**Transposition ciphers using Fibonacci series**

**SRINIVAS KOPPU SIR**

**AKSHAY BHANDARI & SHIVANI KAKDE**

**ABSTRACT**

Data is now the most important aspect in todays world, it as become an asset in many cases and is being used to understand human behaviour. In a place where we can break down most aspects of humans, we need to keep this information secure. There have been a lot of advancements in keeping data safe, but while sending and receiving data it may get intercepted, to make sure unethical hackers do not make sense of this information we can at least encrypt it. The top rated encryption algorithms are AES, DES, blowfish, twofish and RSA are now great at encrypting data, but most of them have a large execution time and are too complex for even simple data.

Thus we plan to, in this project, Transform a basic Row transposition algorithm into a modified version where we implement Fibonacci series in a simple yet robust and harder to crack algorithm explained later

**INTRODUCTION**

Data needs be secured, but different types of data have different sensitivity, and it does not make sense to use the most complex algorithms for all data. Thus different data of low sensitivity can use more simple algorithms.

We are devising this algorithm as a low complexity algorithm that can be used anywhere so data that isn’t encrypted, is also encrypted.

Our system includes

**Row transposition –** it is a method where the message to be encrypted (called plain text) is written down letter by letter in a predefined matrix size column wise, after which the columns shift technique of matrices are applied to make sure diffusion and random shuffling occurs.

**Fibonacci series –** A series of numbers where the seed values (first and second terms are 0 and 1) and the next term is the sum of previous two terms i.e. F(n+2)=F(n)+F(n+1). There are many interesting finds in this series that is why it is very largely regarded:

1. Two consecutive numbers in this series are co-prime
2. The ratio of any two consecutive numbers in this series is approximately 1.6

And many more

Thus we plan to mix these two concepts in a way that will make the row transposition matrix into a better version than the original while keeping the simplicity the same

The technique will ensure that the data that is sent will be encrypted to a level where it will be hard for the hacker to decipher back to plain text

**PROPOSED SYSTEM:**

The proposed system is heavily dependent on both Fibonacci series and the basic algorithm of row transposition matrix.

Key generation:

We will initially have the plain text written row wise in a predefined matrix, lets say 3X3 for an example. Once this is done, a key will be generated from the Fibonacci series,

The format of the key:

* [NO OF TERMS | START TERM | DIMENSION OF THE MATRIX]
* So for example if we write : 643
* It means we will be using 6 terms starting with 4th term of the series and the matrix we use is of dimension 3X3
* Therefore for encryption we use derived key 3 5 8 13 21 34
* Now to make sure these numbers can be used for transposition in a 3X3 matrix we need to make sure they are a number less than or equal to 3 thus we will take the modulus of each term.
* New derived key – 0 2 2 1 0 1
* As we can see 0 -refers to first column 1- to the first column and 2- to the second column
* Now we will take these numbers in pairs so
* New derived key – 02 21 01
* The intuition of this key is column shifting this means exchange the first and third column, then in the new matrix exchange third and second column then exchange first and second column.

**KEY GENERATION ALGORTIHM:**

The key will be generated in 3 steps:

The first part k1, will be a **NONZERO** random number greater than 2 generated by the function RAND function in MATLAB and other softwares. (LENGTH OF k1 is L1)

The second part k2, will also be **NONZERO** random number which will hold the number of terms of Fibonacci to be taken, this will also be generated using the random function **RAND** of softwares (LENGTH OF K2 is L2)

Both k1 and k2 are integers, non zero. (By L1 and L2 we mean no of digits)

The third part of the key, k3 will be generated using the length of the plain text to be encrypted,

K3=SQUAREROOT(lengthOfPlainText)

In case of non perfect squares, the ceiling value of the root is taken, thus

K3=CEIL(SQRT(LengthOfPlainText)).

To make K3 more random, we take average of L1 and L2 and use the ceiling function again

K3=CEIL(AVG(L1,L2))

The final key which is sent as L1L2L3 and K1K2K3 to receivers side as public key.

Public key implies that the user (receivers side) and the (senders side) use the same key for encryption and decryption.

The key generation has only basic computation (addition and subtraction and one string function length) thus it is computationally inexpensive for this algorithm which we will try to prove later in this paper.

**ENCRYPTION:**

For example we use – a sample matrix – [ 1 2 3;4 5 6;7 8 9]

In this if we perform the above operations we get [1 3 2; 4 6 5;7 9 8]

Now this matrix is used to written column wise (while writing it is written row wise) so we get – CIPHER – 147369258

PLAIN – 123456789

* In case of odd number of terms, we ignore the last digit so as to make pairs.

**DECRYPTION:**

While retrieving the plain text from cipher text we will key again, we write down the derived key and using string techniques we will reverse the entire text and redo the entire proves in reverse order as we need to follow the exact same sequence in reverse order.

So for example we get the key:

* 643
* We derive the derived key by writing 6 terms of Fibonacci series starting from 4th term.
* Thus we get 3 5 8 13 21 34,
* Now we modulus this series by 3 as our matrix dimension is of size 3.
* Now we get 0 2 2 1 0 1
* Next we reverse the order of this sequence 1 0 1 2 2 0
* Now we create the pairs 10 12 20
* And using each pair do the same column transform as before.
* After all three pairs of transforms are done we write down each element of the final matrix (ROW WISE) ad we can see the original text back!

**LITERARY SURVEY**

Previous works have done on using Fibonacci series for encryption, one by Manoj Mukherjee and Debabrata Samanta enlightens us on using this series for image encryption, the basic idea is insertion of Fibonacci series with the plain text (addition) and then inserting this new series in the diagonal of a random matrix, which is then used to put into the picture elements of the original image. [1]. Another work explains how they use Fibonacci series along with the plain text in unicode characters and this new cipher text is then embedded into an image which is in turn is shuffled randomly Here three numbers are assigned representing each Unicode symbol. This is done by using logic where the three numbers represent the RGB values which forms a pixel and a matrix. And then we shuffle the matrix to obtain an image in .png format. [2].

Another work involves a triple encryption technique which uses two keys to encrypt by, encrypting using first key, decrypting using second key and third key to encrypt. [4]

Based on the formula C1 = EK1(PT),the first cipher text. 2. Second layer of encryption: C2 = Dk2(EK1(PT)),the second cipher text. 3. Third layer of encryption : C = Ek1(Dk2(EK1(PT)),

A encrypt the plain text P=(p1, p2,p3,… pn )

with using offset rule to Fibonacci numbers or Pell numbers key K1

as sequential order. A gets primary encrypted message C1 and A

sends (the key K1 through secure channel) Pi+Fi =C1 [3]

Another work shows encryption using Fibonacci series to create the first cipher text, this is then converted to Unicode from cipher text, this is further embedded into an image using steganography (embedding or encrypting using images)[5]

[1] Fibonacci Based Text Hiding Using Image Cryptography Manoj Mukherjee and Debabrata Samanta Email: manoj.mukherjee@ymail.com, [debabrata.samanta369@gmail.com](mailto:debabrata.samanta369@gmail.com)

[2] Data Encryption through Fibonacci Sequence and Unicode Characters Prachi Agarwal Navitha Agarwal Richa Saxena

[3] TRIPLE ENCRYPTION OF AFFINE CIPHER AND VIGENERE CIPHER WITH FIBONACCI NUMBERS

[4] “Security of Multiple Encryption by Ralph C. Merkle and Martin E.Hellman-Juy 1981”) Fibonacci numbers or Lucas numbers

[5] Secured Communication through Fibonacci Numbers and Unicode Symbols A. Joseph Raphael, Dr. V. Sundaram