Block Encryption using Skew tent map and Logistic map
Sangeeta Yadav1 Abhishek Didel2 Nisha Khushwah3
1,2,3 Department of Computer Science & Engineering
1,2,3RTU India

Abstract— Number of cryptosystems has been proposed in recent years that are based on chaotic maps. Nisha Khushwah in 2013 proposed an improved symmetric encryption scheme using one dimensional chaotic map in order to obtain chaotic sequences with better cryptographic features. In this proposed system for encryption/decryption the two different chaotic maps are used with a 128 bit external key. Based on the simulation result, more secure cryptosystem is proposed. As two chaotic maps, one skew tent and one logistic map are used in order to obtain chaotic sequences with better cryptographic features.

Keywords: Cryptography, skew tent map, logistic map, confusion, diffusion, permutation operation.

I. INTRODUCTION

Computer Application and Internet service have been contributing extensively to our life experiences. Internet also allows users to access information, wherever they may be and provides communication services, with or without computer security. This rapid development of internet and communication network increased the risk of theft, unauthorized access, disclosure, disruption, inspection, recording or destruction of proprietary information because of the insecure channel through which two different parties communicate with each other. This led to the development of various techniques for secure communication and adoption of cryptography so that the information can be transmitted in unreadable format.

In the field of cryptography the use of chaotic maps has been very popular in now days. And the cryptography procedure is basically depends on the external factor key that is used to encrypt the plaintext into ciphertext.

Cryptography refers to a suite of algorithms to implement a particular form of encryption and decryption. In an ideal cryptosystem confusion reduces the correlation between the plaintext and ciphertext while diffusion transposes the data located at some co-ordinates of the input block to other co-ordinates of the output block [4]. In recent years, many researchers’ shows there interest in studying the behavior of chaotic systems. An interesting relationship between chaos and cryptography has been developed during last two decades, according to which many properties of chaotic systems such as: periodicity, sensitivity to initial conditions/system parameters, mixing property, deterministic dynamics and structural complexity can be considered analogous to the confusion, diffusion.

For designing of new digital chaotic cryptosystems, logistic map is the most widely used. Baptista uses logistic map in his system in which itterates are generated using the equation:

\[ X_{n+1} \rightarrow f (\lambda, x) = \lambda X_n (1-X_n) \quad \lambda \in [0, 1] \]

Chaotic Cryptography can be classified into two parts, which are analog chaos-based cryptosystems and digital chaos-based cryptosystems. First type of chaotic cryptosystems is based on the chaotic synchronization technique, whereas digital chaotic cryptosystems are based on one or more chaotic maps in such a way that the secret key is either given by the control parameters and the initial conditions or determines those values. The inclusion of more than one 1D maps increases the confusion in the encryption process and results in a more secure cryptosystem due to the fact that more confusion in encryption makes the cryptosystem more secure. We have found that the present cryptosystem is faster than the existing chaotic cryptosystems. We have used only two prototype chaotic maps in the present algorithm however; it can be easily extended to any number of 1D chaotic maps.

The skew tent map and the logistic map are topologically conjugate, and thus the behaviors of the two maps are in this sense identical under iteration. Because of the above advantages of both the maps, here in this project we are using logistic and skew tent map both, in the recently proposed cryptosystem.

The proposed system is a symmetric key block cipher algorithm, in which plaintext is rearranged to form a groups of fixed length i.e. of 64 bits (size of each block). These blocks are encrypted sequentially; one logistic map and one skew tent map are used here for encryption. And 128-bit external secret key determines number of iterations and initial condition for the chaotic maps. The whole process of block by block encryption/decryption of 64-bit block, depend on number of iterations and initial condition and encryption of previous block of plaintext/ciphertext. Detailed step by step procedure of the encryption/decryption of the proposed cryptosystem is explained below.

II. PROPOSED ALGORITHM

Step 1: First for encryption/decryption in this algorithm, we divide plaintext/ciphertext of any size into blocks unit of 64-bits. Plaintext and Ciphertext of any size are rearranged in the block unit of 64 bits Plaintext and ciphertext of n blocks can be represented as:

\[ \text{Plaintext} (P) = P_1, P_2, P_3, \ldots, P_b \quad (2) \]

\[ \text{Ciphertext} (C) = C_1, C_2, C_3, \ldots, C_b \quad (3) \]

Where, the subscript b stands for the block number. P_1, P_2, P_3,.., P_b are plaintext block unit of 64 bits C_1, C_2, C_3,.., C_b are ciphertext block unit of 64 bits

Step 2: Now Secret key of 128-bits is dived in into blocks of 64-bits named as session keys, as using a secret key of 128-bit is long and inconvenient for encryption/decryption. Secret key is in hexadecimal mode, so a 128-bits key will
contain total 32 alphanumeric characters (out of 0 to 9 and A to F). This session key of 64 bit is further sub divided to determine the initial conditions two maps and iteration number, as show below. The secret key (K) is chosen from a 128-bit external binary sequence, and is represented in Fig1.

![Fig. 1: Representation of 128-bit secret key (K)](image)

Where, \(K = (A_1;A_2;B_1;B_2)\),
A1, A2, B1, and B2 are 32-bit blocks.
A = (A1;A2), B = (B1;B2) are 64-bit blocks.

This 128-bit external binary sequence determines the initial condition of two maps (X0 and Y0), as well as the initial iteration number (T0).

**Step 3:** Now, set \(b = 0\), and this 128-bit external binary sequence determines the initial condition of two maps logistic and skew tent maps \((X_0 \text{ and } Y_0)\) respectively. 128-bit key is converted to the valid value range of initial condition of chaotic maps \([0, 1]\) with \(2^{64}\) possible values. And also determine chaotic iteration, for this we set a key-dependent value for \(T_0\).

Block number \(b = 0\)
Initial condition for \(t_1\) and \(t_2\)
\[
X_0 = (0.A \oplus B)_2 \quad (4)
\]
\[
Y_0 = (0.A_1B_2)_2 \quad (5)
\]
Initial iteration number \((T_0)\)
\[
(T_0) = (K_{60}K_{61}K_{62}K_{63}K_{64}K_{65}K_{66}K_{67})_2 \quad (6)
\]
Where, \(\oplus\) is bit-wise exclusive-OR (XOR) operation. \((0.A\oplus B)_2\) represents fraction written in binary mode. It has 64-bit decimal digits which are represented by \((A\oplus B)_2; (0.A_1B_2)_2\) has the similar meaning. \(K(i)\) denotes the \(i^{th}\) bit of \(K\).

In this way, 128-bit key is converted to the valid value range of initial condition of chaotic maps \([0, 1]\) with \(2^{64}\) possible values. Also for the chaotic iteration we set a key-dependent value for \(T_0\).

**Encryption of Plaintext and Decryption of Ciphertext:**
For \(b = 0\)
\[
X_0 = A \oplus B
\]
\[
Y_0 = A_1B_2
\]
\[
T_0 = (K_{60}K_{61}K_{62}K_{63}K_{64}K_{65}K_{66}K_{67})_2
\]
\[
C_b = (A_2B_1)_2
\]
**Step 4:** For, \(b > 0\), \(X_b\), \(Y_b\) and \(T_b\) are updated by (7), (8)and(9), respectively.
\[
b = b + 1
\]
\[
X_b = C_{b-1} \oplus X_{b-1} \oplus (B_2A_1)_2 \quad (7)
\]
\[
Y_b = C_{b-1} \oplus X_{b-1} \quad (8)
\]
\[
T_b = z(P_{b-1}) \oplus T_{b-1} \quad (9)
\]
Where, \(z(\cdot)\) is a bit-wise XOR function between bytes, e.g. \(z(X) = X_{0:7}\oplus X_{8:15}\oplus \ldots\)

**Step 5:** \(X_b\) and \(Y_b\) are updated to the latest status (10) and (11) by iterating the first logistic map with the initial condition \(X_0\) from (7) by \(T\) times and second skew tent map with initial condition \(Y_0\) from (8), just for once.
\[
X_b = t^T(X_b) \quad (10)
\]
\[
Y_b = t^T(Y_b) \quad (11)
\]
**Step 6:** Now \(b^{th}\) plaintext is encrypted by using updated \(X_bY_b\). The updated \(X_b\) and \(Y_b\) are also used to decrypt the \(b^{th}\) ciphertext.
\[
C_b = S(P_b) \oplus C_{b-1} \oplus X_b \oplus Y_b \quad (12)
\]
\[
P_b = S^{-1}(P_b) \oplus C_{b-1} \oplus X_b \oplus Y_b \quad (13)
\]
Where, \(S(\cdot)\) is a permutation operation, formed by two steps, i.e.

1) Byte-wise rotate right operation: - In this step the rotate number is determined by the byte-wise sum modulo the length of bytes in plaintext block and then byte-wise rotation is performed on plaintext block. For example, \(\cdot = (AABBCCDD)_{16}\), which is denoted in hex, the rotate number is \((AA)_{16} + (BB)_{16} + (CC)_{16} + (DD)_{16}\) mod 4. If this result is 2, \(S(\cdot) = (AABBCCDD)_{16}\) is rotated to \((CCDDAABB)_{16}\) [1].

2) Bit exchange operation: - In bit exchange operation, dividing length is determined by the number of non-zero bits in \(A_2B_1\) and then swapping the left part of certain length in the block with the remaining right part.
\[S^{-1}(\cdot)\] is the inverse operation of \(S(\cdot)\), formed by two steps, i.e.

1) Similar bit exchange operation and ,
2) Then byte-wise rotate left operation.

**Step 7:** Repeat the process (i.e. go to step (3)) until the whole plaintext/ciphertext is encrypted or decrypted.

### III. SIMULATION RESULT

For this proposed algorithm the cipher block chaining (CBC) mode is used. A secret key \((K) = (a1b23cd4e5f6abcdedef7890abcdef1234)_{16}\) was taken and for plaintext, a simple .txt (text file) of size 2.5 kb was taken.

**Results are show in above Fig. 2, 3, 4 spikes like modal shows frequency of occurrence of 8-bit value. Proposed cryptosystem shows uneven distribution while Nisha’s system shows flat distribution. This uneven distribution contributes to difficulty of predicting variables.**

![Fig. 2: Distribution of plaintext of a 2.5kb .txt file](image)


A. Confusion effect:

For confusion effect of the proposed cryptosystem, for this first we plot the plaintext \( P = \) “Formatting Numbers with C++ Output Stream” and the ciphertext \( E(P, K) \) generated by different cryptosystems which is shown in below Fig. 5, 6, 7.

Fig. 2 represents a histogram showing the frequency of occurrence of byte-value in Plaintext. While, Fig. 3 and 4, represent the frequency of occurrence of byte-value in cipher text generated by Nisha Khuswah’s cryptosystem and proposed cryptosystem, respectively.

Plaintext and ciphertext generated by the both cryptosystem are totally different both in byte-value and number of occurrence of byte value. Fig. 5 shows confusion effect clearly in cipher generated by both the cryptosystems.

B. Diffusion Effect

Now we change first character of plaintext from ‘F’ to ‘T’, let \( P_0 \) be the new plaintext. On changing the first character of the plaintext there will be always be a different ciphertext because of the cipher block chaining (CBC) mode used in both systems. The diffusion effects when plaintext is changed are demonstrated from Fig. 8, 9, 10. On changing the plaintext there is slight variation in the plot of plaintext \( P_0 \) shown in Fig. 8 which is not even visible, but due to this change a large variation is observed in the ciphertext shown in Fig. 9, 10.
Then, we change the secret key by changing one hexadecimal number of \( K \) from 8 to F; here we are replacing ‘a’ with ‘f’

\[
\begin{align*}
(K) &= \text{('a'1b2c3d4e5f6abcdef7890abcdef1234)}_{16} \\
K_0 &= \text{('f'1b2c3d4e5f6abcdef7890abcdef1234)}_{16}.
\end{align*}
\]

And let the new key be

The diffusion effect shows when slightly change in secret key from fig. 11, 12. Both cryptosystems shows sensitivity to the secret key \( k \), but on changing the secret key slightly, NishaKhuswah’s cryptosystem, show even distribution. Fig. 11 reveals that a little change in secret key leads to significant difference in ciphertext. In proposed cryptosystem, distribution and difference of ciphertext are uneven therefore when there is slight change in secret key, major difference in ciphertext is observed in Fig. 12.

**IV. SECURITY ANALYSIS**

Nisha Khushwah’s schemes eliminated all the existing weaknesses of the cipher based on Xiang’s scheme. The proposed cipher is based on Nisha Khushwah’s schemes, so all the advantages of the Nisha Khushwah’s system are kept and problem of even distribution of chaotic variable is removed.

By expanding the block size of plaintext/ciphertext and the precision of chaotic variable to 64 bits, provides much larger space \( (2^{64}) \) for the plaintext/ciphertext in a block as well as the initial condition of the chaotic map, while in 8-bit block size and only 256 possible values for \( X_0 \) in the original schemes, this improvement gets rid of the brute-force attack that can serve as a foundation for further cryptanalysis. Initial conditions of the two maps \( X_0 \) and \( Y_0 \) are determined by key dependent transformations so they cannot be recovered by an adversary [13]. Many chaotic cryptosystems do not possess any confusion or diffusion operation within the block. Permutation scheme into the plaintext block is included in the system to provide further confusion or diffusion operation within the block. \( S \) (*) is both plaintext and key dependent permutation operation in our proposed algorithm.

In our proposed scheme, the one logistic map and one skew tent map are adopted for the chaotic iteration which is also key dependent, whose initial values are related to the key and are adopted for the encryption and decryption of the plain text, for the elimination of the even distribution and difference of ciphertext present in the Nisha Khuswah’s scheme that causes only significant difference in cipher text that may cause the easy prediction of chaotic variable by the attackers. Logistic map shows uneven distribution of chaotic variable and remove the problem of easy prediction. And Skew tent map have the uniform invariant density. In this improved scheme sensitivity to the secret key and uneven distribution of chaotic variable generate a ciphertext, in which predicting a chaotic variable is quite impossible and therefore, risk of plaintext attack is removed. Hence, security level is further enhanced in proposed scheme by using the advantages of both the maps. The performance of the cryptosystem is improved according to the observation made from the simulation result and security analysis. While selecting the secret key practically, one should take care that not to take a secret key whose four parts \( A_1, \ A_2, \ B_1, \ \text{and} \ B_2 \)
are exactly identical. If we take four parts $A_1, A_2, B_1, $ and $B_2$ exactly identical, then initial conditions of chaotic maps becomes zeroes and then under this condition no valid chaotic iterations exist.

V. CONCLUSION
On the basis of the improved chaotic cryptosystem with external key by Nisha Kushwah, we have proposed the security of the cryptosystem in this scheme by using the advantages of both maps i.e., Logistic map and Skew tent map. A generalized description of Nisha Kushwah’s cryptosystems is given here and their weaknesses and also their solution to provide more security. We have notice that both proposed and Nisha Kushwah’s cryptosystem have same size of plaintext and ciphertext and size of ciphertext generated by both systems are similar and their encryption time are also same. Based on the above analyses, more secure cryptosystem is proposed. For this secure scheme, one logistic and one skew tent map are used instead of two logistic maps in order to obtain chaotic sequences with improved cryptographic feature. All these advantages make this more secure cryptosystem for the use of information transmission over insecure channel and secure application.

REFERENCES
[12] Alvarez G and Shujun Li. “Some Basic Cryptographic Requirements for Chaos-Based Cryptosystems”.