International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Generation of Keystream for Symmetric Cipher using U-Matrix

Mani. K, Devi. A

Abstract: ***Symmetric-key cryptography is a classical cryptography in which both sender and receiver use the same key K to encrypt and decrypt the message. The main challenge between sender and receiver is to agree upon the secret-key which should not be revealed to public. Key management is the major issue in symmetric-key cryptosystem. To avoid these, a novel approach in generating the keystream K_s for any symmetric-key algorithms using U-matrix is proposed in this paper. The advantage of this method is generation of key K from K_s is based on some deterministic procedure which is then applied to DES algorithm and K is not necessarily remembered by both sender and receiver. Further, in each round different key is used as opposed to usage of single key in classical DES. Experimental results clearly show the security is increased when it is compared with classical DES.***

Keywords : DES, Keystream, Symmetric-key, U-matrix.

I. INTRODUCTION

At present, security and privacy are the major problems facing by private sector to prevent an unauthorized extraction of information from communication over an insecure channel. Security can be provided to communications among people by using a derivative measure called cryptography. Since traditional wireless sensor networks affected by many types of attacks, security architects and services like integrity, authenticity, confidentiality and non-repudiation are vital requirements to handle these attacks [1]. Cryptography is an art of using mathematics for encrypting and decrypting the data. Today many cryptography algorithms (also called cipher) being used and some algorithms are good and some are not so good. The quality of a cryptographic algorithm is determined by its ability to prevent an unrelated party recovering the plaintext from ciphertext. Cryptosystem is a combination of possible plaintext, ciphertext and possible keys. Encryption and decryption algorithms are associated with each K [2]. Symmetric and asymmetric are the different types of cryptosystems. In symmetric-key cryptosystem, the secret key is used for both encryption and decryption [8]. Even though, symmetric-key encryption is extremely fast and secure, key management is the major issue because it must be shared by both parties before any message is encrypted [5]. The Data Encryption Standard (DES), Advanced Encryption Standard (AES) etc.

Manuscript published on 30 September 2019
* Correspondence Author
Mani. K*, MCA and M.Tech. from the Bharathidasan University, Trichy, India
Devi. A MCA and M.Phil. from Bharathidasan University, Trichy, India

are some famous symmetric-key encryption. In classical DES [5], the keys used are sometime weak keys or semi weak keys. A weak is the one which consists either of all 0's, all 1's or half 0's and half 1's after parity drop operation. Also four out of 2^{56} possible keys are weak keys. A semi weak key creates only two different round keys and each of them is repeated eight times [9]. In DES, there is a possibility of six key pair semi-weak keys. In DES, initially 64-bits DES-key is given as input and in each round a 56-bits round is generated to produce the ciphertext [3]. The Hamming distance between a plaintext M and a ciphertext C denoted as H(M,C), produced by DES may be less if DES keys are weak keys and the eavesdropper may easily recover M from C [4]. To overcome these, a U-matrix based keystream is generated in this paper. Once the K_s is generated, keys are taken from it by choosing the starting position which is determined by both sender and receiver. To generate the U-matrix, the size of the U-matrix is given as input which is also known by both sender and receiver. Further, different key is generated for each round from U-matrix based K_s which differs from classical DES in which only one key is used in all rounds. The rest of the section in this paper is organized as follows. Related work is presented in section 2. Section 3 presents the mathematical background for U-matrix. The generation of U-matrix with an example is discussed in section 4. The proposed methodology for generating keystream with an example is discussed in section 5. The experimental result is presented in section 6. Finally, section 7 ends with conclusion.

II. RELATED WORK

JIANG Hong and et.al.[15], they have made comparison between LKS (LAN-based Key Server) and KSP (Key Selection Protocol) and after this the GKSP (Group based Key Selection Protocol) is used to enhance the security of large Ethernet networks. Each MKA (MAC Security Key Agreement message) contains the status of a node and the key server is identified. DES algorithm as described by Davis R [11] takes a fixed length of string in plaintext bits and transforms it through a series of operations into cipher text bit string of the same length and its each block is 64 bits. Sekar, S. Radhika, K. Anand [12] proposed a new innovative method to enhance the AES algorithm by increasing the key length to 512 bits and the number of rounds increased to provide a strong encryption method for secure communication. Code optimization is performed to improve the speed of encryption/decryption using the 512-bit AES.

This method is not modifying the structure of AES but only increase the number of rounds so that the attacks need the same key are still serious to this algorithm and also this algorithm increase the processing time which will limit the use of AES in real applications. Fauzan Mirza [13] gives a basic introduction to block cipher design and analysis and also the concepts and design principles of block ciphers are explained, particularly the class of block ciphers known as Feistel ciphers.

Paul C. Kocher [14] explains how attackers may be able to find fixed Diffe-Hellman exponents, factor RSA keys, and break other cryptosystems by carefully measuring the amount of time required to perform private key operations.

III. Mathematical Background

The following definitions and theorems are useful in generating the U-matrix.

A. Definition (Unit Orthogonal Matrix U-matrix)

In U-matrix [10], the term "orthogonal" is used in describing both vectors and matrices. Let V be a row or column matrix with n entries and W be a row or column matrix with n entries so that

$$V = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{bmatrix} \text{ or } \quad V = [V_1\ V_2\ \dots\ V_n] \qquad (1)$$

and

$$W = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_n \end{bmatrix} \text{ or } \quad W = [W_1\ W_2\ \dots\ W_n] \qquad (2)$$

Given V and W, both non-zero we say that V and W are orthogonal if the dot product of V and W is zero i.e., $V.W=V_1W_1+V_2W_2+\dots\dots+V_nW_n=0$ (3)

B. Definition (Quadratic Residue)

Let p be an odd prime and gcd(a, p)=1. If the quadratic congruence $x^2 \equiv a \pmod p$ has a solution, then a is said to be quadratic residue of p. Otherwise a is called quadratic non residue of p. Based on Euler's criterion , let p be an odd prime and gcd(a,p)=1. Then, a is a quadratic residue of p if

$a^{(p-1)/2} \equiv 1 \pmod p$ …(4)

To fill the values in U-matrix (1,-1), the following theorems are used.

Theorem1: If A is nxn Matrix, A is a U-Matrix then n=1,2, or n≡0(mod 4).

If n>2, then A=$[aij]$ has at least three rows so that,

$$\sum_{j=1}^{n}(a_{1j}+a_{2j})(a_{1j}+a_{3j}) = \sum_{j=1}^{n} a_{1j}{}^2 + \sum_{j=1}^{n} a_{2j}a_{1j} + \sum_{j=1}^{n} a_{2j}a_{3j} + \sum_{j=1}^{n} a_{1j}a_{3j}$$

$$= \sum_{j=1}^{n} a_{1j}{}^2 = n \qquad \dots(5)$$

Where three of the sums are zero because different rows of A are orthogonal and when the remaining sum is n because a_{ij} $=\pm 1$.

where,

$$n=\sum_{j=1}^{n}(a_{1j}+a_{2j})(a_{1j}+a_{3j}) \equiv 0(mod\ 4)$$

In order to construct U-matrices,1's and -1's are to be generated. For that Legendre symbol $\left(\frac{b}{p}\right)$ is used and it is defined as

If p is prime, and p > 2, then

$$\left(\frac{b}{p}\right) = \begin{cases} 1 & if\ b\ is\ a\ quadratic\ residue\ of\ p \\ -1 & if\ b\ is\ a\ nonquadratic\ residue\ of\ p \\ 0 & if\ p \mid b \end{cases}$$

Theorem 2: Method of generating U-Matrix

For a prime p such that p≡3 (mod 4) and n=p+1, define the n×n matrix A=[a_{ij}] as follows.

$a_{ij=1}$ if i=0 or j=0

$a_{ij} = \left(\frac{j-i}{p}\right)$ if 1≤ i, j ≤ p and i≠j

$a_{ii=-1}$ if 1≤ i≤p

where the parenthetical expression is the Legendre Symbol. Then, A is a U-matrix.

IV. Generation of U-Matrix-An Example

In order to generate U-matrix of order 20, i.e, 20×20 U-matrix, let p=19 since p≡3(mod 4) and n=20. To fill the values of 20×20 U-matrix, the quadratic residue Q_r and quadratic non residue Q_{nr} of p is found. For that b is computed as b= (p-1)/2=9. Then, find $Q = i^b \bmod p$. If Q=1 then i ϵ Q_r, otherwise i ϵ Q_{nr}.

From table 1, it is noticed that {1,4,5,6,7,9,11,16,17} ϵ Q_r and {2,3,8,10,12,13,14,15,18,19} ϵ Q_{nr}. Once Q_r and Q_{nr} are found, the values in U-matrix are filled either 1 or -1 is filled based on Legendre symbol. For example, to obtain the value of the fifth and sixth rows of U-matrix A=$[a_{ij}]$ i.e., row with i=5 and i=8, theorem 2 is used and the values of U-matrix are shown in table 2. Using the above procedure, 20 × 20 U-matrix is generated and it is shown in fig. 1.

V. Proposed Methodology

It consists of two phases viz., (i) generation of key K_i, i=1, 2…,16 from keystream K_s (ii) generation of DES key and performing encryption and decryption.

A. Generation of Key from Keystream using U-matrix

It is noted that DES is a block cipher encryption in which the size of M and K are taken as 64- bits initially [7]. Also, 16 rounds are performed in producing the C. Moreover, same K is used in all rounds which results in lack of security. To enhance the security, different key is taken from the K_s which is generated using U-matrix. Let W is a set containing 4n-1 elements and there is a U-matrix of dimension 4n×4n. In the U-matrix, first row and first column consisting of only 1's and S is the submatrix obtained by omitting the first row and first column of matrix A. Then, there are 4n-1 subsets of W each containing 2n-1 elements of W. Further, each such W has exactly n-1 elements in common with any other such w.

Concatenate all *w* and the resultant is called Ks. From Ks, keys K_i, i=1, 2,…,16 are generated and they are used for DES encryption. It is noted that the difference between traditional and modified DES is that in traditional DES only one is used in all 16 rounds but in modified DES different keys are used. The keys are taken from U-matrix generated K_s. Based on the selection of stating position of the

Table- I: Quadratic and Quadratic Non residue for p=19

J	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
j-5	14	15	16	17	18	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
(j-5)/p	1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1
j-8	11	12	13	14	15	16	17	18	0	1	2	3	4	5	6	7	8	9	10	11
(j-8)/p	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	-1	-1	1

$$A = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 \\
1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\
1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 \\
1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 \\
1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 \\
1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 \\
1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 \\
1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 \\
1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 \\
1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 \\
1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 \\
1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 \\
1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 \\
1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & 1 \\
1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 \\
1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & -1 \\
1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\
1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
1 & -1 & 1 & 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1
\end{bmatrix}$$

Fig. 1: Generation of 20×20 U-matrix

Table- II. Computation of U-matrix elements for 6th and 9th row

I	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Q	1	18	18	1	1	1	1	18	1	18	1	18	18	18	18	1	1	18	0

K, the modified DES is classified into (i) U-matrix based sequential selection of starting position of the keys from K_s (USDES) (ii) U-matrix based random selection of starting position of the keys from K_s (URDES). In USDES, once starting position of the first round key is initially determined by both sender and receiver (or it is determined based on random number), then the starting position of the next round key is selected immediately followed by the last character (or number or alphanumeric) of the previous. But, in URDES, the starting position of all keys is determined on the 16 random numbers. The following notations are used in generating the K_s:p:prime no; q:order of U-matrix; W:set, w:subset, w ϵ W; l(w):length of w; n(W):number of elements in W; n(w):number of elements in w; n_r: number of repeated elements in w;

L: Length of K_s r_i: ith random number; Spos: starting position in K_s; S_p: Selecting the labels from K_s consecutively starting from Spos till pos+127 is reached; S_{ri}: Selecting the labels from K_s consecutively starting from r_i till r_i+7 is reached. Algorithm 1 is used to retrieve K_i from K_s.

int function generate_key (p, flag)

Algorithm 1: Generation of K from K_s

1. begin{main}
   1. Generate the U-matrix with order q, q← p+1
   2. Compute n, where n is the largest integer such that 4n-1=p
   3. Compute the number of elements of w, w←2n-1
   4. Determine the order of U-matrix, U_{order} ←q ×q

5. Generate U-matrix using theorem 2
6. Assign the set of labels where the label is either alphabets or numerals or alphanumeric on the top of U-matrix
7. Compute the number of rows s←p and generate w for each row of S
8. Form a K_s by concatenating all w of U-matrix
9. Compute $l(K_s) \leftarrow s^* w(n-1)$; i←0; j←0
10. If(flag=0) then
    begin
    Generate a random number r1 < L
    Spos ← r_1
    While (i≤16)
    j ← j+7; Epos ← Spos+j
    If Epos>L, then Epos ← Epos mod L
    K_i=Retrieve the label from K_s(Spos) to K_s(Epos)
    end{if}
    i ← i+1; Spos←Epos+1
11. end{while}
12. If(flag=1) then
    begin
    while(i≤16)
    begin
    Generate a random number r_i< L
    Spos ← r_i ; j ← j+7;
    Epos ← Spos+j
    If Epos>L, then Epos ← Epos mod L
    K_i=Retrieve the label from K_s (Spos) to K_s (Epos)
    end{if}
    i ← i+1
    end{while}
    return K
    end{main}

B. Encryption using DES with U-matrix based Keystream

It is noted that in conventional DES, only one set of 64-bit K is given as input and a 56-bit subkey is generated in each round from it [6]. In modified DES, when the generation of first round key, eight characters (64-bit) are taken from U-matrix based K_s by giving the starting position where the staring position is determined randomly only by both sender and receiver. From the U-matrix based K_s, only 56-bit subkey is generated. For the second round, the starting position is again taken randomly by both sender and receiver and the eight characters of U-matrix based K_s are taken accordingly and the next subkey is generated in similar manner. The process is repeated for all other rounds. Since, in each round a different key is taken from the K_s, subkeys generated from it is also different and the relationship between the subkey generated from the current and previous rounds are not predictable which eventually results in a way of enhancing security.

Table- III. Generation of Keystream from U-matrix

C. Proposed Methodology - An Example

In order to understand the relevance of the work, let p=19 and 4n-1 elements in set W, i.e., 4n-1=19. Then, n=5. Each w has the size 2n-1=9, i.e., 2n-1= (2*5)-1=9 and n-1=4 elements in common with every other W. Since W=19 and the elements taken care say,

W= {H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z}, U-matrix order is 20×20 and S-matrix order is 19×19 . Then, there are 19 subsets of W each containing 9 elements which is shown in table 3. Further, each such w has exactly 4 elements in common with any other such w.

After concatenating all w, the resultant K_s obtained from U-matrix is

ILMNOQSXYJMNOPRTYZHKNOPQSUZHILOPQRT
VIJMPQRSUWJKNQRSTVXK
LORSTUWYLMPSTUVXZHMNQTUVWYINORUVW
XZHJOPSVWXYKPQTWXYZ
HJLQRUXYZHIKMRSVYZHIJLNSTWZHIJKMOTU
XIJKLNPUVZJKLMOQVWZIJ
LMNPRWX.

Let the plaintext to be encrypted is M= "KANNANBA" and flag=1. Let the first random number taken is r_1=3. Then, K chosen from K_s by both sender and receiver is "MNOQSXYJ". To determine the starting position of other K let r_2, r_3,…,r_{15} and r_{16} are 20,32,…,70 and 82 respectively. Correspondingly, K_2, K_3,…,K15 and K_{16} are taken from K_s and they are shown in table 4.

D. Generation of First Round Key from First Key

From table 4, the first key taken is K= MNOQSXYJ. Then, the generation of first round key based on K is shown in table 5. Similar computation can also be performed in generating the key for other rounds.

E. Generation of First Round Ciphertext

M is KANNANBA. Then, using DES algorithm, the steps are as follows.

SNo.	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	subset w
1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	ILMNOQSXY
2	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	JMNOPRTYZ
3	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	HKNOPQSUZ
4	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	HILOPQRTV
5	-1	10	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	IJMPQRSUW
6	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	JKNQRSTVX
7	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	-1	KLORSTUWY
8	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	1	LMPSTUVXZ
9	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	-1	HMNQTUVWY
10	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	1	INORUVWXZ
11	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	-1	HJOPSVWXY
12	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	1	IKPQTWXYZ
13	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	1	HJLQRUXYZ
14	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	1	HIKMRSVYZ
15	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	1	HIJLNSTWZ
16	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	-1	-1	HIJKMOTUX
17	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	1	IJKLNPUVZ
18	-1	-1	1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	1	JKLMOQVWZ
19	-1	1	1	-1	1	1	1	-1	1	-1	1	-1	-1	-1	-1	1	1	-1	-1	IJLMNPRWX

Table- IV. Random K Generation from W

Round	1	2	3	...	15	16
Pos	3	20	32	...	70	82
K	MNOQSXYJ	KNOPQSUZ	PQRTVIJM	...	ZHMNQTUV	WYINORUV

Table- V. Generation of First Round Key

R_1	Ki	$ASCK_i$	$B=(ASCK_i)$	$DESK_i$	K_iPC1	C_0	D_0	After Shifting C_i	After Shifting D_i	R_{Ki}
1	M	77	1001101	10011011	1111111	1111	101	1111	1011	10111
	N	78	1001110	10011101	1000000	1111	1101	1110	1011	110100
	O	79	1001111	10011110	1111	0	1001	0	10	10110
	Q	81	1010001	10100010	1110	0	110	0	1100	10101
	S	83	1010011	10100111	101110	111	0	1111	0	101001
	X	88	1011000	10110000	1100101	1000	111	1	1110	1010
	Y	89	1011001	10110011	1000000	1110	111	1101	1110	11001

	J	74	1001010	10010100	1110111					11111

R_1 – 1st round K_1 – Key for R_1 $ASCK_i$ - ASCII value for K_i $DESK_i$ – i^{th} DES Key C_1 – Left order bits for 1st round D_1 – Right order bits for 1st round R_{K1} – 1st Round key K_1PC1–Key based on PC1

$(M)_2$: 0100 1011 0100 0001 0100 1110
0100 1110 0100 0001 0100 1110 = 64
0100 0010 0100 0001 bit
After applying the IP, then
$(IP(M))_2$ 1111 1111 0000 0000 0010 1100
: 1001 0011 0000 0000 0000 0000 = 64
0010 1101 0110 1101 bit

After applying DES procedure, the first round C is shown in table 6.

Table- VI. Generation of Ciphertext Using First Round

R_{K1}	L_0	R_0	$E(R_0)$	$f(R_0, R_{K1})$	$L_i = R_0$	Using S-Boxes	$R_1 = L_i \oplus f(R_0, R_{K1})$
010111	1111	0000	100000	1110	0000	1110	0001
110100	1111	0000	000000	1001	0000	1001	0110
010110	0000	0000	000000	1100	0000	1100	1100
010101	0000	0000	000000	1100	0000	1100	1100
101001	0010	0010	000101	0010	0010	0010	0000
001010	1100	1101	011010	0011	1101	0011	1111
011001	1001	0110	101101	1001	0110	1001	0000
011111	0011	1101	011010	0111	1101	0111	0100

R_1 – 1st round K_1 – Key for R_1 $ASCK_i$ - ASCII value for K_i $DESK_i$ – i^{th} DES Key C_1 – Left order bits for 1st round D_1 – Right order bits for 1st round R_{K1} – 1st Round key K_1PC1–Key based on PC1

VI. RESULT AND DISCUSSION

The proposed methodology is implemented in VC++ using Pentium processor with various file sizes. The security level is measured using All Block Cipher (ABC) Universa l Hackman tool which uses dictionary attack. The time taken for encryption and decryption for the three methods viz., DES, USDES, URDES with varying file sizes are shown in table 7.

From table 7, it is observed that time taken for encryption and decryption process for classical DES is less than that of USDES. This is because for USDES, the key is taken from keystream which is generated using U-matrix, the starting position of the first key is determined by both sender and receiver or it is determined based on random number. Once it is determined, the next key is selected immediately after the last character of the previous key. But in the case of URDES, for each round starting position of the key is determined on the basis of random number which results in increasing the encryption time and decryption time. Since each round requires a considerable amount of time, as a result it takes more time than the classical DES, same single subkey is considered for every round, which is then used for encryption. The decryption time for USDES and URDES is also increasing due to the reverse process.

Table- VII. Encryption and Decryption Time for Three Methods

Method	Encryption Time (ms)					Decryption Time (ms)				
	File Size (MB)					File Size (MB)				
	1	2	3	4	5	1	2	3	4	5
DES	2171 2462 2435 2652 2763	3806 2462 2435 2652 2763	5973 2462 2435 2652 2763	8218 2462 2435 2652 2763	10312 2462 2435 2652 2763	2118 2462 2435 2652 2763	3985 2462 2435 2652 2763	5964 2462 2435 2652 2763	8116 2462 2435 2652 2763	10153 2462 2435 2652 2763
USDES	2652	4584	7302	9898	12416	2587	4791	7265	9684	12334
URDES	2763	4865	7502	10081	12886	2795	4940	7558	10236	12741

The memory taken for consecutive file sizes are shown in table 8.

Table- VIII. Memory required for three methods

Method	Memory Taken (M)				
	File Size (MB)				
	1	2	3	4	5
DES	3263	3731	5473	7163	9044
USDES	5398	6217	8987	12019	15185
URDES	5640	6322	9408	12481	15441

The security level produced by ABC Hackman tool is recorded in table 9 and it reveals that security level is increasing for both proposed USDES and URDES than the classical DES.

Table- IX. Security levels by three methods

Method	Security Level (%)					Avg
	File Size(MB)					
	1	2	3	4	5	
DES	80	81	80	81	80	80.4
USDES	95	94	95	95	95	94.8
URDES	95	96	96	95	95	95.4

The average security level is calculated for classical DES, proposed, USDES, and proposed URDES as 80.4%, 94.8% and 95.4% respectively. Among the proposed methods URDES is outperforming than the proposed URDES and classical DES by increasing the security level.

Table-X. Hamming Distance

Method	DES	USDES	URDES
H(M,C)	21	51	57

The H(M,C) for the said methods and it is shown in table 10. In the proposed USDES and URDES methods, the H(M,C) is also increasing when they are compared with classical DES which means that M is not easily recovered from C.

VII. CONCLUSION

U-matrix based K_s have been proposed in this paper and they are used in generating the subkey for each round of DES. The K generation based on proposed USDES and URDES differ from the conventional DES key generation in the sense that in the conventional DES, the same 64-bit K value which is initially accepted as input is used for generating the subkey in all rounds. It is noticed that in each round 64-bit K is taken from the K_s using U-matrix by considering the starting position randomly if USDES is used and for all the rounds

starting position is determined based on r_i, if URDES is used. It provides an additional level of security. In the proposed DES, the H(M,C) is increasing when it is compared with conventional DES. As H(M,C) is increasing, the relationship between M and C may not easily be predictable. The idea used in this chapter is also novel and innovative. It enhances the security too.

REFERENCES

1. Giruka, V.C., et al., 2008. "Security in wireless sensor". Kalita, H.K. and A. Kar, 2009."Wireless sensor networks, Wireless communications and mobile network security analysis", *International Journal of computing*, 8(1): 1- 24.
2. Https://en.wikipedia.org/wiki/Cryptography
3. Nie T, Zhang T. *"A study of DES and blowfish encryption algorithm". TENCON Proceedings in IEEE Region 10 Conference*, 2009. p. 1-4.
4. Dutt I, Paul S. Chaudhuri SN," Implementation of network security using genetic algorithm", *Int. J Adv Res Computer Science Software Eng.*, 2013, 3(2):234–41.
5. Khan S, Shahzad W, Khan FA," Cryptanalysis of surrounded DES using Ant Colony Optimization", *International Conference on Information Science and Applications (ICISA)*, 2010. p. 1–7.
6. Sreelajaa NK, Paib GAV," Stream cipher for binary image encryption using Ant Colony Optimization based key generation", *Journal of Applied Soft Computing*,2012, 12(9):2879–95.
7. Sreelaja NK, Pai GAV, "Swarm intelligence based key generation for stream cipher", *International Journal of Security and Communication Networks*, 2009 Aug, 4:181–94.
8. AbdulHalim MF, Hameed SM.," Binary particle swarm optimization for attacking knapsacks cipher algorithm", Proceedings *of the International Conference on Computer and Communication Engineering*, 2008, 77–81.
9. Charles P, Shari LP," Security in Computing", *3rd ed. Prentice Hall of India*, 2003.
10. James A Anderson and James M. Bell, *Number Theory with Applications*, Prentice-Hall, Inc., New Jersey, 1997.
11. Davis.R, "The Data Encryption Standard in Perspective", *Proceeding of Communication Society magazine*, IEEE, Vol.16, Nov 1978.
12. A. Sekar, S. Radhika, K. Anand, "Secure communication using 512 bit key", *European journal of scientific research*, Vol. ,No.1, pp 61-65,2012.
13. Fauzan Mirza,"Block Ciphers and Cryptanalysis", *PhD Thesis, Department of Mathematics*, Royal Holloway University of London,2001.
14. Paul C. Kocher, "Timing Attacks on Implementations of Diffe-Hellman, RSA, DSS, and Other Systems", *Cryptography Research Inc.*, San Francisco, USA.
15. Jiang Hong, Yu Qing-song, Lu Hui, "Simulation and Analysis of MAC Security Based on NS2" *Multimedia Information Networking and Security*, Volume: 2, p. 502 - 505, 2009.

AUTHORS PROFILE

Mani. K received his MCA and M.Tech. from the Bharathidasan University, Trichy, India in Computer Applications and Advanced Information Technology respectively. Since 1989, he has been with the Department of Computer Science at the Nehru Memorial College, affiliated to Bharathidasan University where he is currently working as an Associate Professor. He completed his PhD in Cryptography with primary emphasis on evolution of framework for enhancing the security and optimizing the run time in cryptographic algorithms. He published and presented around 15 research papers at international journals and conferences.

Devi. A received her MCA and M.Phil. from Bharathidasan University, Trichy, India in Computer Science Applications. During 2004-2016(April), she had been with the Department of Computer Science at the Lowry Memorial College, affiliated to Bangalore university, Karnataka, India where she was working as an Associate Professor. During 1998-2001, She was working as a programmer in different software companies. She is currently working as a Professor in Reva University Karnataka, India. She has submitted her PhD thesis in Compressed Cryptosystem, Bharathidasan University, Trichy, India.