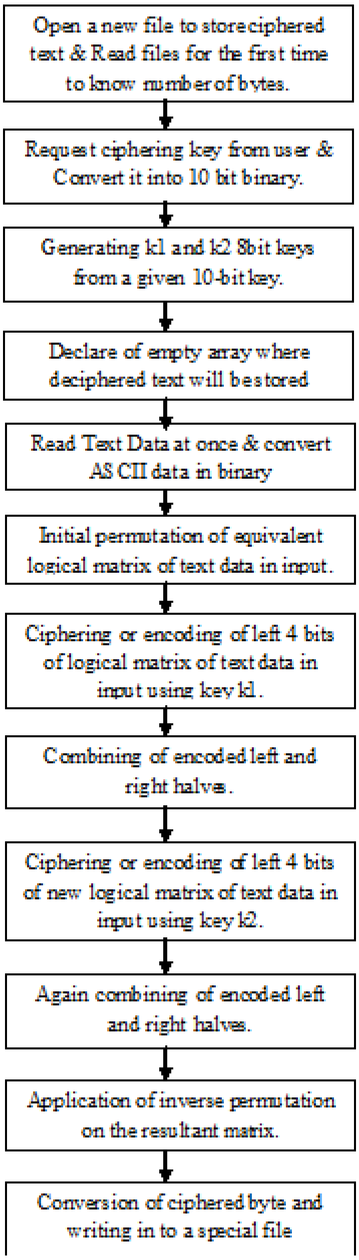

Open a new file to store ciphered text & Read files for the first time to know number of bytes.
Request ciphering key from user & Convert it into 10 bit binary.
Generating k1 and k2 8bit keys from a given 10-bit key.
Declare of empty array where deciphered text will be stored
Read Text Data at once & convert ASCII data in binary
Initial permutation of equivalent logical matrix of text data in input.
Ciphering or encoding of left 4 bits of logical matrix of text data in input using key k1.
Combining of encoded left and right halves.
Ciphering or encoding of left 4 bits of new logical matrix of text data in input using key k2.
Again combining of encoded left and right halves.
Application of inverse permutation on the resultant matrix.
Conversion of ciphered byte and writing in to a special file

Fig. 4 Flow Chart of Enciphering input data

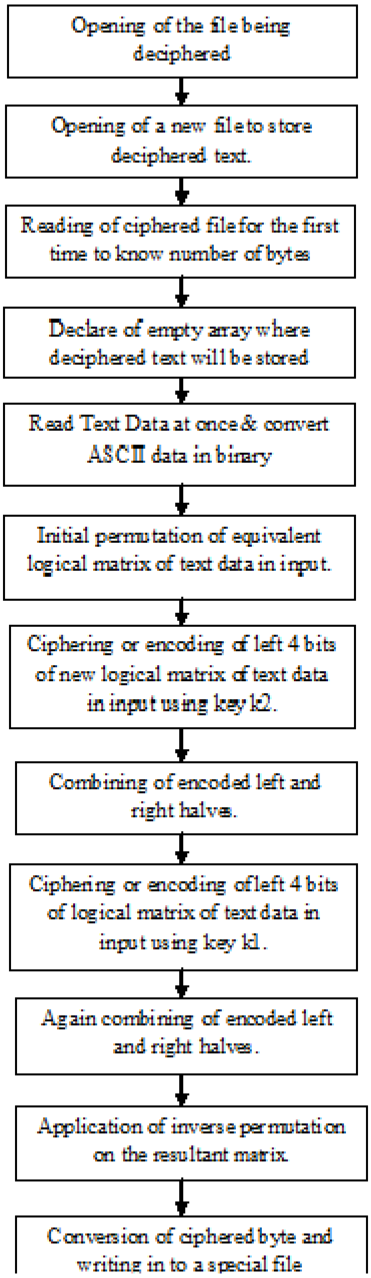
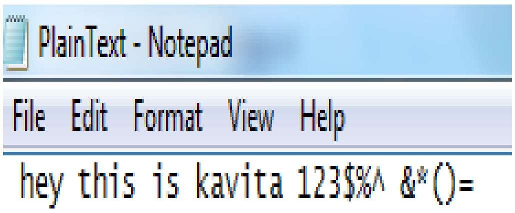

Opening of the file being deciphered
Opening of a new file to store deciphered text.
Reading of ciphered file for the first time to know number of bytes
Declare of empty array where deciphered text will be stored
Read Text Data at once & convert ASCII data in binary
Initial permutation of equivalent logical matrix of text data in input.
Ciphering or encoding of left 4 bits of new logical matrix of text data in input using key k2.
Combining of encoded left and right halves.
Ciphering or encoding of left 4 bits of logical matrix of text data in input using key k1.
Again combining of encoded left and right halves.
Application of inverse permutation on the resultant matrix.
Conversion of ciphered byte and writing in to a special file

Fig. 5 Flow Chart of Deciphering input data

22. Reading of text data at once from original file.
23. Conversion of ASCII data into binary values.
24. Initial permutation of equivalent logical matrix of text data in input.
25. Ciphering or encoding of left 4 bits of logical matrix of text data in input using key k2
26. Combining of encoded left and right halves.
27. Ciphering or encoding of left 4 bits of new logical matrix of text data in input using key k1.
28. Again combining of encoded left and right halves.
29. Application of inverse permutation on the resultant matrix.
30. Conversion of ciphered byte and writing in to a special file.
31. Closing of original and ciphered text files.

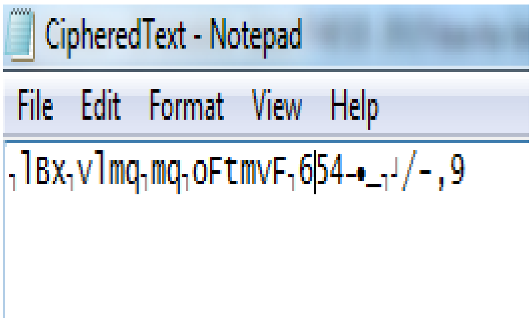
3. EXPERIMENTAL RESULTS

An advanced cryptographic method based on SDES is proposed in this work. Random permutation along with divison of data is implemented in proposed method. Both divided parts of the data undergoes different encoding operations, which makes the proposed system more robust and more secure. These methods don't enhance the mathematical complexions due to which proposed algorithm takes lesser time to encrypt and decrypt the inputted text. Figure 4 is a snapshot of text file containing the input text to be encrypted first and than to be decrypted. Figure 5 is a snapshot of text file containing the encrypted text. Figure 6 is a snapshot of text file containing the decrypted text. It is clear from figure 6 and 4 that proposed algorithm is working efficiently from problem formulated in the last section. Also, it would be impossible for an intruder to get the original information from encrypted text as shown in figure 5.



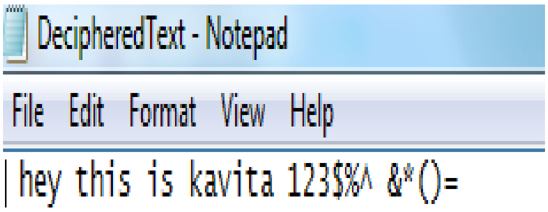
hey this is kavita 123$%^ &*()=

Fig. 6 snapshot of text file containing the input text to be encrypted first and than to be decrypted



¸7Bx¸v7mq¸mq¸oFtmvF¸654-•_¸˩/-,9

Fig. 7 snapshot of text file containing the encrypted text



hey this is kavita 123$%^ &*()=

Fig. 8 snapshot of text file containing the decrypted text

4. CONCLUISON & FUTURE SCOPE

An advanced cryptographic method based on SDES is proposed in this work. The proposed method is based on DES and it has similar properties and structure to DES with much smaller parameters. The proposed encryption algorithm takes an 8-bit block of plaintext (example: 10111101) and a 10-bit key as input and produces an 8-bit block of cipher text as output. The proposed decryption algorithm takes an 8-bit block of cipher text and the same 10-bit key used to produce that cipher text as input and produces the original 8-bit block of plaintext. A time effective symmetric key algorithm for small amount of data has been proposed. The performance of proposed technique will be evaluated through text received and computation time. Random permutation along with divison of data is implemented in proposed method. Both divided parts of the data undergoes different encoding operations, which makes the proposed system more robust and more secure. These methods don't enhance the mathematical complexions due to which proposed algorithm takes lesser time to encrypt and decrypt the inputted text. In future analysis we can also reduce the computation time & complexity of data encryption algorithms.

REFERENCES

[1] Sanjay Kumar and Sandeep Srivastava "Image Encryption using Simplified Data Encryption Standard (S-DES)" International Journal of Computer Applications (0975 – 8887) Volume 104 – No.2, October 2014. Pp. 38-42.

[2] Fabien Teytau and Cyril Fonlupt "A Critical Reassessment of Evolutionary Algorithms on the Cryptanalysis of the Simplified Data Encryption Standard Algorithm" International Journal on Cryptography and Information Security (IJCIS), Vol. 4, No. 2, June 2014. Pp. 1-11.

[3] Hans Eberle "A High-speed DES Implementation for Network Applications" September 23, 1992.

[4] Young-Chang Hou "Visual cryptography for color images" Pattern Recognition 36 (2003) 1619 – 1629.

[5] Sombir Singh, Sunil K. Maakar and Dr.Sudesh Kumar "Enhancing the Security of DES Algorithm Using Transposition Cryptography Techniques" International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 6, June 2013.pp. 464-471.

[6] Jawahar Thakur1\, Nagesh Kumar "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis"International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, Volume 1, Issue 2, December 2011).pp. 6-12.

[7] Dr. Prerna Mahajan & Abhishek Sachdeva "A Study of Encryption Algorithms AES, DES and RSA for Security" Global Journal of Computer Science and Technology Network, Web & Security Volume 13 Issue 15 Version 1.0 Year 2013.

[8] Sunita Bhati, Anita Bhati, S. K. Sharma "A New Approach towards Encryption Schemes:Byte – Rotation Encryption Algorithm" Proceedings of the World Congress on Engineering and Computer Science 2012 Vol II WCECS 2012, October 24-26, 2012, San Francisco, USA.

[9] Vimalathithan.R, Dr.M.L.Valarmathi " Cryptanalysis of S-DES using Genetic Algorithm" International Journal of Recent Trends in Engineering, Vol 2, No. 4, November 2009. PP. 76-79.

[10] William Stallings "Appendix G Siimpliifiied DES" Copyright 2010.

[11] Vikas Agrawal, Shruti Agrawal, Rajesh Deshmukh Analysis and Review of Encryption and Decryption for Secure Communication" International Journal of Scientific Engineering and Research (IJSER) Volume 2 Issue 2, February 2014. pp. 1-3.