© 2017 IJSRST | Volume 3 | Issue 1 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Cryptographic Techniques using Binary Tree and Tree Traversal

M Shanmugam Shoba
Senior Assistant Professor, Department of Information Science and Engineering, New Horizon College of Engineering, Bangalore, Karnataka, India

ABSTRACT

In 21st century one of the most emerging problems is Data Security. For secure communication we always use different security algorithms e.g. Caesar cipher, modified Caesar cipher, IDEA, AES, RSA algorithm etc. and in data structures using C we have Linked list, stack ,queue ,tree and graph. In graph we have graph traversal i.e. BFS (Breadth First Search) or DFS (Depth First Search). This paper presents a new idea for Data encryption and decryption in order to provide security for data. Also it provides the new technique for security which is combination of Caesar Cipher and graph traversal, Binary search tree together then security will be much higher than only using Caesar cipher or graph traversal or binary search tree.

Keywords : Cryptography, encryption, decryption, hacker,Binary search tree, Breadth First Search, Caesar cipher, Depth FirstSearch, graph traversal

I. INTRODUCTION

Cryptography is an art of hiding information. A lot of research has been done in the field of cryptography. Using cryptography we can send our information to receivers securely and hackers cannot see this communication if we use good cryptographic algorithms There are various encryption algorithms used for secure data transmission. But still new algorithms are emerging because still we require a better technique for data encryption and decryption.The proposed technique provides a secure way of communication,because it is more difficult to decrypt the information by any unauthorized user.

In this mechanism we are using one of the important part of Data Structures that is Trees. Now let us discuss about trees to understand this paper.Tree is a widely used abstract data type (ADT) or data structure implementing this ADT that simulates a hierarchical tree structure, with a root value and subtrees of children, represented as a set of linked nodes.

For example your family can also be represented in tree like your grandfather will be root your grandfather children's are nodes to your grandfather and you are node to your father like this a family can be represented in the form of a tree.

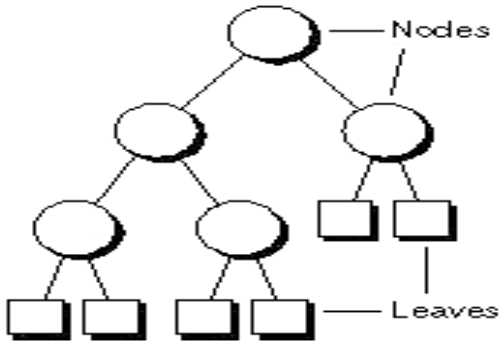


Figure 1: Represents the Tree Data Structure

Tree traversal (also called tree search) could be a sort of graph traversal and refers to the method of visiting (examining and/or updating) every node in an exceedingly tree system, precisely once, in an exceedingly systematic method. Such traversals area unit classified by the order within which the nodes area unit visited.

There are mainly three types of tree traversals they are INORDER, PREORDER, POSTORDER. Order of visiting nodes by In-order is "Left Root Right" by Pre-order is" Root Left Right" by Post-order is "Left Right Root".

These traversals are clearly represented in below diagram.

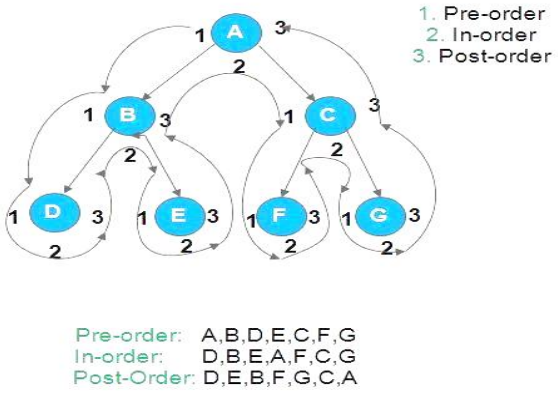


Figure 2 : Represents the Three Tree Traversals

In binary search tree, less than or equal to elements are stored on left hand side and greater elements are stored on right hand side [1].In BFS, we traverse level by level [2]. In cryptography, many algorithms are available for protecting our online important information which we are transferring from one person to another person [3]. One of these algorithms is Caesar cipher. It is very basic algorithm. In Caesar cipher we replace corresponding letters left or right by some number [4]. In DFS, we select some node and go as deep as possible and we use concept called as backtracking here [5]. Using graph traversal we can reach each and every vertex of the respective graph and for this we use BFS and DFS methods.

II. METHODS AND MATERIAL

IMPLEMENTATION

Let us take Given plain-text message is : **evaluation technique**

Step 1: Apply Caesar Cipher Here rule used is position of the character in the alphabet+1.
e.g. Take first character from the word **evaluation** and e has 5th position in the alphabet series. So 5+1=6 and f is the 6th position character in the alphabet series. So our 1st character is f. So using above formula we got words as: **fwbmvbujpo ufdiojrvf**

Step 2: Give numbers to each characters. The TABLE I and TABLE II shows fwbmvbujpo and ufdiojrvf word.

Table I. fwbmvbujpo word

Character	f	w	b	m	v	b	u	j	p	o
Position of characters in the alphabet	6	23	2	13	22	2	21	10	16	15
Position number in the word	1	2	3	4	5	6	7	8	9	10

Table II. ufdiojrvf word

Character	u	f	d	i	o	j	r	v	f
Position of characters in the alphabet series	21	6	4	9	15	10	18	22	6
Position number in the word	1	2	3	4	5	6	7	8	9

Step 3: Arrange the above **position of characters in the alphabet** in ascending order.

Ascending order of fwbmvbujpo word: 2, 2,6,10,13,15,16,21,22,23

Ascending order of ufdiojrvf word: 4, 6,6,9,10,15,18,21,22

Step 4: Divide each word into 2 equal parts according to the numbers given in step 3. Lesser than or equal to numbers will go to left side and greater elements will go to right hand side just like binary search tree [1]. If the word is completely divisible by 2 then half part will go to left side and half part will go right side and if word is not completely divisible by 2 then half part+ 1 character will go to left side, half side will go to right side. Each left and right hand side node has at the most 2 children.
fwbmvbujpo: 10 characters . So 10 characters/2=5 and therefore 5 characters will go left hand side and 5 characters will go to right hand side.

First half part: 2,2,6,10,13,
Next half part: 15, 16,21,22,23
ufdiojrvf : 9 characters. 9 characters/2=4.5 and therefore 5 elements will go to left side, 4 elements will go to right side
First half part: 4, 6, 6, 9,10 **Next half part:** 15,18,21,22

Step 5: Draw diagram for the word position and its numbers. Fig. 3 denotes **fwbmvbujpo** word and Fig. denotes **ufdiojrvf** word**.**

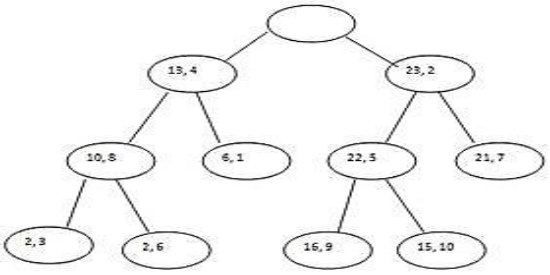


Figure 3. fwbmvbujpo word with its respective word position and its numbers

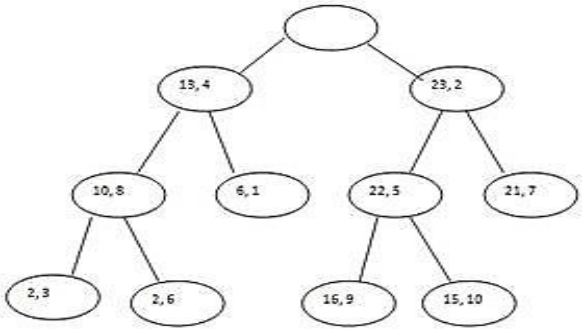


Figure 4. ufdiojrvf word with its respective word position and its numbers

Step 6: Figure out cipher text i.e. write down alphabet number and between each alphabet number, write down position number of the character given in step 2 and if the alphabet number is only a single digit then write down alphabet number first and immediately after that respective number write down position number of the character given in step 2(here 6one). Write down values from each circle from the figure 1 and 2 in such a way that we will get the cipher text as below:

Cipher Text for the words fwbmvbujpo and ufdiojrvf respectively are**:**

1four3_2two3_1eight0_6one_2five2_2seven1_2three_ 2six_ 1nine6_1ten5
1six0_2eight2_9four_6two_2one1_1seven8_6two_4th ree_1f ive5

Decryption:

Step 1: Take the cipher text

Cipher Text:
1four3_2two3_1eight0_6one_2five2_2seven1_2three _2six_ 1nine6_1ten5
1six0_2eight2_9four_6two_2one1_1seven8_6two_4th ree_1f ive5

Step 2: Figure out position of each character in the alphabet series and position number of the character in the respective word from the cipher text. **e.g.** 1four3

Means 13 is the position of characters in the alphabet series and four is the position number in the respective word. Each character in the cipher text is separated by underscore (_). So position of each character in the alphabet series **and** position number in the **fwbmvbujpo** and **ufdiojrvf** words are given in the following TABLE III:

TABLE III. fwbmvbujpo and ufdiojrvf word with its respective word position and its position numbers

Cipher text	1four3	2two3	1eigh t0	6one	2five2	2seven1	2three	2six	1nine6	1ten5
Position of characters in the alphabet	13	23	10	6	22	21	2	2	16	15
Position number in the word	Four	two	eight	one	five	Seven	Three	six	Nine	ten

Cipher text	1six0	2eight2	9four	6two	2one1	1seven8	6two	4three	1five5
Position of characters in the alphabet	10	22	9	6	21	18	6	4	15
Position number in the word	Six	eight	four	two	One	Seven	two	three	five

Step 3: From the chart drawn in step 2 of decryption, we can find out plain text.
e.g. Consider position number in the word and position of characters in the alphabet.

Position number in the word: four **Position of characters in the alphabet:** 13 that means at fourth position, we have 13 number character from the alphabet i.e. m. Similarly, we can find out remaining characters from the respective cipher text which is given in the TABLE IV.

TABLE IV. fwbmvbujpo and ufdiojrvf words with its respective word position and its position numbers

Position number in the word	1	2	3	4	5	6	7	8	9	10
Position of characters in the alphabet	6	23	2	13	22	2	21	10	16	15
Character	f	w	b	m	v	B	U	j	p	o

Position number in the word	1	2	3	4	5	6	7	8	9
Position of characters in the alphabet series	21	6	4	9	15	10	18	22	6
Character	u	f	d	i	O	J	r	v	f

So we got two words: fwbmvbujpo and ufdiojrvf

respectively.

Step 4: Apply Caesar Cipher Here rule used is **position of the character in the alphabet-1** ... (2)

e.g. f has 6th position in the alphabet series. So 6-1=5 and e is the 5th position character in the alphabet series. So our 1st character is e and using this formula we got **plain text** as:

Evaluation Technique

This is our original plain text message. The other technique we are going to propose is mainly divided into two types one is mechanism followed at sender for encryption and another mechanism followed at receiver for decryption.

The steps to be followed during encryption is explained in below flow chart :

First read the text wishing to send identify the length of the sending text identify the required node length and divide the length of text with node size then create the number of nodes based on the outcome from the above operation. Arrange the text in the nodes in the form of BFS mechanism after arranging the text we will get an trees structure then apply the in-order and either pre-order or post-order on the arranged tree. Send the traversal outcomes to the receiver then receiver apply the decryption mechanism.

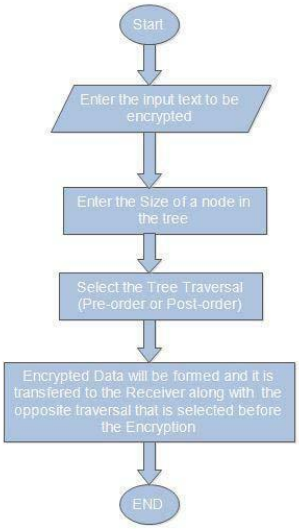


Figure 5 : Steps @ sender for Encryption

This is explained with an example let us consider a text as "Hello World this is the new encryption technique" and node size as "7" means each node can hold 7 character including spaces provided in the text. Tree formed contains 7 nodes since when we divide length on the above string with node size we will get output as 7 i:e No. of Nodes=Abs ((length of string(49)/Node size (7)).

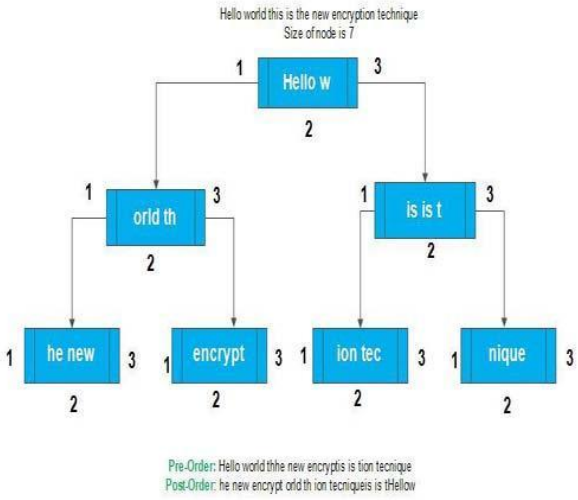


Figure 6 : Steps @ sender for Encryption

Pre-order and post-order is shown in the above diagram now apply in-order traversal to the above then send the in-order and pre-order traversal data to receiver. Receiver can decrypt only when he had in-order and either pre-order or post-order data otherwise no one can decrypt it.

The steps to be followed at reception end by the desired receiver are First receiver have to receive both the in-order and either pre-order or post-order data. After that construct a tree with max 2 child for each node. After forming the tree apply the mechanism of BFS When you apply the BFS for the tree formed from the received traversal data the receiver will get the original data send by the sender.

In this way the reception procedure follows the above procedure is explained with an example in below section along with required flow chart. 3

III. RESULTS AND DISCUSSION

Take the sample example that is taken at sender side or for encryption. i:e "Hello World this is the new encryption technique". So receiver receives in-order and post-order for the above text.

In-order Data: *"he neworld th encrypt Hello W ion tec is is t nique"*

Post-order Data*: "he new encrypt orld th ion techniqueis is t Hellow ".*

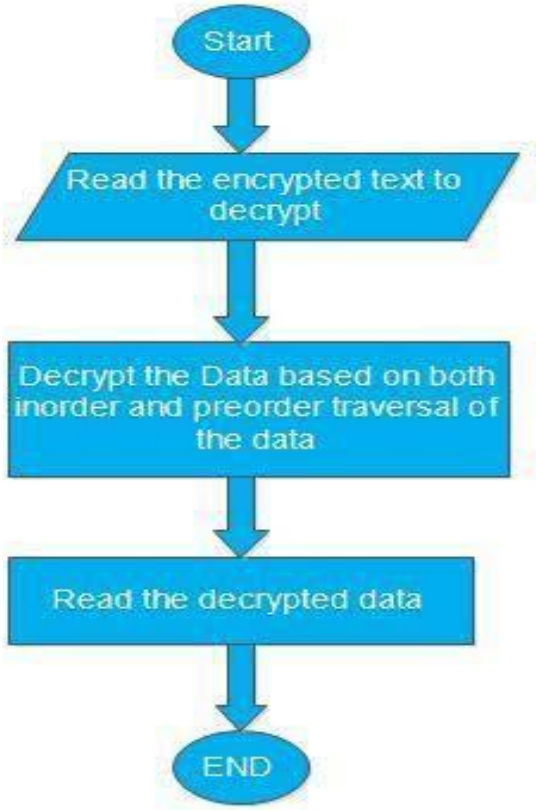
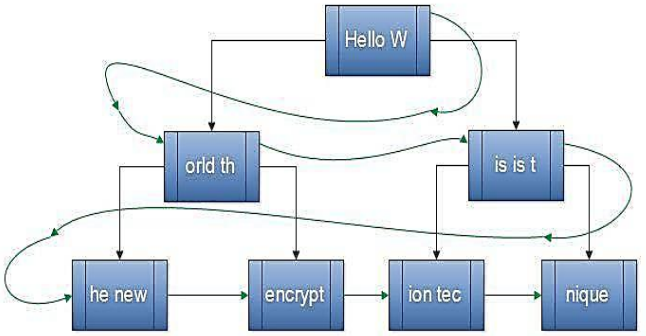


Figure 7. Describes the method to be followed at decryption end

After receiving the data construct a binary tree with the help of in- order and post –order data received after that apply the BFS you will get the required data.

First take the 7 characters from the received post-order data and match the position of selected data in in-order and keep it as root node for the tree that is to be constructed from the root node the left side text is left tree and right side text is right tree.

Follow the procedure till you reach the last node of a tree. In this way a tree is constructed after constructing apply BFS.



For the Above tree if you apply BFS (Berth First Search) we will get the original data as
Hello World this is the new encryption tecnique

a) Advantages

- This mechanism provides more security for the data from hackers means the performance of secure level is more

b) Dis-Advantages

- Memory wastage because of sending the data in two different traversals
- Developing of algorithm for this mechanism is somewhat complex.

IV. CONCLUSION

According to me this the best encryption and decryption techniques using tree traversal mechanism which provides high end security to the data and the development of the mechanism is explained in this paper completely. When we use Caesar cipher with some concepts of data structure using C then result obtained is much harder to detect than only using Caesar cipher or some concepts of data structure using C. We got both plain-text i.e. evaluation technique and cipher text i.e. 1four3_2two3_1eight0_6one_2five2_2seven1_2three_2six_1nine6_1ten51six0_2eight2_9four_6two_2one1_1seven8_6two_4three_1f ive5 using above steps. So security is maintained.

V. REFERENCES

[1]. "A New Approach For Complex Encrypting And Decrypting Data" by Obaida Mohammad Awad Al-Hazaimeh in International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.2, March 2013

[2]. Data Encryption and Decryption using Deterministic Random Key for Transmission: A Review" by Er. Vidiksha1 in JARCSSE Volume 3, Issue 8, August 2013 ISSN: 2277 128X.

[3]. Dragos Trinca, "Sequential and Parallel Cascaded Convolution Encryption with Local Propagation: Toward Future Directions in Cryptography", Proceedings of The third International Conference on information Technology- New Generations. (ITNG'06), 2006, IEEE Computer Society.

[4]. JON L. BENTLEY, M, " Multidimensional Binary Search Trees in Database Applications", IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. SE-5, NO. 4, JULY 1979, PP 333

[5]. Scott Beamer Krste Asanovic David Patterson," Direction-Optimizing Breadth-First Search", SC12, November 10-16, 2012

[6]. Programmer Enas Ismael Imran, Programmer Farah abdulameerabdulkareem, "Enhancement Caesar Cipher for Better Security", IOSR Journal of Computer Engineering (IOSR-JCE), Volume 16, Issue 3, Ver. V (May-Jun. 2014), PP 01

[7]. S G Srikantaswamy and Dr. H D Phaneendra "IMPROVED CAESAR CIPHER WITH RANDOM NUMBER GENERATION TECHNIQUE AND MULTISTAGE ENCRYPTION", International Journal on Cryptography and Information Security (IJCIS), Vol.2, No.4, December 2012, page number 39