An Improved Algorithmic Implementation of Rail Fence Cipher

Samarth Godara[1], Shakti Kundu[2] and Ravi Kaler[3]

Department of Computer Science & Engineering,
[1,3]DIT University, Dehradun, Uttarakhand - 248009, India
[2]Shivalik College of Engineering, Dehradun, Uttarakhand - 248197, India
samarth.godara@dituniversity.edu.in, shaktikundu@gmail.com,
ravi.kaler@dituniversity.edu.in

Abstract

In today's world, internet is one of the biggest platforms where huge amount of information had been exchanged to fulfil the individual's objective or purpose. Due to this overall scenario, internet has really called for vital security. The increase number of data transfers via electronic means from one system to the other has an extensive growth in the past few years. Before the mode of transfer, there are a number of algorithms available for encrypting data and transportation ciphers. They are widely used along with other ciphers to increase the encryption complexity in order to form a product cipher [1]. The rail fence cipher is a form of transposition cipher which is also referred as zigzag cipher. The working is preceded in a manner that it extracts information via approach in which it is encoded. The same key is used for both encryption and decryption via rail fence cipher which is one of the categories of data encryption algorithm. The number of rows used to breakup data to be encrypted into rows and column arrangement which acts as the key in rail fence cipher [3]. This paper proposed an algorithm to implement rail fence transposition cipher in the time and space complexity of O(n), further we discussed the optimized rail number which gives best randomness in the cipher and a new method has been proposed using rail fence cipher to increase the complexity of the cipher, without increasing the time complexity of the encryption algorithm.

Keywords: *Cipher, decryption, encryption, rail fence, transposition*

1. Introduction and Motivation for Research

In the last few years, internet has evolved drastically. People use internet for things like shopping, money transfer from bank accounts, *etc*., where any kind of unexpected failure is intolerable. This makes the security of data during transmissions a mandatory need. Earlier the information is transferred from one system to another via online mode. This network contains all kinds of users, including those, who want to illegally access or alter the information passed from sender to receiver. Since this information is very sensitive, we don't want it to reach to an unauthorised person. Hence the information is first converted into an encrypted message and then sent to the receiver. People who have the decryption algorithm along with the decryption key are the only authorized people who are able to change that cipher into the readable form. Being an authorised person means, that user, by sender's will, has the key to decrypt the message into readable format.

Since communication has been a subject of interest for age. With a vast expansion of internet, massive data information is travelling day by day. So security of data is very important subject in this matter during the transmission of data over internet. It becomes vulnerable to unauthorized attacks by hackers or illegitimate users. Intended receiver

Received (January 5, 2018), Review Result (March 7, 2018), Accepted (March 12, 2018)

ISSN: 2233-7857 IJFGCN

should obtain the information which maintains its confidentiality, integrity, availability and authenticity.

- Confidentiality implies that information should not be access by unauthorized user.
- Integrity implies that Information should be complete and should not be altered by hacker.
- Availability implies that information should be available to its legitimate user.
- Authenticity ensures that only the authorized participants can involve in communication.

Cryptography provides the technique to maintain the above properties of information. It is the art and science of achieving security by encoding message to make them non readable [4]. In addition, it is a technique of decoding message from its non-readable format back to readable format without knowing who they were initially converted from readable format to non-readable format [4].

Basic terminologies used in cryptography are as following:

- Plain Text synonymously is an original message that the sender want to transmit to the receiver.
- Encryption is a technique use for converting the plain text into encoded (Cipher) text.
- Cipher Text is a text obtained after applying the encryption on the plain text.
- Decryption is a technique to convert the cipher text into plain text.

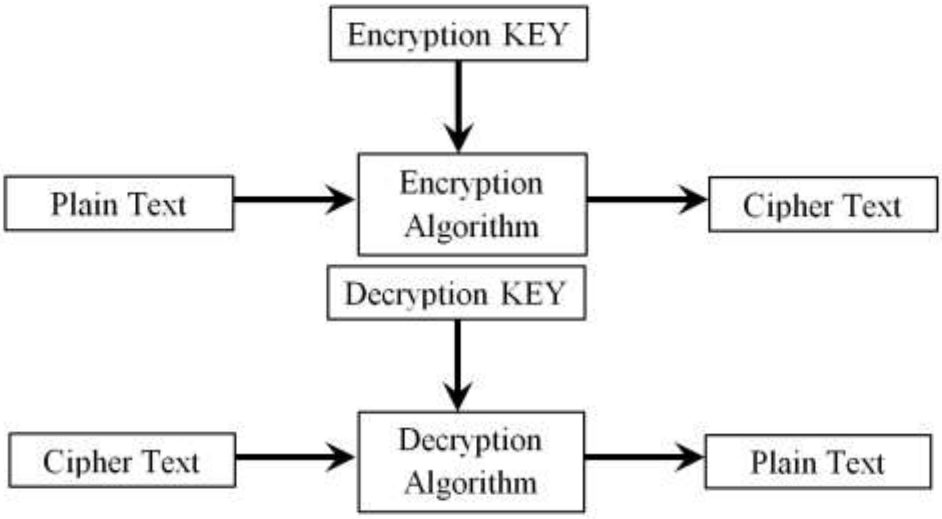


Figure 1. Encryption and Decryption Mechanism

The conversion of plain message into cipher text is done in such a way, that there exists a method to reconvert the cipher into plain or readable message. The field which deals with secure communication between authorised users, in the presence of third party is called Cryptography. A ciphered text is not an unbreakable text and there are possibilities that any third party can break it down to understand it, all this is the matter of the complexity of the cipher and the key. To make the cipher text more complex to breakdown, it can be encrypted many times with same or different algorithm [2].

The process of converting the plain or understandable message into a cipher text with a key, using some algorithm is known as Encryption. And the reverse process, to change the cipher text into plain text is called Decryption [4] as shown in Figure 1.

There are two primary ways to encrypt a message: Substitution Cipher and Transposition Cipher as highlighted in Figure 2 and Figure 3. Substitution ciphers are the ciphers generated by replacing the characters of the original message with some other characters generated by some mathematical function whereas the Transposition ciphers are those ciphers which are generated by interchanging the positions of the characters of the original message to make it an unintelligible message [5].

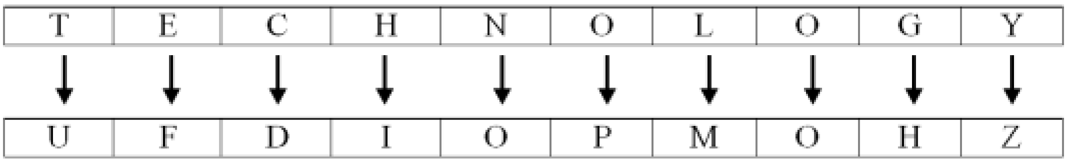


Figure 2. Encryption by Substitution Cipher (Ceaser cipher)

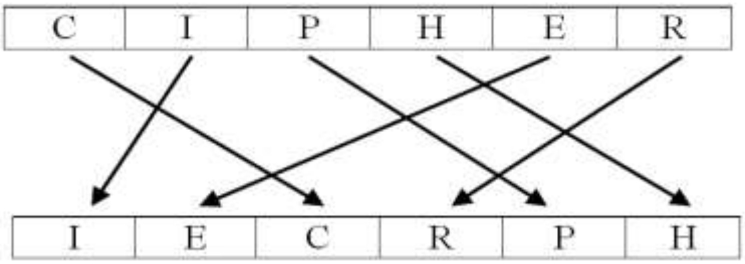


Figure 3. Encryption by Transposition Cipher

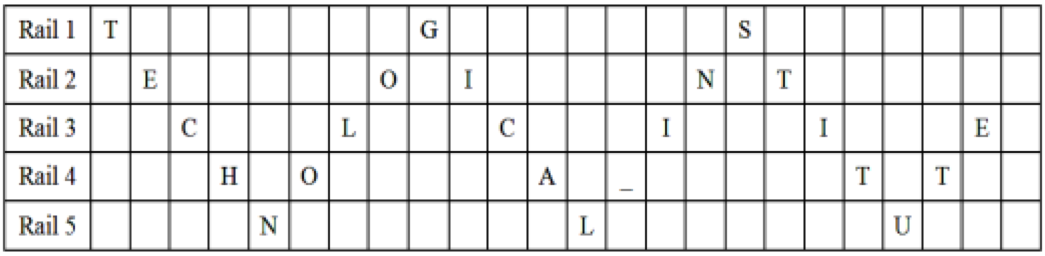
The key plays a vital role in encryption and decryption, that is, why it must be of sufficient length, so that the hacker should not easily guess the key to break the cipher down into readable form. Depending upon the method used for encryption and decryption, the keys can be divided into two categories: Symmetric keys and Asymmetric keys.

Symmetric key system is the system where the key used for decrypting the cipher text is same as the key used for encrypting the plain text. In this system the key must be kept secret or private, that is, why this system is also called as private key system and the system where the key used for decryption is different than the key used for encryption is called as Asymmetric key system. This system is also known as Public Key system [6].

1.1 Rail Fence Cipher

Rail Fence cipher is one of the basic transposition ciphers. The advantage of the Rail Fence cipher over other transposition ciphers like, Sawtooth cipher is that there is a variable distance between consecutive letters. What we mean by variable distance is that the letters need not be arranged in fixed vertical columns that descend, but it can also be arranged in a zig zag manner. Therefore, this increases the difficulty of cracking the code.

In this cipher the plain text is arranged downwards diagonally on successive rails up to the bottom rail and then arranged moving up with the same pattern up to the top most rail, this pattern is repeated until every character of the text gets written on the rails [7]. After arranging the characters in this way, the characters are read from left to right from each rail one by one, starting from the top most rails. For instance, if we want to encrypt the message "TECHNOLOGICAL_INSTITUTE" with 5 rails, then this text must be arranged in the order shown below in Figure 4.



Rail 1	T								G								S							
Rail 2		E						O		I						N		T						
Rail 3			C				L				C				I				I				E	
Rail 4				H		O						A		_						T		T		
Rail 5					N								L								U			

Figure 4. Encryption of Message through Rail Fence Cipher Technique

After the arrangement, the characters are read line by line from left to right to get the cipher text as: TGSEOINTCLCIIEHOA_TTNLU. In rail fence cipher, the key is the number of rails chosen for the encryption. It is to be noted that if the number of rails is greater than the number of characters in the message, then the cipher text will be same as the input message and if we use lesser number of rails then the letters in the cipher text will be placed closer to their actual position, which would make it easier for the third party to guess the plain text, hence the number of rails should be chosen accordingly.

As we can see, the rail fence cipher is being decrypted by arranging it in rows and columns before reading it. Therefore, it is quite an easy and fast process and also it is less prone to mistakes.

One of the problems that the rail fence cipher face is that the security of the code is dependent on the fact that a cryptanalyst does not know the method of encryption. Hence, once the method of encryption is broken, the code is broken already.

Another problem with the rail fence cipher is that it is not very strong. This means that the number of possible solutions is so small that a cryptanalyst can try them all by hand. This makes the rail fence cipher is easy to break as we only have to test all the possible divisors up to half the length of the text. Therefore, we propose a new method of using rail fence cipher called as the block rail fence cipher in order to increase its complexity without increasing the time complexity of the encryption algorithm.

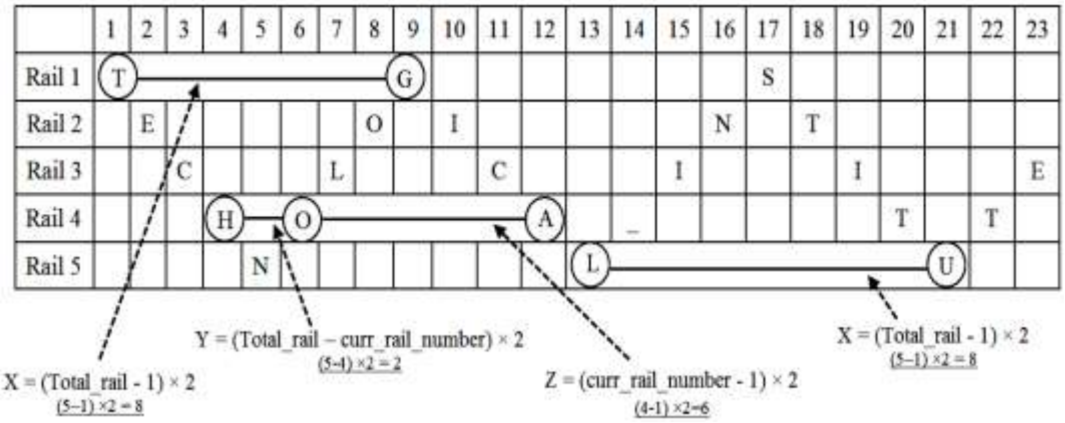
Roadmap: Introduction and Motivation for Research have been discussed in Section 1, Explanation of the proposed algorithm is available in Section 2, Implementation of the proposed algorithm has been highlighted in Section 3, Discussion on the optimized number of rails for encryption is described in Section 4, Block Rail Fence cipher is presented in Section 5 and finally conclusions are provided in Section 6.

2. Rail Fence Cipher Algorithm

The proposed algorithm consists of the steps of instructions needed to be executed in a computer, in the given sequence to solve a specific problem. The objective of proposed algorithm is to encrypt the message in the form of Rail fence cipher so as to reduce the time complexity. The approach of Rail fence cipher algorithm takes 2 inputs, the plain text message which is to be encrypted and the number of rails to be used for encryption, which must be greater than or equal to 2. The overall activity used in rail fence cipher technique is simplest in its respective domain. This feature makes the proposed algorithm popular in transformation of plain text or messages in a very wiser way.

2.1 Explanation of the Proposed Algorithm

The proposed algorithm converts given plain text into rail fence cipher text by generating a permutation of the plain text. The outer loop in the algorithm chooses the rail number, starting from 1, and the inner loop puts the letters into the cipher and computes the place of the letter of the message which will be the subsequent letter in the rail chosen by the outer loop, until all the letters in that rail gets written out. In this way, the algorithm writes the letters of each rail one after another without actually arranging the letters of the message in the zigzag pattern as shown in Figure 5.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Rail 1	T								G								S						
Rail 2		E						O		I						N		T					
Rail 3			C				L				C				I				I				E
Rail 4				H		O						A		_						T		T	
Rail 5					N								L								U		

X = (Total_rail - 1) × 2
(5–1) ×2 = 8

Y = (Total_rail – curr_rail_number) × 2
(5-4) ×2 = 2

Z = (curr_rail_number - 1) × 2
(4-1) ×2=6

X = (Total_rail - 1) × 2
(5–1) ×2 = 8

Figure 5. Calculation of the Placement of Characters in Rail Fence Cipher Text

The proposed algorithm uses three formulas as shown in Figure 3. The formula ‘X’ is used to find the position of the characters present in the first and last rail. The formulas ‘Y’ and ‘Z’ are used to calculate the cipher characters of the intermediate rails.

The Decryption algorithm is just the reverse of it, the letters from the cipher text are chosen one after another, and the same steps are preceded as followed in encryption technique. After that we get the actual position of that letter which gets placed in its actual position. This is done until all the letters from the cipher text are arranged to form the original message.

3. Implementation of the Proposed Algorithm

3.1 The Proposed Encryption Algorithm

Function Rail_Fence_Encryption (plain_text, rail_no)

```
cipher_char ← 1
p_len ← length of plain text
for i ← 1 to rail_no
    flag ← 1
    plain_char ← i
    while plain_char ≤ p_len
        cipher_text[cipher_char]← plain_text[plain_char]
        cipher_char ← cipher_char+1
        if i = 1 or i = rail_no then
            plain_char ← plain_char + (rail_no-1) × 2
        else
            if flag = 1 then
                plain_char ← plain_char + (rail_no - i) × 2
                flag ← 0
            else
                plain_char ← plain_char + ((i -1) ×2)
                flag ← 1
            end if
        end if
    end while loop
```

```
21)     end for loop
```

3.2 The Proposed Decryption Algorithm

```
Function Rail_Fence_Decryption (cipher_text, rail_no)
1)      cipher_char ← 1
2)      c_len ← length of cipher text
3)      for i ← 1 to rail_no
4)          flag ← 1
5)          plain_char ← i
6)          while plain_char ≤ c_len
7)              plain_text[plain_char] ← cipher_text[cipher_char]
8)              cipher_char ← cipher_char+1
9)              if i = 1 or i = rail_no then
10)                 plain_char ← plain_char + (rail_no -1) × 2
11)             else
12)                 if flag = 1 then
13)                     plain_char ← plain_char + (rail_no - i) × 2
14)                     flag ← 0
15)                 else
16)                     plain_char ← plain_char + ((i -1) × 2)
17)                     flag ← 1
18)                 end if
19)             end if
20)         end while loop
21)     end for loop
```

3.3. Time Complexity of the Proposed Algorithm

The time complexity is the computational complexity that measures or estimates the time taken for running an algorithm. Time complexity is commonly estimated by counting the number of elementary operations performed by the algorithm, supposing that an elementary operation takes a fixed amount of time to perform. Thus, the amount of time taken and the number of elementary operations performed by the algorithm differ by at most a constant factor.

The number of times outer loop (line [3]) gets executed in terms of the number of rails,

$$T_1 = r + 1 = O(r)$$

Number of times inner loop (line [6]) gets executed in terms of the number of rails and the number of letters in each rail,

$$T_2 = \sum_{i=1}^{r}(m_i + 1) = n + r = O(n)$$

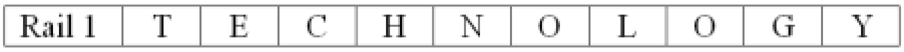
In the above equations, ‘r’ is the number of rails, ‘n’ is the number of characters in plain text and ‘m_i’ is the number of characters in i^{th} rail. The total time complexity of the algorithm depends on the inner loop (line [6]). Hence time complexity of the algorithm can be given by,

$$T(n) = O(n)$$

Hence the proposed algorithm is able to encrypt the message in linear time complexity.

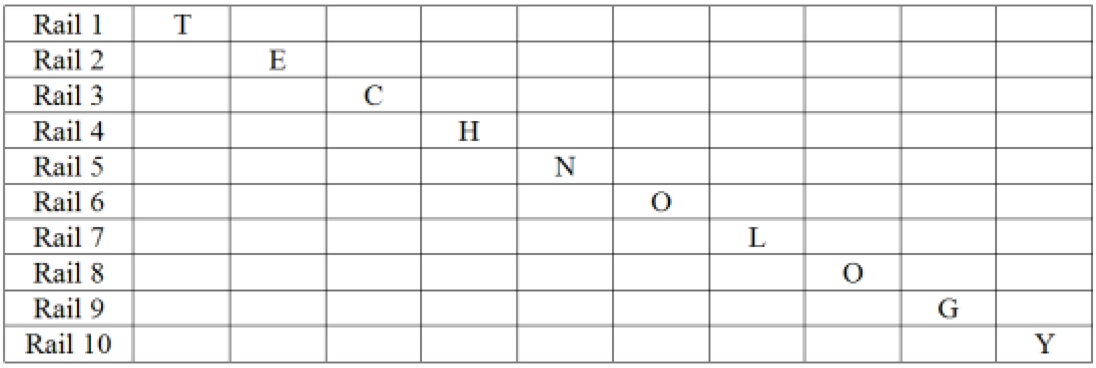
4. Optimizing the Number of Rails

It can be easily observed that if the number of rails is less, say 1, then there won’t be any randomness in the cipher text, in fact if only one rail is chosen for the encryption then the cipher will be same as the message as shown in Figure 6. If the number of rails is large, near or greater than the number of characters in the message then the same problem will arise as described in Figure 7. In order to find the optimal number of rails, we tested all the possible rail numbers from 1 to the message size, and we found that if we choose rails equal to the square root of the length of the message (integer part) then the cipher will have maximum randomness in the placement of the characters as shown in Figure 8.



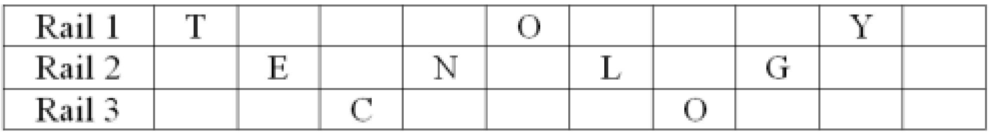
Rail 1	T	E	C	H	N	O	L	O	G	Y

Figure 6. Encryption using one Rail, Cipher: TECHNOLOGY



Rail 1	T									
Rail 2		E								
Rail 3			C							
Rail 4				H						
Rail 5					N					
Rail 6						O				
Rail 7							L			
Rail 8								O		
Rail 9									G	
Rail 10										Y

Figure 7. Encryption using Rails Equal to the Length of Message, Cipher: TECHNOLOGY



Rail 1	T				O				Y	
Rail 2		E		N		L		G		
Rail 3			C				O			

Figure 8. Encryption using $\sqrt{message\_length}$ Rails, Cipher: TOYENLGCO

5. The Block Rail Fence Cipher

Due to the fact that the number of rails have a limited domain, rail fence ciphers are easy to break. In order to increase the complexity of the cipher, we propose a better way to implement the cipher. In our design of 'The Block Rail Fence Cipher', the key consists of a vector of positive integers which indicates the number of characters that are to be encrypted. The rails used for encryption will be equal to the square root of the corresponding integer.

For example, if the key is [12, 4, 23, 7, 16] This indicates that the first 12 characters of the message are encrypted using 3 rails, next 4 characters are encrypted using 2 rails, next 23 characters are encrypted using 4 rails, and so on as shown in Figure 9. This is to be noted that only the integer part of the square root is used.

The block rail fence cipher can be used to encrypt any size of message by dividing it in blocks of equal parts and those parts can be further divided into unequal parts for encryption using different rails. This increases the complexity of the cipher without increasing the time complexity used to encrypt the message. The time complexity of encryption using the block rail fence cipher will be O(n) where 'n' is the number of characters in the message.

This method of encryption uses rail fence cipher as a block cipher and hence it has many advantages. The whole message is not to be encrypted at once. If we use simple rail fence over a block of characters then it is very easy to find out the number of rails used for encryption. In order to guess the key for 'n' characters block, the cipher breaker will have to just try out ‘n’ rails to find out the key whereas in the block rail fence cipher the third party has to go through approximately ‘n^m’ numbers where ‘m’ is the size of the key.

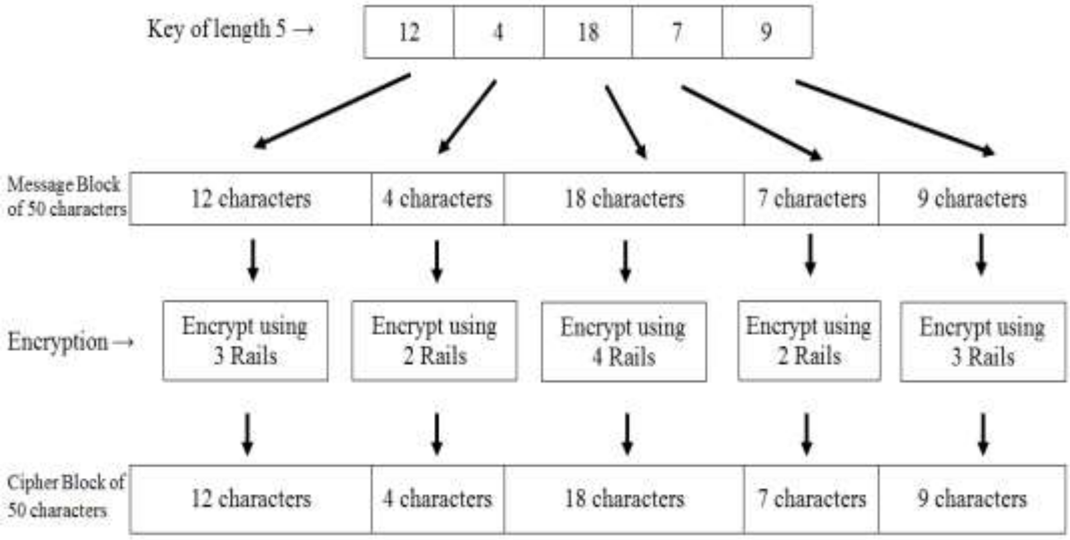


Figure 9. Encryption using the Block Rail Fence Cipher

6. Conclusions

Sending the data securely is a major important task now-a-days. There are many mechanisms which are satisfying the purpose. But still there are lot many chances of cracking the code which is sent to the receiver. Rail Fence Cipher is a cipher system that uses same key for both its encryption and decryption that belongs to the class of symmetric key encryption algorithm. Encryption key depends on the number of rows used to break down the message into row and column arrangement to resemble a rail of fence. The proposed algorithm was designed to show a simple way to develop programs for the encryption and decryption, using any number of rails. The time and space complexity of the proposed algorithm is linear, *i.e.*, O(n). If the number of rails is small number then the placement of cipher characters in cipher text then it will be close to its neighbouring plain text’s characters whereas if the number of rails is greater than the number of characters in message then the cipher text will be same as the specified message. After that we proposed a new method using rail fence cipher called Block Rail Fence Cipher which increases the complexity of the cipher without affecting the time complexity of the encryption.

The Future research direction is to compare the common transposition ciphers to find out the level of randomness in each cipher and also to introduce new methods to increase the complexity of the ciphers.

References

[1] Pearson Instant Learning Series, “Cryptography and Network Security”, Pearson Education India, **(2012)**, pp. 36-37.
[2] D. R. Patel, “Information Security: Theory and Practice”, PHI Learning Pvt. Ltd., **(2008)** April, pp. 48-49.
[3] W. Stallings, “Cryptography and Network Security”, 4th edition, **(2005)** November, pp. 49-50.
[4] A. Kahate, “Cryptography and Network Security”, 3rd edition, McGraw-Hill Education India Pvt. Ltd., **(2013)**, pp. 2-23.
[5] B. A. Forouzan, “Cryptography and Network Security”, Special Indian Edition, Tata McGraw-Hill Education, **(2007)**, pp. 80.
[6] F. Halsall and L. G. Kulkarni, “Computer Networking and the Internet”, 5th edition, Pearson Education India, **(2006)**, pp. 632.

[7] M. Gardner, “Codes, Ciphers and Secret Writing”, Chapter-1, Courier Corporation, **(2013)** April.

[8] W. Diffie and M. E. Hellman, “New Directions in Cryptography”, IEEE Transactions on Information Theory, vol. 22, no. 6, **(1976)** November, pp. 644-654.

[9] H. Gupta and V. K. Sharma, “Multiphase Encryption: A New Concept in Modern Cryptography”, International Journal of Computer Theory and Engineering, vol. 5, no. 4, **(2013)** August, pp. 638-640.

[10] Y. Wang and M. Hu, “Timing - evaluation of the known Cryptographic Algorithms”, Proceedings of International Conference on Computational Intelligence and Security, Beijing, China, **(2009)** December, pp. 233-237.

[11] J. Heath, “A Survey on Secure Email and Private Electronic Data”, Available at: http://www.viacorp.com/crypto.html. Accessed, **(2017)** December 15.

[12] R. Bose, “Information Theory, Coding and Cryptography”, the Tata McGraw Hill Publication, second reprint, **(2008)**, pp. 312-313.

[13] R. C. Merkle and M. E. Hellman, “On the Security of Multiple Encryption”, Department of Electrical Engineering, Stanford, CA published in ACM, A technical note on Programming Technique & Data Structure in Stanford University, vol. 24, no. 7, **(1981)**.

[14] B. Schneier, “Applied Cryptography: Protocols, Algorithms, and Source Code in C”, John Wiley and Sons, second edition, **(1996)**, pp. 757-758.

[15] A. J. Menezes, P. C. V. Oorschot and S. A. Vanstone, “Handbook of Applied Cryptography”, CRC Press, **(1996)**, October, pp. 13.

Authors

Samarth Godara at present working as Assistant Professor in department of Computer Science & Engineering at DIT University, Dehradun, India. He earned his M.Tech. degree in Information Security from Dr. B. R. Ambedkar National Institute of Technology, Jalandhar, Punjab in 2016 and his B.Tech. degree in Computer Science & Engineering from Rajasthan Technical University, Kota, Rajasthan in 2014. His research interests are Network Security, Embedded Systems and Machine Learning.

Shakti Kundu submitted his Ph.D. thesis in Computer Science & Engineering to DIT University, Dehradun, India in November 2017. He earned his M.Tech. degree in Computer Science & Engineering from Guru Jambheshwar University of Science & Technology, Hisar, Haryana, India in 2010. The author current research interests are Information Security and Web Data Mining. He is life member of CSI / ISTE / IAENG / AIRCC / IAEME.

Ravi Kaler at present working as Assistant Professor in department of Computer Science & Engineering at DIT University, Dehradun, India. He earned his M.Tech. degree in Information Security from Motilal Nehru National Institute of Technology, Allahabad, Uttar Pradesh in 2017 and his B.Tech. degree in Computer Science & Engineering from Rajasthan Technical University, Kota, Rajasthan in 2013. His research interests are Network Security, Machine Learning and Algorithm Design.