Journal of Advances in Science and Technology
Vol. 18, Issue No. 2, September–2021, ISSN 2230-9659

A Study of Logical Exclusive Operations and Linear Transformations in Cryptography

Deepak Kumar Sharma[1]*, Dr. Birendra Singh Chauhan[2]

[1] Research Scholar, Shri Krishna University, Chhatarpur M.P.

[2] Associate Professor, Shri Krishna University, Chhatarpur M.P.

Abstract - In the present study the message is encrypted in blocks and each data block is encrypted in 8 rounds. In each round different linear transformations are applied on different elements of the data block. These linear transformations are sub-key dependent. The sub-key of each round is derived from the session key of that particular data block. The session key of each data block is generated from the master key (secret key) agreed upon by the communicating parities before communicating the messages. Between two successive linear transformation operations Exclusive-OR (XOR) operation is applied on each element with its nearest four neighbors in the matrix array to ensure good avalanche. In the key scheduled algorithm proposed in this study different keys are used for encrypting different data blocks which are called session keys generated from the master key (secret key between the sender and the receiver) and the key used for the encryption of each round is different and is derived from the session key of the corresponding block which is called the sub key. As different keys are used for encrypting different data blocks cipher is less vulnerable to passive attacks. The binary bits of each element of the message matrix M in each round are rotated right in the encryption of the message. The rotation is not fixed rotation. The number of rotations of the bits of each element of the matrix M depends on the sub key of that particular round of encryption. Logical XOR operation is performed on each element between two successive rotation operations with all its nearest neighboring elements. Hence, the identical characters in the plain text are mapped to different cipher characters even though they are in the same text block or in different text blocks. So, cipher text is not easily amenable to cryptanalysis. Even the change of a single character of the message changes almost the entire cipher block, i.e., to say that the proposed algorithm of this study has achieved a good avalanche effect.

Keywords - Logical Exclusive Operations, Linear Transformations, Cryptography, linear transformations, blocks cipher.

---------------------------------x--------------------------------

INTRODUCTION

In the key scheduled algorithm for encryption/decryption of the data proposed in this study, the size of the data block is selected to be 64 characters. The characters of each data block are coded to hexadecimal numbers using ASCII code table and are written as an 8x8 matrix row-wise. Each hexadecimal element is encrypted in 8 rounds using linear transformation operation so that the outcome is a new element. The key used in the linear transformation operation is different for different elements in each round of encryption i.e. the linear transformation is not fixed, but depends on the sub key. The sub key is derived for each round of encryption from the session key of that particular data block. Several researchers of cryptography used the logical operation XOR in their encryption protocols. The famous Hill cipher uses the linear transformation (involutory transformation of the type z z T) operation for enciphering the messages. Xiaolin Wang et.al discussed the uses of linear transformations in cryptography based on the definition of linear transformations, the Walsh linear spectrum and cyclic spectrum and their inverse transformations.

Hence, active attacks such as chosen plain text attacks chosen cipher text attacks are quite difficult to execute. Hence, the proposed algorithms are less vulnerable to active attacks. The present encryption algorithm is secure against man-in middle attack because the entire master key is agreed upon by the sender and the receiver rather than the electronic exchange of the parts of the key. The communicating parties can securely transport the key using key transport protocols using ECC as proposed in part (i) of study 4.The proposed key scheduled algorithm in this study is less prone to timing attacks because the time required to encipher or decipher a data block is same for all data blocks since time for enciphering or deciphering is independent of characters in the data block. Even though the original message contains less than 64 characters the remaining characters are filled at random, so that each data block contains exactly 64 characters. The size of the key is 64 decimal digits where each decimal digit takes values from 0 to 7. Hence 648 different keys are possible. It is estimated that on a 4GHz single core processor the time required to encipher/ decipher a text block is 18µsec. Hence, the vulnerability to brute force attack is very less [the life time of a human being i.e., 100years] is approximately equal to 3Gsec. The time required to try all possible keys to decipher a single cipher block by brute force method is roughly 5Gsec. This block cipher ensures high level of security if the secret key agreed upon by the communicating parties is unbreakable.

LINEAR TRANSFORMATION:

Let V and W be two vector spaces over the field F. A linear transformation from V into W is a function T from V into W such that T (cα + β) = c (T α) + Tβ for all α and β in V and all scalars c in F.

PROPERTIES OF LINEAR TRANSFORMATIONS:

If T is a linear transformation from the vector space V(F) to the vector space W(F) then

i). T(x) = x for all x in V, Here T is called the identity transformation

ii). (-T) x = - [T(x)], -T is the negative transformation of T

iii). T (0) = 0 where the left hand zero belongs to V and the right hand zero belongs to W

iv). T (-x) = -T(x) for all x in V

v). T(x-y) = T(x) – T(y) for all x,y in V

ALGEBRA OF LINEAR TRANSFORMATIONS:

If V, W be two vector spaces over the field F and let T and U be two linear transformations from V into W then

i). (T + U) (α) = Tα + Uα

ii). if c is a scalar in the field F then (cT) (α) = c T (α).

PROPOSED WORK:

For describing the algorithm the following notation and definitions are adopted:

1 Symbols and Notation:

Symbol	Expression	Meaning
M,m,n,l	${}^{l}M_{n}^{m}$	The 8x8 matrix whose elements are hexadecimal numbers, where *l,m,n* take integer values, $n \in N$ $l, m \in$ {0,1,2……8]
K,m,n	K_{n}^{m}	The 8x8 matrix whose elements are the hexadecimal codes of ASCII characters **excluding the null character**
R,E, ∧	$\hat{L}_E$	Encryption Operator using Linear Transformation
R,D, ∧	$\hat{L}_D$	Decryption Operator using Linear Transformation
X,E, ∧	$\hat{X}_E$	The logical XOR operator used in the encryption process
X,D, ∧	$\hat{X}_D$	The logical XOR operator used in the decryption process
A	A = {1,3,5,7,9,B,D,F}	A is the set of all numbers which are relative primes to 16, used for obtaining the principal key matrix for the encryption of the matrix ${}^{l}M_{n}^{m}$ using linear transformation
P, ∧	$\hat{P}$	The operator used to obtain the principal key matrix from the sub-key used in that particular round
P,n,m	P_{n}^{m}	The principal key matrix obtained from K_{n}^{m}
S, ∧	$\hat{S}_{n}^{m}$	The operator used for deriving the sub key for the m^{th} round encryption from the key used for the first round encryption of n^{th} data block
G, ∧	$\hat{G}_{n}$	The operator used for generating the session key for the encryption of the n^{th}data block from the session key used in the encryption of the $(n-1)^{th}$data block.
M,l,m,n,i,j	$\left[{}^{l}M_{n}^{m}\right]_{ij}$	represents the element in the i^{th} row and j^{th} column of the matrix $\left[{}^{l}M_{n}^{m}\right]$

2 Definitions:

1) ${}^{l}M_{n}^{m}$ denotes an 8x8 matrix whose elements are hexadecimal numbers. The right superscript m denotes the number of times the linear transformation operator $\hat{L}_E$ is performed on the elements of the matrix M. The left superscript l represents the number of times the logical XOR operator $\hat{X}_E$ is applied on the matrix M. The right subscript n indicates the number of data block that is being encrypted.

2) K_{n}^{m} represents an 8x8 matrix whose elements are the hexadecimal codes of ASCII code table excluding the null character which is called the key matrix used in the encryption operation of the (m+1)th round of the n th data block matrix m m Mn using linear transformation. The right superscript m of K represents the number of round of encryption using the linear transformation on the matrix ${}^{m}M_{n}^{m}$, where m ranges from 0 to 7. The right subscript indicates the data block which is being encrypted.

3). P_n^m is the Principal Key matrix obtained from the main key matrix K_n^m for $(m+1)^{\text{th}}$ round of encryption of nth data block using linear transformation.

Let $(K_n^m)_{ij} = K_1K_0$, where K_1, K_0 the hexadecimal digits in 16' place and 1's place

$$(P_n^m)_{ij} = P_1P_0 \text{ Where}$$

$$P_0 = K_0, P_1 = Y \text{ if } Y \in A \text{, else } Y+1 \text{, where } Y = (K_1K_0 + K_1 + K_0)_{\text{mod}16}$$

$$\hat{P}(K_n^m) = P_n^m$$

4). The linear transformation used in the algorithm is of the form

$Z = (az+b)_{\text{mod}16}$ where a∈ A and b may be any hexadecimal numbers from 0 to F. Then z can be obtained by the inverse linear transformation $z = [a^{-1}(Z+b^{-1})]_{\text{mod}16}$

For all $a \in$ A , a^{-1} is defined in the following table 1 and b^{-1} is defined in the table 2

TABLE 1

a	1	3	5	7	9	B	D	F
a^{-1}	1	B	D	7	9	3	5	F

TABLE 2

b	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
b^{-1}	0	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1

5). The Operator $\hat{L}_E[{}^mM_n^m, P_n^m]$ is the operator that affects linear transformation. It has two arguments ${}^mM_n^m$ and P_n^m. The first argument is the matrix on which the operation $\hat{L}_E$ is applied. The second argument is the key matrix used for performing the operation $\hat{L}_E$ in the $(m+1)^{\text{th}}$ round encryption of n^{th} data block matrix. With this operation the right superscript m of the matrix ${}^mM_n^m$ increases by one unit. With this operation each hexadecimal element $[{}^mM_n^m]_{ij}$ of the matrix ${}^mM_n^m$ is encrypted using the linear transformation.

Let $[{}^mM_n^m]_{ij} = M_1M_0$ which is a hexadecimal number. M_1M_0 is encrypted as

$$M_1^E = (M_1P_1 + P_0)_{\text{mod}16} \quad M_0^E = (M_0P_1 + P_0)_{\text{mod}16}$$ where P_0 and P_1 can be obtained as in the above definition

For example let $[{}^mM_n^m]_{11} = 7F$ and let $P_1 = 5$ and $P_0 = 3$ then 7F is encrypted as

$$(7.5+3)_{\text{mod}16} = 6 \text{ and } (F.5+3)_{\text{mod}16} = E \text{ So, 7F is encrypted as 6E}$$

$$\hat{L}_E[{}^mM_n^m, P_n^m] = {}^mM_n^{m+1}$$

All the elements of the matrix ${}^mM_n^m$ are encrypted using the linear transformation and are written in the form of 8 bit binary numbers.

6). $\hat{X}_E[{}^mM_n^{m+1}]$ represents the XORing of each element of the matrix ${}^mM_n^{m+1}$ with the nearest four neighboring elements in the encryption process. With this operation the left superscript m increases by one unit. With this operation each element $[{}^mM_n^{m+1}]_{ij}$ of the matrix ${}^mM_n^{m+1}$ which is in the 8 bit binary format is XORed with the nearest four neighboring elements which results in the matrix ${}^{m+1}M_n^{m+1}$

$$\hat{X}_E[{}^mM_n^{m+1}] \to ((((({}^mM_n^{m+1})_{ij} XOR\, {}^mM_n^{m+1}{}_{ij-1}) XOR\, {}^mM_n^{m+!}{}_{i+1,j}) XOR\, {}^mM_n^{m+1}{}_{ij+1}) XOR\, {}^mM_n^{m+!}{}_{i-1j})$$

$$\hat{X}_E[{}^mM_n^{m+1}] = {}^{m+1}M_n^{m+1}$$

Example $[{}^mM_n^{m+1}]_{24} = 10010100$, $[{}^mM_n^{m+1}]_{23} = 10100010$, $[{}^mM_n^{m+1}]_{34} = 00111000$

$[{}^mM_n^{m+1}]_{25} = 10010010$, $[{}^mM_n^{m+1}]_{14} = 01100101$ then

$$\hat{X}_E[{}^mM_n^{m+1}]_{24} = ((((10010100\,XOR\,10100010)\,XOR\,00111000)\,XOR\,10010010)\,XOR\,01100101)$$

$$[{}^{m+1}M_n^{m+1}]_{24} = 11111001$$

All the elements of the matrix ${}^{m+1}M_n^{m+1}$ which are in 8 bit binary numbers are converted into hexadecimal numbers

7. $\hat{X}_D[{}^mM_n^m]$ represents the XORing of each element of the matrix ${}^mM_n^m$ with the neighboring elements in the decryption process. With this operation the left superscript m decreases by one unit. All the elements of the matrix ${}^mM_n^m$ are XORed with the nearest neighboring elements as

$$\hat{X}_D[{}^mM_n^m] \to (((({}^mM_n^m{}_{ij}\,XOR\,{}^mM_n^m{}_{i-1j})\,XOR\,{}^mM_n^m{}_{ij+1})\,XOR\,{}^mM_n^m{}_{i+1j})\,XOR\,{}^mM_n^m{}_{ij-1}).$$

$\hat{X}_D[{}^mM_n^m] = {}^{m-1}M_n^m$ where m ranges from 0 to 7.

$[{}^mM_n^m]_{45} = 11111001$, $[{}^mM_n^m]_{35} = 10100010$ $[{}^mM_n^m]_{46} = 00111000$ $[{}^mM_n^m]_{55} = 10010010$

$[{}^mM_n^m]_{44} = 01100101$ then

$$\hat{X}_D[{}^mM_n^m]_{45} = ((((11111001\,XOR\,01100101)\,XOR\,10010010)\,XOR\,00111000)\,XOR\,10100010)$$

$$[{}^{m-1}M_n^m]_{45} = 10010100$$

All the elements of the matrix ${}^{m-1}M_n^m$ which are in the 8 bit binary format are written as hexadecimal numbers.

8). The Operator $\overset{\wedge}{L_D}\left[{}^{m-1}M_n^m, P_n^m\right]$ is the decryption operator using linear transformation. It has two arguments ${}^{m-1}M_n^m$ and P_n^m. The first argument is the matrix on which the operation $\overset{\wedge}{L_D}$ is performed in the (9-m)th round of decryption of the nth data block matrix. The second argument P_n^m is the key matrix used for performing the linear transformation operation $\overset{\wedge}{L_D}$ in the mth round decryption of the nth data block matrix. With this operation the right superscript m of the matrix ${}^{m-1}M_n^m$ decreases by one unit. With this operation each hexadecimal element $[{}^{m-1}M_n^m]_{ij}$ of the matrix ${}^{m-1}M_n^m$ is decrypted using the linear transformation

Let $[{}^{m-1}M_n^m]_{ij} = M_1^E M_0^E$ then M_1 and M_0 can be obtained by taking the inverse linear transformation as

$$M_1 = [P_1^{-1}\{M_1^E + P_0^{-1}\}]_{\text{mod}16}$$

$$M_0 = [P_1^{-1}\{M_0^E + P_0^{-1}\}]_{\text{mod}16}$$

$$\text{where } P_1^{-1} = D, P_0^{-1} = 3$$

For example if $[{}^{m-1}M_n^m]_{11} = 6E$ then 6E is decrypted as

$$[D\{6+13\}]_{\text{mod}16} = 7$$
$$[D\{E+13\}]_{\text{mod}16} = F$$

$$\text{Where } P_1^{-1} = D \text{ and } P_0^{-1} = D$$

$$\overset{\wedge}{L_D}\left[{}^{m-1}M_n^m, K_n^m\right] = {}^{m-1}M_n^{m-1}$$

All the elements of the matrix ${}^{m-1}M_n^{m-1}$ which are hexadecimal numbers are converted into binary numbers.

9). $\overset{\wedge}{S_n^m}$ is the operator used for deriving the sub key for the (m+1)th round encryption of nth data block from the session key used for the first round operation i.e. $\overset{\wedge}{S_n^m}[K_n^0] = K_n^{m-1}$

The key matrix K_n^{m-1} for the mth round encryption of the nth data block is obtained from K_n^0 by shifting the columns of the matrix K_n^0 to the right by (m-1) places.

$$\text{i.e. } \left[K_n^{m-1}\right]_{ij} = \left[K_n^0\right]_{ij-m+1}$$

10). $\overset{\wedge}{G_n}$ is the operator which defines the session key generation for the first round encryption of the nth data block from the session key used for the first round encryption of the (n-1)th data block.

$$\overset{\wedge}{G_n}\left[K_{n-1}^0\right] = K_n^0, \text{where} \left[K_n^0\right]_{ij} = \left[(K_{n-1}^0)_{ij} + (K_{n-1}^0)_{ij+1}\right]_{\text{mod}8}$$

In all the above definitions i and j take values from 0 to 7. If i+1 or i-1 or j+1 or j-1or any subscript fall out of the range {0,1,2,….7} then modulo 8 of that number be considered. The session key of each data block is itself sub key for the first round of encryption of the data block. Before communicating the messages both the sender and the receiver agree upon to use the secret key which is in the form of an 8x8 matrix K (master key) whose elements are the hexadecimal numbers of ASCII code table excluding null character. This matrix K (master key) is denoted by 0 K1 in the encryption/ decryption process i.e. the master key itself is the session key for the encryption/decryption of first data block. For implementing the algorithm the entire message is divided into data blocks D1, D2, D3……Dn of 64 characters each where n is a natural number. The characters in each message block are hex coded using ASCII code table and are arranged in the form of 8x8 matrices ${}^0M_1^0, {}^0M_2^0, {}^0M_3^0, \ldots\ldots {}^0M_n^0$ row wise. The number of characters in the message always may not be the integral multiple of 64. Hence, at the end of the message the sender adds three # characters (###) and ensures that the message fills integer number of text blocks by adding random different characters after the three # characters.

CONCLUSION

In the proposed algorithm the adjacency matrix and the powers to which it is raised in each round of encryption are converted into decimal numbers and are sent to the receiver as a separate communication in public channel. This makes the retrieval of the secret key matrix from cryptanalysis is very difficult. Moreover the algorithm proposed here is much secured as long as the key is secret between the communicating parties. Single round of encryption offers inadequate security but multiple rounds of encryption offers increasing security due to improvement of avalanche effect. With this the identical characters in the plain text space are mapped to different characters of the cipher text space even though they are in the same text block or different text blocks. So, cipher text is not easily amenable to cryptanalysis. Even the change of a single element of the message matrix changes almost the entire cipher block matrix, i.e., to say that the proposed algorithm has achieved a good avalanche effect. Even though the original message contains less that 49 characters the remaining characters are filled at random, so that each data block contains exactly 49 characters. Thus the algorithm provides sufficient security against timing attacks at relatively low computational overhead. In the block cipher proposed above, linear transformation (not fixed) operation is used for the encryption of each character of the message. The session key is different for different data blocks. The sub-key is also different for different rounds of encryption. The logical XOR operation on each elements of the block is carried with four different elements of data array to ensure good avalanche. Cryptanalysis of the cipher proposed is difficult and the cipher is not easily amenable to all known types of attacks.

REFERENCES

[1] Brualdi, R. A. “Introductory combinatorics”, 4th ed. New York: Elsevier, 1997

[2] Canetti R. Halevi S. and Katz J. “Chosen cipher text security from identity-based encryption”, Advances in Cryptography-EUROCRYPT 2004, Vol. 3027 of LNCS, SpringerVerlag.

[3] Carlistle Aams and Stafford Tavares, “The structured design of cryptographically good s-boxes, Journal of Cryptology, 1990, Vol.3, No.1, Pages 27-41.

[4] Chandra Sekhar A., Suneetha CH., Naga Lakshmi G. “Self-encrypting data streams for digital signals”, Int. Journal computer and network security, Vol.2, No:4 April 2010, pp 111- 113.

[5] Chandra Sekhar A., Sunetha CH., Naga Lakshmi G. and Ravi Kumar B. “Fast Fourier transforms and quadratic forms for digital audio watermarking”, International Conference on Advances in Recent Technologies in communication and Computing”, Proc. IEEE Transactions, Oct 2009.

[6] Chandra sekhar A. et.al. “Some Algebraic Curves in public Key crypto systems”, International Journal of Ultra Scientist of Physical Sciences, 2007.

[7] Chandra sekhar A., Prasad Reddy PVGD., Murthy ASN, Krishna Gandhi B. “Self-encrypting data Streams using graph structures”, IETECH Journal of Advanced Computations Vol. 2, No:1, 007-009, 2008.

[8] Chen Hun-Chen, Yen Jui-Cheng and Guo Jiun-In “Design of a new cryptography system”, Lecture Notes in Computer Science, 2002,Vol2532/2002,211-219

[9] Cohn, H. “Advanced topics in computational number Theory”. New York: SpringerVerlag, 2000.

[10] Daemen J. and Rijmen V. “The design of Rijedael”, Information security and Cryptography conference, Springer 2002.

[11] Darrel Hankerson, Alfered Menezes, Scott Vanstone, “A Gide to elliptic curve cryptography”, Springer, 2004.

Corresponding Author

Deepak Kumar Sharma*

Research Scholar, Shri Krishna University, Chhatarpur M.P.