ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

Journal of Computational Mathematica

Open Access

Encryption Approach on Graph Theory

Dharani J[1], Maheswari V[2] and Balaji V[3*]
[1,3]PG and Research Department of Mathematics, Sacred Heart College, Tirupattur, Vellore District - 635 601, Tamil Nadu, S.India.
[2]Department of Mathematics, Vels University, Chennai - 600 117.

Abstract

Graph theory is now being a dominant research area along with number theory, which is primary source for cryptography. In particular, we use the concept of graph theory is being used in areas of cryptography in various ways. In this paper, we use spanning tree concept of graph theory to encrypt the message.

Key words: Public key, cryptography, graphs, encryption, network security.

1.Introduction

Definition 1.1 ***Weighted graph*** is a graph in which each branch is given a numerical weight. A weighted graph is therefore a special type of labeled graph in which the labels are numbers.

Definition 1.2 A cycle graph of order n is a connected graph whose edges form a ***cycle*** of length n.

Definition 1.3 A ***spanning tree*** T of a graph G is a sub graph containing all the vertices of G. It is a minimal set of edges that connects all the vertices of G without creating any cycles or loops. Out of all the spanning trees of G, the minimum spanning tree is one with least weight.

Cryptography is the art of protect information by transforming it to unreadable format called Cipher text. The process of converting plain text to cipher text called encryption, and the process of converting cipher text on its original plain text called decryption.

[3*]pulibala70@gmail.com

ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

The remainder of this paper is a discussion of intractable problem from graph theory keeping cryptography as the base.
Firstly we represent the given text as node of the graph. Every node represent a character of the data. Now every adjacent character in the given text will be represented by adjacent vertices in the graph.

2.Proposed Application

Example 2.1 We will encrypt the text or data, say ***R A I N*** , which we will be sending to the receiver on the other end.

Now we change this text into graph by converting each letter to vertices of graph.

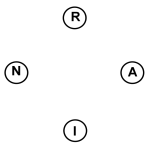

Figure 1: Conver the letter to vertex(node)

To form a Cycle graph, we link each two characters.

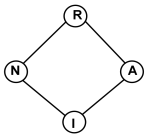

Figure 2: Cycle Graph

[3]*pulibala70@gmail.com

ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

Further we label each edge by using the encoding table, which is followed by most researchers.

A	B	C	D	-	-	-	-	W	X	Y	Z
1	2	3	4	-	-	-	-	23	24	25	26

Table 1: Encoding table

Distance = code(A) − code(R) = $1 - 18 = -17$.
Similarly we can deduce the distances of other edges. Then we label the graph containing all the plain text letters and we get weighted graph which is given below.

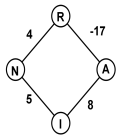

Figure 3: Weighted Graph

After that, we keep adding edges(links) to form a complete graph and each newadded edge(link) has a sequential weight starting from the maximum weight in the encoding table which is 26.Therefore we can add 27,28 and so on.

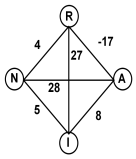

Figure 4: Complete Plain Graph

Then add a special character before the first character to point to the first character, say A is special character, then we get

[3]*pulibala70@gmail.com

ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

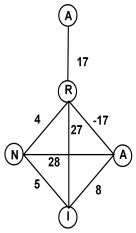

Figure 5: Complete plain graph with special Character

Now represent the above graph in the form of a matrix.

$$A = \begin{bmatrix} 0 & 17 & 0 & 0 & 0 \\ 17 & 0 & -17 & 27 & 4 \\ 0 & -17 & 0 & 8 & 28 \\ 0 & 27 & 8 & 0 & 5 \\ 0 & 4 & 28 & 5 & 0 \end{bmatrix}$$

We now construct a minimal spanning tree of the above graph

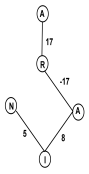

Figure 6: Minimal spanning tree

$$B = \begin{bmatrix} 0 & 17 & 0 & 0 & 0 \\ 17 & 0 & -17 & 0 & 0 \\ 0 & -17 & 0 & 8 & 0 \\ 0 & 0 & 8 & 0 & 5 \\ 0 & 0 & 0 & 5 & 0 \end{bmatrix}$$

Encryption Process :

Now we store the character order in the diagonal instead of zeroes as follows:

[3]*pulibala70@gmail.com

ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

character	A	R	A	I	N
order	0	1	2	3	4

Then the modified B is $\begin{bmatrix} 0 & 17 & 0 & 0 & 0 \\ 17 & 1 & -17 & 0 & 0 \\ 0 & -17 & 2 & 8 & 0 \\ 0 & 0 & 8 & 3 & 5 \\ 0 & 0 & 0 & 5 & 4 \end{bmatrix}$

we multiply matrix A by B to form C.

$$C = \begin{bmatrix} 289 & 17 & -289 & 0 & 0 \\ 0 & 578 & 182 & -35 & 151 \\ -289 & -17 & -225 & 164 & 152 \\ 459 & -109 & -443 & 89 & 20 \\ 68 & -472 & 28 & 239 & 25 \end{bmatrix}$$

Then use a Public Key K to encrypt C

let $K = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

so cipher text C_t=KC

$$C_t = \begin{bmatrix} 527 & -3 & -747 & 457 & 348 \\ 238 & -20 & -458 & 457 & 348 \\ 238 & -598 & -640 & 492 & 197 \\ 527 & -581 & -415 & 328 & 45 \\ 68 & -472 & 28 & 239 & 25 \end{bmatrix}$$

We now send the encrypted data C_t to the receiver.

[3]*pulibala70@gmail.com

ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

527 − 3 − 747 457 348 238 − 20 − 458 457 348 238 − 598 − 640 492 197 527 581 415 328 45 68 472 28 239 25.

Decryption Process :

On the receiver side, C is got from multiplying the cipher text received with the inverse of shared Key K^{-1} Then calculate B by multiplying C by A^{-1}

$$\text{since } K^{-1} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{Therefore } B = CA^{-1} = \begin{bmatrix} 0 & 17 & 0 & 0 & 0 \\ 17 & 0 & -17 & 0 & 0 \\ 0 & -17 & 0 & 8 & 0 \\ 0 & 0 & 8 & 0 & 5 \\ 0 & 0 & 0 & 5 & 0 \end{bmatrix}$$

Then B represent the below graph fig.7, regardless of te diagonal , we use it to retrieve the original text.

suppose that the vertex 0 is A, and by using encoding table

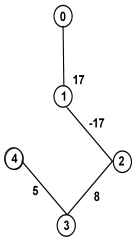

Figure 7: Decrpted Graph

Node 1 = code(A) + 17 = 18, which is character R
Node 2 = code(R) − 17 = 1, which is character A
Node 3 = code(A) + 8 = 9, which is character I
Node 4 = code(I) + 5 = 6, which is character N

[3]*pulibala70@gmail.com

ISSN: 2456-8686, Volume 2, Issue 1, 2018:57-63
DOI : http://doi.org/10.26524/cm27

Which gives us the original text R A I N.

4.Acknowledgement

One of the author (Dr. V. Balaji) acknowledges University Grants Commission, SERO, Hyderabad and India for financial assistance ($No.FMRP5766/15(SERO/UGC)$).

References

[1] Ustimenko VA. On graph-based cryptography and symbolic computations, Serdica, Journal of Computing, 2007, 131-156.

[2] Uma Dixit, CRYPTOGRAPHY A GRAPH THEORY APPROACH, International Journalof Advance Research in Science and Engineering, 6(01), September 2017,BVCNSCS 2017

[3] Wael Mahmoud Al Etaiwi , Encryption Algorithm Using Graph Theory, Journal of Scientific Research and Reports, 3(19), 2014, 2519-2527; Article no. JSRR.2014.19.004

[4] Yamuna M, Meenal Gogia, Ashish Sikka, Md. Jazib Hayat Khan. Encryption using graph theory and linear algebra, International Journal of Computer Application,2012, 2250-1797.

[5] Corman TH, Leiserson CE, Rivest RL, Stein C. Introduction to algorithms, McGraw-Hill, 2nd edition,

[3]*pulibala70@gmail.com