Hybridize Dynamic Symmetric Key Cryptography using LCG

Zeenat Mahmood
PG Research Scholar Department of Computer Science and Engineering, RITS, Bhopal (M.P.)

Anurag Jain
Head of the Department of Computer Science and Engineering RITS, Bhopal (M.P)

Chetan Agrawal
Asst. Professor Department of Computer Science and Engineering RITS, Bhopal (M.P)

ABSTRACT

In the present work, a block cipher is presented which deals with a dynamic symmetric key cryptographic method using substitution and transposition technique. In this work a dynamic secret key is generated using Linear Congruential Generator (LCG). The striking feature of the present work is creation of a new secret key for every pair of encryption and decryption operation, thus secret key is a dynamic key. After encryption and decryption operation the dynamic key is discarded. The proposed cryptography method is a process consisting of two rounds of encryption and two rounds of decryption. The key generation algorithm uses SHA-1 hashing scheme to produce key of size 196 bits. This key is used to encrypt plain text of variable size. Since this is a block cipher method authors divide the variable size plaintext into 'n' fixed size (49 bit) blocks. The notion of dynamic key has evolved from the concept of the one time pad. Different parts of dynamic secret key are used in different places in order to make it harder for cryptanalysis or attacks.

Keywords

Symmetric key cryptography, Encryption, Decryption, Dynamic secret key, Linear Congruential Generator, Hybrid technique, Transposition, Substitution.

1. INTRODUCTION

The field of Cryptography is concerned with transmission of messages in a secure manner. To accomplish this task the original message is translated i.e. encrypted into unreadable message which is sent to the receiver. The recipient performs inverse translation i.e. decryption to get original message. Conventional cryptography can broadly be divided into three categories: stream ciphers, block ciphers and MAC algorithms. Stream ciphers and block ciphers are used to achieve data confidentiality. Hash functions and MAC algorithms are used respectively for integrity and authentication. This paper proposes a scheme for cryptography using shared dynamic key. The plaintext can be of any size i.e. of varying size the key size is 196 bits and as it is a block cipher technique thus the size of the block is 49bit. The rest of the paper is organized as follows: Section 2 gives a brief introduction to the terms and the concept used in the work, Section 3 describes the proposed approach. Section 4 shows the simulation results and performance analysis. Section 5 concludes the work.

2. RELATED THEORIES AND CONCEPTS

2.1 Basic Terminologies

- *Cryptography* deals with creating documents that can be shared secretly over public communication channels
- Cryptographic documents are decrypted with the key associated with encryption, with the knowledge of the sender.
- The word cryptography comes from the Greek words: Krypto (secret) and graphein (write)
- ***Cryptanalysis*** deals with finding the encryption key without the knowledge of the sender.
- ***Cryptology*** deals with cryptography and cryptanalysis
- ***Cryptosystems*** The combination of algorithm, key , and key management functions used to perform cryptographic operations. Cryptosystem is represented by 5 tuples:

  Quintuple (E, D, M, K, C)

  M set of plaintexts

  K set of keys

  C set of ciphertexts

  E set of encryption functions

  e: $M \times K \rightarrow C$

  D set of decryption functions

  d: $C \times K \rightarrow M$

- ***Plaintext*** – A message in its natural format readable by an attacker
- ***Ciphertext*** – Message altered to be unreadable by anyone except the intended recipients

- ***Key*** – Sequence that controls the operation and behaviour of the cryptographic algorithm**.** Keyspace – Total number of possible values of keys in a crypto algorithm. Keys are rules used in algorithms to convert a document into a secret document

Keys are of two types:

- Symmetric
- Asymmetric

A key is symmetric if the same key is used both for encryption and decryption. A key is asymmetric if different keys are used for encryption and decryption

- ***Initialization Vector* –** Random values used with ciphers to ensure no patterns are created during encryption

2.2 Cryptography issues:

- ***Confidentiality***: Only sender, intended receiver should "understand" message contents sender encrypts message receiver decrypts the message.

- ***End-Point Authentication*:** Sender, Receiver want to confirm identity of each other
- ***Message Integrity*:** Sender, Receiver want to ensure message not altered (in transit, or afterwards) without detection.

2.3 Requirements of Symmetric Key Cryptosystem:

The algorithm needs not to be kept secret but the entity needed to keep secret is the key. In conventional cryptography the key must be shared between sender and receiver in a very secret manner.

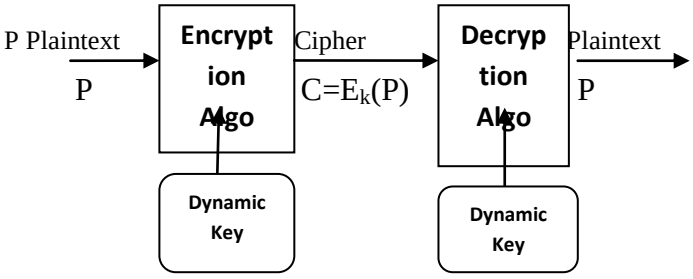

Fig 1: Cryptography

2.4 Dynamic Key

The dynamic key is the new and advance concept in the field of cryptography. These keys are used only once and produced for every pair of encryption and decryption operation and after decryption they are discarded hence the name dynamic key. In order to make cryptanalysis hard different part of keys are applied in different section of process. The dynamic key cryptography is one of the advance technique in cryptography where either a long message is divided in too many part or there are many message in both case each message is encrypted with the help of different parts of key i.e sub keys, these all sub keys are not shared between the both parties but only very few information are shared and on the basis of these information both parties generated the dynamic key. [8][9].

2.5 The Linear Congruential Generator (LCG)

Cryptography often involves solving an equation or a set of equations of one or more variables with coefficient in $\mathbf{Y_n}$. This section shows how to solve equations when the power of each variable is 1 (linear equation) [11].

Let a, b and $Y_0 \in Z_M$. The linear congruential generator (abbreviated as "LCG") with parameters M, a, b and Y_M defines a sequence $(Y_n)_{n>=0}$ in Z_M by

$Y_n = a.Y_{n-1} + b \ (n>0)$

And a sequence $(x_{n)n>=0}$ of pseudorandom numbers in [0,1] by

$$X_n = \frac{Y_n}{M} \quad (n>=0)$$

As the sequence LCG (p, a, b, Y_0) = $(Y_n)_{n>=0}$ is defined by a recursion of order one on a finite set it must be periodic. The longest possible length is M in the case of $b \neq 0$ and M-1 in the case of b=0. Word of computer (e.g. 2^{32}) which reduces the modulo operation.

2.5 One Time Pad

In cryptography, the **one-time pad (OTP)** is a type of encryption which has been proven to be impossible to crack if used correctly. Each bit or character from the plaintext is encrypted by a modular addition with a bit or character from a secret random key (or *pad*) of the same length as the plaintext, resulting in a cipher text. If the key is truly random, as large as or greater than the plaintext, never reused in whole or part, and kept secret, the cipher text will be impossible to decrypt or break without knowing the key [13][14]. Every cryptographic key is only secure for a certain amount of time. In addition, larger keys often require higher computational resources, especially in asymmetric cryptography. In practice, excessively large keys may admit denial of service possibilities whereby adversaries can cause excessive cryptographic processing. However, the security of these algorithms relies on long term shared keys that contradict the original idea of one time pad. However, increasing the cryptographic key size is not always the best solution, since no matter how large the key is, its cryptography is still ultimately breakable [5][1].

3. PROPOSED ALGORITHM

In the present work author proposed a dynamic secret key cryptography method called "Hybridize Dynamic Key Symmetric Cryptography using LCG"**".** In the proposed work encryption of variable size plaintext is performed using a dynamic key of size 196 bits. Once the

cryptographic operation executed the key has been discarded. The concept of dynamic key is based on one time pad.

3.1 Dynamic Key Generation

In the proposed work the dynamic key is generated using Linear Congruential Generator (LCG). User input a text key 'IK' .Minimum size of IK is 6 bits and it can have maximum 14 bits. Depending upon text key size a base value (y_0) is determined from base table i.e table 1. An inbuilt key 'IBK' is concatenated with the key entered by user to produce a matrix of size 14X14.A randomize function (F_r) is used to generate the key. The function F_r involves various matrix operations such as multiplication, shift operation, modulus etc .In which the random number y_0 is added to the final matrix to produce the dynamic key.

3.2 Algorithm

Summarized algorithm:

a. Call DetNB function

% Process to determine number of blocks

b. Call KeyGen function

%dynamic Key Generation

% function

c. Call Encrypt

% Encryption procedure

d. Call Decrypt

%Decryption procedure

Detailed Algorithm

1. User input a key (IK) of size between 6 to 14 char.

a1=length (IK); % MATLAB CODE

% [Ik (Input Key) will be used for

% transposition & substitution]

sk =a1; % size of IK

2. Data is processed in the form of binary digits i.e bits ,

So, size of IK become a1*8.

3. Determine corresponding base value (y_0) of a1 from table 1

Y_0 (1,a1);

4. Calculate Inbuilt key (IBK)

IBK=rand (1,16);

b=a+IK;

5. Calculate y1

Y1=a*y0+b mod m;

%% MATLAB Pseudo Code for matrix % %% operation function (MOP):%

6. A[sk][1]=IK ;
% Text entered by user i.e IK;

7. B[1][sk]=IK ;
%Text entered by user i.e IK;

8. C[14][1]=find(IBK,14,'first');
% First 14 char of IBK

9. D[1][14]=find (IBK,14,'last');
%Last 14 chat of IBK

10. E[14][sk]=A[14][1]*B[1][sk]

11. F[sk][14]=A[sk][1]*D[1][14]

12. G[14][14]= E[14][sk]*F[sk][14]

13. M[14][14]=G[14][14]+y_1

14. DSK1 = M[14][14]

% DSK1 is Dynamic Secret

%This algorithm produces key % of size 1568 bits.

15. Call hash function(HF) to calculate 196 bit key

%196 bits of the key are %extracted by using SHA-1 %hash algorithm

% S=size of variable length plaintext;

% **Process to determine number of %blocks**

16. K= S mod 49;

17. I =49-K; % I=number of padding bits

18. NPT=(PT-K)/49;

%NPT=number of plaintext blocks of size 49 %bits

19. B_{NPT+1}=find (PT, K, 'last') + 'I' number of padding bits;

20. If (K==0)

N=NPT;

% n= total number of blocks

Else

N=NPT+1;

21. For (j=1 to N)

Call process ENT;

%ENT is the encryption process of %one Block

ENT Process % Encryption

Round 1

Encryption of block1

a. T11= Transposition is performed on Plaintext block 1 and IK
b. T12=Perform transpose
c. T13= XOR T2 & DFK1
d. T14= ADD T3 & DFK4
e. Round 1Cipher block 1 (RPC 11) = T4

Encryption of Block 2 to Block n

a. T21=ADD T14 & PlainTextBlock2
b. T22=Transpose T21
c. T23=XOR T22 & DFK2
d. T24=Perform shifting on T23
e. Round 1Cipher block2 (RPC 12)=T24
f. Exit

Round 2

Encryption of block 1

a. H11= XOR Round 1Cipher block 1 & DKL1
b. H12=Transposition of H11 & IK
c. H13=Substitution of H12 & IK
d. Round 2 Partial Cipher Block 1 =H13

Encryption of Block 2 to n

a. H21= XOR Round 1Cipher block 2 & DKL2
b. H22= XOR H21 & Round 2 Partial Cipher Block 1
c. H23=Transposition of IK & H22
d. H24=Substitution of H23
e. Round 2 Partial Cipher Block 2 =H24

Decryption Process-

Round 1

Decryption of block1

a. K11=SUB Round 1Cipher block 1 (RPC 11) & DFK4
b. K12= XOR K11 & DFK1
c. K13= Perform inverse transpose
d. K14=Transposition of IK & K13
e. PlainTextBlock 1 =K14

Decryption of Block 2 to n

a. K21= Shifting of **Round 1Cipher block 1 (RPC 11)**
b. K22=XOR of K21 & DFK2
c. K23=Transpose of K22
d. **K24=SUB K23 & K11**
e. PlainTextBlock 2=K24
f. Exit

Round 2

Decryption of Block 1

a. X11= Substitution of Round 2 Partial Cipher Block 1
b. X12=Inverse Transposition of X11 & IK
c. X13= XOR of DKL1 & X12

Block 2 to n

a. X21=XOR of DKL2 & Round 1Cipher block 2
b. X22= XOR of Round 2 Partial Cipher Block 1
c. X23= Transposition of X22 & IK
d. X24= Substitution of X23
e. Round 2 Partial Cipher Block 2=X24
f. Exit

A	6	7	8	9	10
Y_0	16	15	14	13	12

A	11	10	09	08
Y_0	11	12	13	14

Table I: Base Table

3.3 Function used in Encryption (Reverse will be used for Decryption):

3.3.1 SHIFITNG FUNCTION:

The shifting function used in encryption algorithm is

Cipher Text = mod ((plaintext + 1), 26)

Algebraic description of substitution technique is: The letters A-Z are taken to be the numbers 0-25.The addition operation is modulo 26.

C_t=DK (M_t) = (M_t + DK_t) mod 26

Where $1 \leq t \leq 196$

At the receiver end substitution operation is as follows:

M_t=DK_t(C_t)=(C_t-DK_t) mod 26

3.3.2 TRANSPOSITION METHOD

In the proposed work columnar transposition technique is used twice in order to make cryptanalysis attackers harder. The columnar transposition used in this paper consists of writing the key out as column headers, then writing the message out in successive rows beneath these headers (filling in any spare spaces with nulls), finally, the message is read off in columns, in alphabetical order of the headers. For example suppose we have a key of 'ZEENAT' and a message of 'THIS IS A NEW CRYPTOGRPHIC ALGORITHM'.We start with:

Z	E	E	N	A	T
T	H	I	S	I	S
A	N	E	W	C	R
Y	P	T	O	G	R
A	P	H	I	C	A
L	G	O	R	I	T
H	M	Null	null	null	null
6	3	2	4	1	5

Table II

Then read it off as:

ICGCInull IETHOnull HNPPGM SWOIRnull SRRATnull TAYALH

To decipher it, the recipient has to work out the column lengths by dividing the message length by the key length. Then he can write the message out in columns again and then re-order the columns by reforming the key word.

3.3.3 Functions developed during implementation:

Bin2Str function:

This function is implemented to convert binary data into string form. This function takes the text as the input string and generate the 5 bit ASCII based Binary Code.

Str2Bin:

This function has been implemented to convert string data into data of binary type. This function takes the text as the input string and generate the 5 bit ASCII based Binary Code

GetBlock function:

This function has been implemented to determine number of blocks from given variable size plaintext. Necessary padding is done in order to get block of size 49 bit.

Not function:

In digital logic, an inverter or NOT gate is a logic gate which implements logical negation NOT gate is implemented in this function to get complement of data.

XNOR function:

The XNOR gate is a digital logic gate whose function is the inverse of the exclusive OR (XOR) gate. The two-input version implements logical equality .A HIGH output (1) results if both of the inputs to the gate are the same. One but not both inputs are HIGH (1), a LOW output (0) results.

XOR Gate:

The $\overline{\text{XOR}}$ gate is a digital logic gate that implements an exclusive OR ,that is, a true output (1) results if one, and only one, of the inputs to the gate is true (1). If both inputs are false (0) and both are true (1), A false output (0) results. It represents the inequality function. The XOR gate with inputs *A* and *B* implements the logical expression A.B +A .B

DataHash function:

This is a built in MATLAB function. This function creates a hash value for an input of any type. The type and dimensions of the input are considered as default, such that UINT8 ([0,0]) and UINT16(0) have different hash values. Nested STRUCTs and CELLs are parsed. In this work SHA-1 algorithm is used.

ENCRYPTION PROCESS:

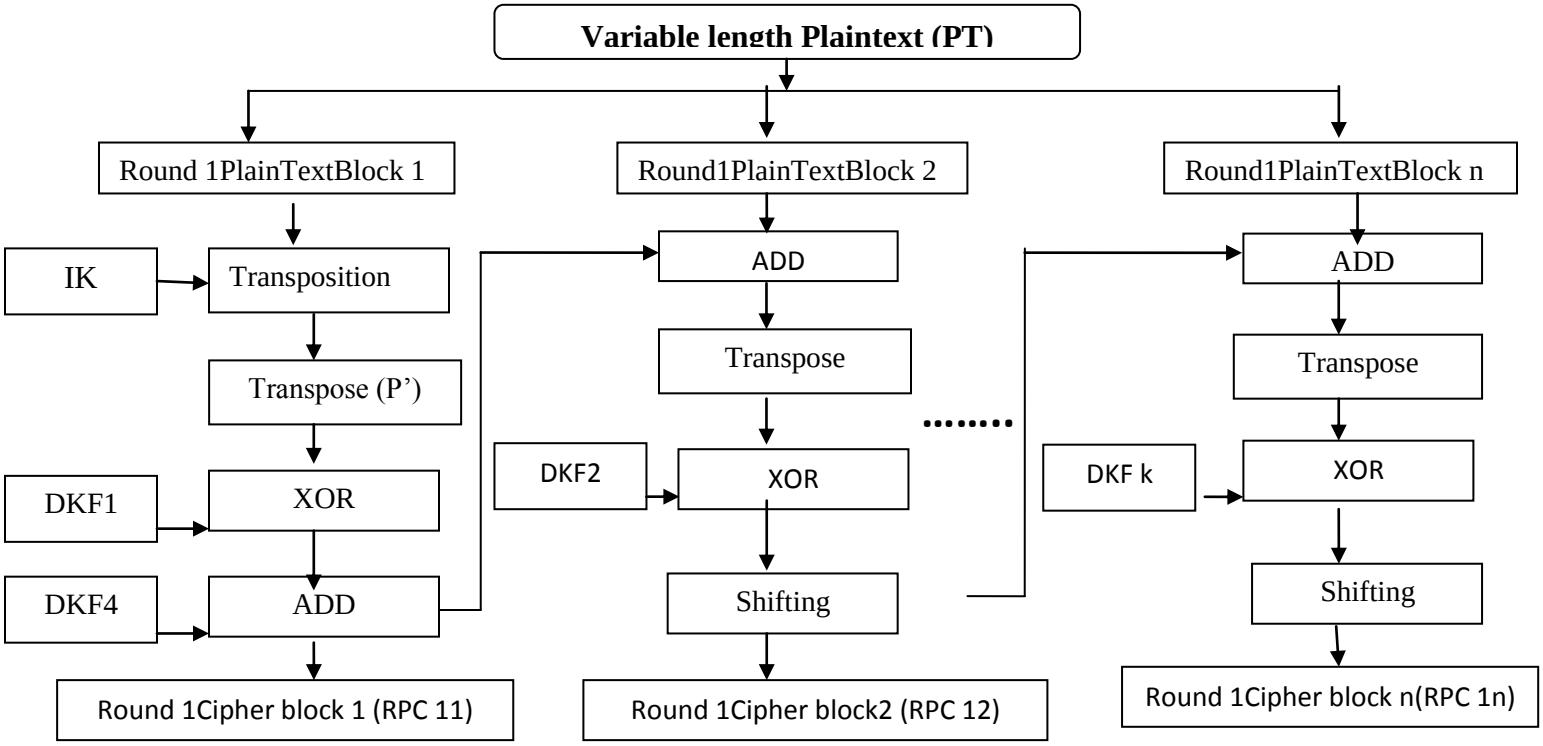

Fig 2: Flow Chart of Encryption round 1

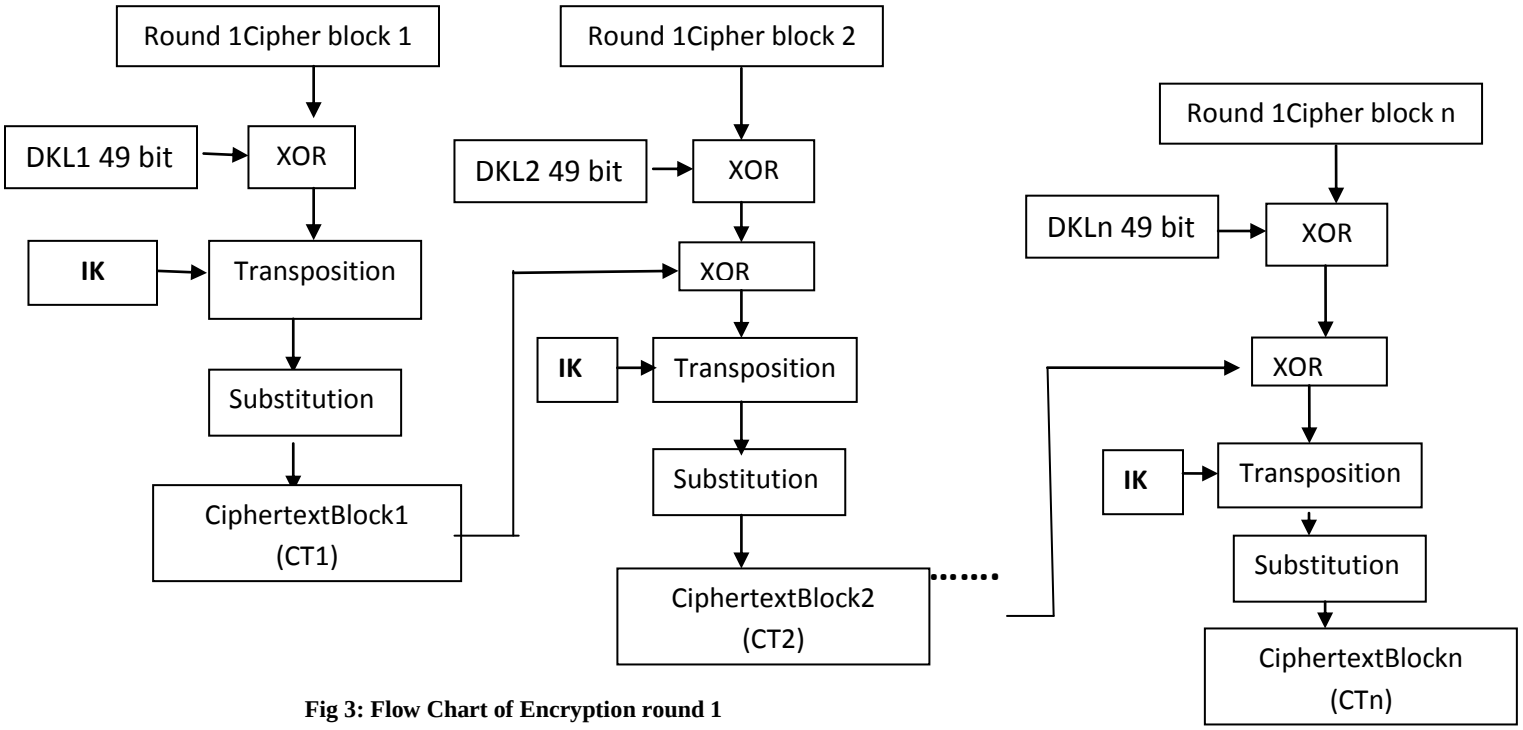

Fig 3: Flow Chart of Encryption round 1

DECRYPTION PROCESS:

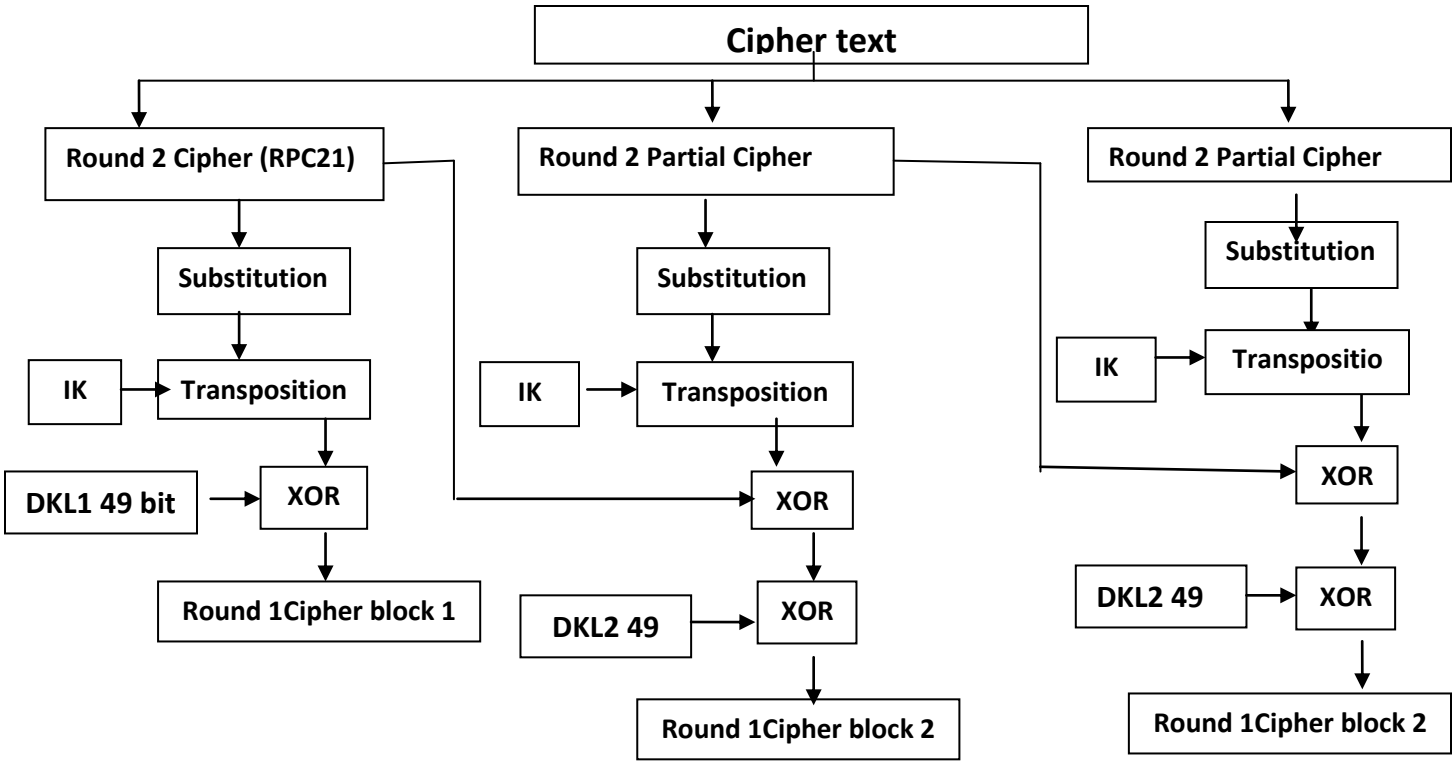

Fig 4: Flow Chart of Decryption round 1

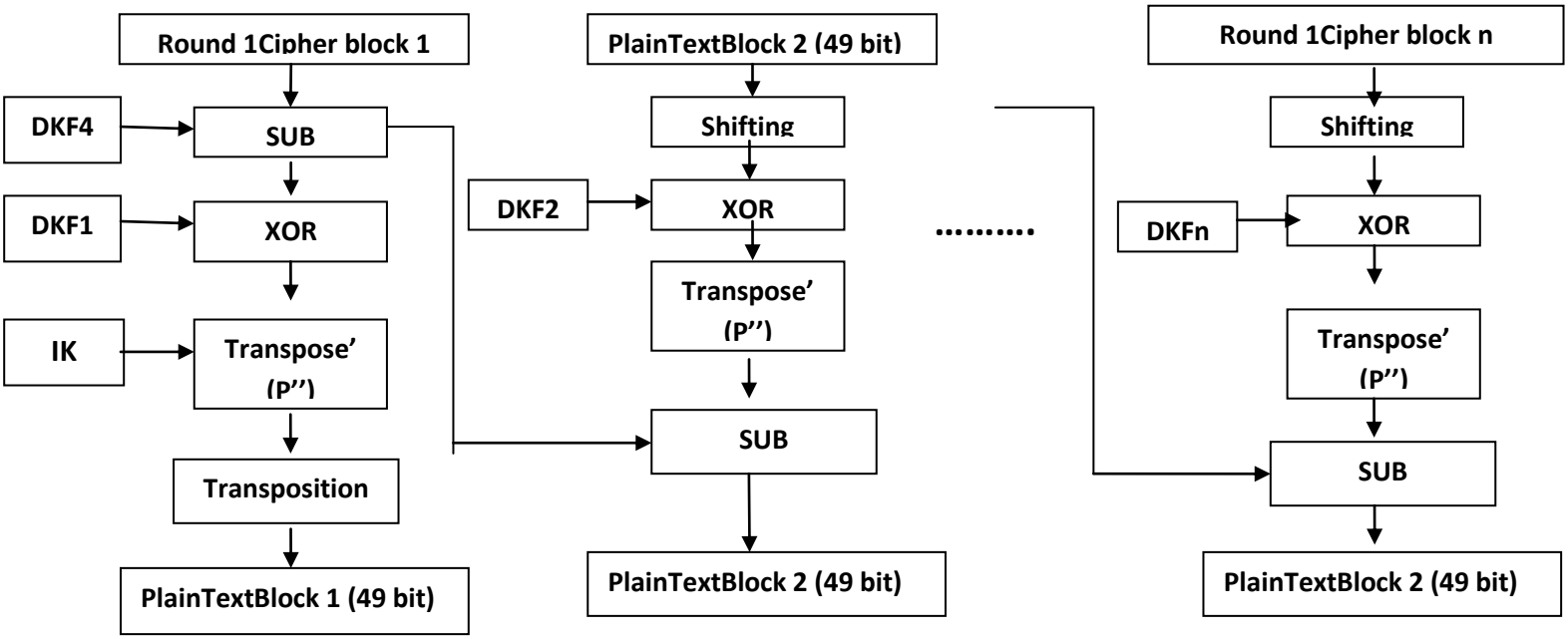

Fig 5: Flow Chart of Decryption round 2

4. SIMULATION RESULT & COMPARATIVE PERFORMANCE ANALYSIS

The striking feature of this algorithm is that the key is dynamic and the encryption and decryption process constitutes of two rounds. Multiple rounds make cryptanalysis even harder. Furthermore the time taken to process an average size plaintext is comparatively very little. However the main merit of the algorithm is little amounting of computational time that one has to spend to encrypt a message.

Table III: Comparison of dynamic key and session key

Issues	Dynamic Key	Session Key
Key Exchange	Once	Every Session
Lifetime	Within a message	Within Session
Key Reusable	No	Yes
Vulnerable under man in middle Attack	No	Yes
From a compromised cryprographic key , adversary can	Decrypt a message	Decrypt all messages in the session
From a compromised key exchange protocol	Cryptographic system is still safe	Cryptographic system and session are vulnerable

Table IV: Analysis of Proposed algorithm

Block Size	Plaintext size	No. of blocks	No. of arithmetic and logical operations
49 bits	90 bits	2	7+8=15
	160 bits	4	7+8*3=31
	256 bits	6	7+8*5=47
	445 bits	10	7+8*9=79
	580 bits	12	7+8*11=95
	840 bits	18	7+8*17=143
	1240 bits	26	7+8*25=207
	1600 bit	33	7+8*32=263

Table V: Comparative Analysis of key size and block size

Algorithm	Key Size (Bits)	Block Size(Bits)
DES	64	64
3DES	192	64
Rijndael	256	128
Blowfish	448	64
Hybridize Dynamic Symmetric Key Cryptography using LCG	196 hashed from 1568 bit long key	49

Table VI: Comparison of Time complexities

Algorithm	Megabytes processed	Time Taken	MB/Second
DES	256	5.998	21.340
3DES	256	6.159	20.783
Rijndael	256	4.196	61.010
Blowfish	256	3.976	64.386
Proposed Algo	256	3.883	65.916

5. CONCLUSION & FUTURE WORK

This is a symmetric key cryptography algorithm which uses the concept of dynamic key & hybrid cipher. Dynamic key is generated by using LCG (Linear Congruential Generator).Variable length plaintext is encrypted by using dynamic key of size 196. This is a block cipher technique. Furthermore the chief merit of the proposed algorithm is that it is almost unbreakable and the time taken to execute the algorithm is comparatively very little.In the future work the algorithm will be tested in front security level are verified in this paper with other well known algorithms. More work on the key size and block size may be accomplished in future.

6. REFERENCES

[1] Zeenat Mahmood , Ashish khare "Symmetric Key Cryptography using Dynamic Key & Linear Congruential Generator (LCG)" , International Journal of Computer Applications Volume 50– No.19, July 2012

[2] Wiliam Stallings," Cryptography and Network" , 3rd edition , Penntice Hall, ISBN 0-13- 111502-2, 2003.

[3] Behrouz A. Forouzan,"Cryptography & Network Security" Tata McGraw Hill, ISBN 13-978-0-07-066046-5.

[4] M. Blaze, W. Diffie, R. L. Rivest, B. Schneier, T. Shimomura, E. Thompson, and M. Wiener," Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security",Report of Ad Hoc Panel of Cryptographers and Computer Scientists, Jan. 1996.

[5] Ayushi, " A Symmetric Key Cryptographic [Algorithm "International Journal of Computer Applications (0975 8887) Volume 1 – No. 15

[6] W. Diffie and M. E. Hellman, "New directions in cryptography, " IEEE Transactions on Information Theory, vol. 22, no. 6, pp. 644-654, 1976

[7] Huy Hoang Ngo, Xianping Wu, Phu Dung Le, Campbell Wilson, and Balasubramaniam Srinivasan "Dynamic Key Cryptography and Applications " International Journal of Network Security, Vol.10, No.3, PP.161{174, May 2010

[8] R. Divya & T. Thirumurugan, "A Novel Dynamic Key Management Scheme Based On Hamming Distance for Wireless Sensor Networks" , International Journal of Scientific & Engineering Research Volume 2, Issue 5, May- 2011,ISSN 2229-5518

[9] Yunpeng Zhang, Fei Zuo, Zhengjun Zhai and Cai Xiaobin. 2008. A New Image Encryption Algorithm Based on Multiple Chaos System. International Symposium on Electronic Commerce and Security. 347-350.

[10] Xukai Zou, Yogesh Karandikar and Elisa Bertino, "A Dynamic key management solution to access hierarchy", International Journal of Network Management 2007; 17: 437- 450

[11] P. Hellekalek "Good random number generators are (not so) easy to find Mathematics and Computers in Simulation "46 (1998) 485±505

[12] Dr. Ranjan Bose and Amitabha Banerjee "IMPLEMENTING SYMMETRIC CRYPTOGRAPHY USING CHAOS FUNCTIONS"

[13] L. Law, A. Menezes, M. Qu, J. Solinas, S. Vanstone, "An effcient protocol for authenticated key agreement," Designs, Codes and Cryptography, vol. 28, no. 2, pp. 119-134, 2003

[14] F. Sun, S. Liu, Z. Li and Z. Lü., 2008. A novel image encryption scheme based on spatial chaos map. Chaos, Solitons and Fractals, 38 (3), 631 – 640.

AUTHOR'S PROFILE

Zeenat Mahmood has completed BE from TIT, Bhopal and pursuing M.Tech from Radharaman Institute of Technology & Science Bhopal. Her research work includes one National paper and three International papers. She is a life time member of ISTE.

Mr. Anurag Jain is an Associate professor and Head of the Department of CSE in Radharaman Institute of Technology & Science Bhopal. He holds M.Tech(IT) from UITBU. Currently he is pursuing PhD from RGPV University. He has organized and attended several national and international conferences. He has published 18 papers in international and 2 papers in national journals. He has also published a book of Basic Computer Engineering. He is life member of CSI (Computer Society of India).

Mr. Chetan Agrawal is Asst. Professor in the dept Department of CSE in Radharaman Institute of Technology & Science, Bhopal. He has completed B.E from BIST, Bhopal and M.Tech from TIEIT, Bhopal.