Journal of Engineering and Applied Sciences 14 (1): 219-223, 2019
ISSN: 1816-949X
© Medwell Journals, 2019

Public Key Cryptosystem Based on Graph Theory

Najlae Falah Hameed Al Saffar
Department of Mathematics, Faculty of Computer Science and Mathematics,
University of Kufa, 54001 Najaf, Iraq

Abstract: This study presents a new type of public key cryptosystem, send and receive a secure a written message using English letters frequencies and some properties of graph theory. Experiment results of this type of public key cryptosystem increased the level confidence for exchanging messages.

Key words: Public key cryptosystem, graph theory, trees, receive, confidence, exchanging

INTRODUCTION

Cryptography relies heavily on number theoretic tools. In particular, systems based on hardness of problems in number theory such as factoring and discrete logarithm, form an important part of modern cryptography (Yan, 2002).

The idea of a public key cryptosystem (also, known as an asymmetric key cryptosystem) was introduced by Diffie and Hellman (1976) when they proposed the Diffie Hellman key exchange to exchange keys. In a public key cryptosystem an algorithm that uses two mathematically tools, (a public key and a private key). One of these keys can be used to encrypt a message, the opposite key is used for decryption processes (Xiao *et al*., 2011). The process of encryption and decryption is shown in the following illustration in Fig. 1.

As only the sender has access to his private key, it is possible that only him can decrypt the encrypted data. Even if someone else gains access to the encrypted data, it will remain confidential as they should not have access to the sender's private key, public key cryptosystem can therefore, achieve confidentiality (Galbraith, 2012).

Many researchers were interested in the topic of harnessing the graph theory subject as a tool in cryptosystems such as Ustimenko (2002), Yamuna *et al*. (2012), Wroblewska (2008) and Al Etaiwi (2014).

Converting a matrix to the tree can implied using some properties of graph theory. Where Graph theory comes with various properties which are used for characterization of graphs depending on their structures. These properties are defined in specific terms pertaining to the domain of graph theory. We will discuss a few basic properties that are common in all graphs.

In English language there are some words or supplementary additions to words or even a part of word occurs in any message more than others such as: the, am, tion, un, etc. Indeed the letters frequency in English text has been studied for use in ciphering purpose. There techniques to counts these frequencies which give different charts for common letters, one of them based on the frequency which use in texts. In this paper we will divided the groups of words or additions into two classes: words such as: the, you are etc and part of words (affixes) such as: tion, auto for as in Table 1 and 2, respectively.

Basic graph definitions and properties: Graph theory is a branch of mathematics concerned about how networks can be encoded and their properties measured (Chartrand and Lesniak, 2016). In this study, we will discuss some of basic concepts that we well encounter throughout our investigation (Gross and Yellen, 2004).

Definition: A graph G is a set of vertex (nodes) v connected by edges (links) e. in other words G = (v, e).

Definition: A node (Vertex) v is a terminal point or an intersection point of a graph.

Definition: A Link (Edge) e is a link between two nodes.

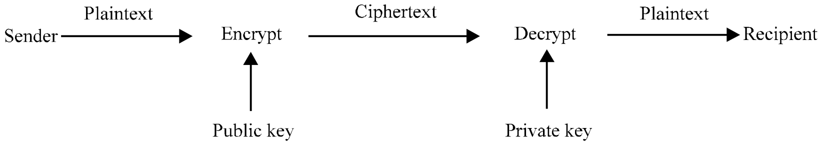

Fig. 1: Modren puplic key cryptosystem (encryption and decryption process)

Table 1: Cods for numbers and letters

Letters	Numbers
A	00000000
B	00000001
C	00000010
D	00000011
E	00000100
F	00000101
G	00000110
H	00000111
I	00001000
K	00001001
L	00001010
M	00001011
N	00001100
O	00001101
P	00001110
Q	00001111
R	00010000
S	00010001
T	00010010
V	00010011
X	00010100
Y	00010101
Z	00010110
0	00010111
1	00011000
2	00011001
3	00011010
4	00011011
5	00011100
6	00011101
7	00011110
8	00011111
9	00100000
10	00100001
Space	00100010
.	00100011
,	00100100

Definition: A node r where every other node is the extremity of a path coming from r is a root.

Definition: A connected graph without a cycle is a tree. The basic structural properties of a graph are:

- A graph is symmetrical graph if each pair of nodes linked in one direction is also linked in the other
- A graph is asymmetric graph if it has no nontrivial symmetries
- Assortative networks are those characterized by relations among similar nodes
- Disassortative networks are found when structurally different nodes are often connected
- A graph is complete if two nodes are linked in at least one direction
- A complete graph is connected if for all its distinct pairs of nodes there is a linking chain

We will use all these as tools in cryptosystem.

Table 2: Cods for special common parts of English words

Number	----------Common parts of English words----------			
00100101	All	ad-	-an-	-able
00100110	And	af-	-at-	-al
00100111	Are	al-	-au-	-ar
00101000	As	As-	-co-	-arly
00101001	At	At-	-dd-	-ary
00101010	Be	Auto-	-ea-	-ate
00101011	But	Be-	-ed-	-ation
00101100	By	Co-	-edt-	-dom
00101101	Can	Com-	-ee-	-ence
00101110	For	Con-	-en-	-ency
00101111	From	De-	-ent-	-ent
00110000	Had	Dec-	-er-	-er
00110001	Have	Di-	-es-	-ery
00110010	He	Dif-	-ff-	-fy
00110011	His	Dis-	-for-	-ful
00110100	In	Dis-	-ha-	-ial
00110101	Is	Ex-	-has-	-ian
00110110	It	Geo-	-he-	-ical
00110111	No	Il-	-in-	-ify
00111000	Not	Im-	-io-	-ing
00111001	Of	In-	-ion-	-ion
00111010	On	Maxi-	-ll-	-ious
00111011	One	Min-	-mm-	-ism
00111100	Or	Neo-	-nce-	-ist
00111101	That	Non-	-nd-	-ity
00111110	The	Of-	-nn-	-ive
00111111	There	Op-	-nt-	-ize
01000000	They	Pan-	-of-	-ly
01000001	This	Per-	-on-	-ment
01000010	To	Post-	-oo-	-ness
01000011	Was	Pre-	-pp-	-once
01000100	We	Pro-	-re-	-or
01000101	Were	Re-	-ss-	-ors
01000110	What	Semi-	-tha-	-ous
01000111	When	Sub-	-the-	-ship
01001000	Will	Suc-	-ti-	-sion
01001001	With	Sur-	-tio-	-tion
01001010	Yes	Tri-	-to-	-tor
01001011	You	Un-	tt-	-yst
01001111	Your	Under-	-th-	-wen

MATERIALS AND METHODS

Proposed method: At the beginning we can adopt two tables, the first one Table 1 gives cods for the numbers and letters in English with some special symbols: space, . and ,. And the second table Table 2 gives cods for special common parts of English words.

From these tables we will convert the plaintext (say M) to the binary form Plaintext (say M_1). Then create the M_2 matrix has column (Sum) which is the summation corresponding row cells. At next step we will translate the matrix M_2 to the T_1 tree using the properties of created it. Then raise each item of the tree to the power e where e (public key*)* is $1<e<\varphi$ (n) with gcd (e. φ (n)) = 1 and n = pq for primes p and q, to get C which is the ciphertext.

The decryption process is get the tree after computing d (private key) $1<d<\varphi$ (n) with de $\equiv$ 1 mod φ (n) and then convert it to matrix to get M_2. Finally using Table 1 and 2 to recover the original M.

RESULTS AND DISCUSSION

Example: Suppose that the original message is the statement "A Beautiful Mind". The first step is to convert M-M$_1$, using Table 1 and 2 as follows:

A	00000000
Be-	00101011
-au-	00100111
-ti-	01001000
-ful	00110011
Space	00100010
Min-	00111011
d	00000011

The second step is create the matrix as follows:

	1	2	3	4	5	6	7	8	Sum
1	0	0	0	0	0	0	0	0	0
2	0	0	1	0	1	0	1	1	4
3	0	0	1	0	0	1	1	1	4
4	0	1	0	0	1	0	0	0	2
5	0	0	1	1	0	0	1	1	4
6	0	0	1	0	0	0	1	0	2
7	0	0	1	1	1	0	1	1	5
8	0	0	0	0	0	0	1	1	2

The third step is translate the matrix to the tree. In the normal tree, every node can have any number of children. One is known as a left and the other is known as a right child. The algorithm of constructing of the tree is as the follows. The node will be for the cell which has the value while the root will be the cell which has value. The last column in the matrix is the summation corresponding row cells, the row which has less value will be the root of the tree. After determine the root, we will remove both row and column of the root, then update the summation corresponding row cells. All the nods have the least sum after removing will be the children of the root. Next for the first child, do the same processes for the remaining children. In the following we will illustrate these steps. For M$_2$, we will remove the row and column for the element that has the least sum which is. So, the matrix will be as follows:

	2	3	4	5	6	7	8	Sum
2	0	1	0	1	0	1	1	4
3	0	1	0	0	1	1	1	4
4	1	0	0	1	0	0	0	2
5	0	1	1	0	0	1	1	4
6	0	1	0	0	0	1	0	2
7	0	1	1	1	0	1	1	5
8	0	0	0	0	0	1	1	2

All the nodes has the least sum will be the children of 1, they are 4, 6 and 8. For the 4, we will remove the corresponding row and column. The least value of its sum will be the child nods of 4.

	2	3	4	6	7	8	Sum
2	0	1	1	0	1	1	4
3	0	1	0	1	1	1	4
5	0	1	0	0	1	1	3
6	0	1	0	0	1	0	2
7	0	1	1	0	1	1	4
8	0	0	0	0	1	1	2

So, the element 5 will be the child of 4 in the T. For the next steps, we will repeat the same for 6 and 8.

	2	3	5	7	8	Sum
2	0	1	1	1	1	4
3	0	1	0	1	1	4
5	0	1	0	1	1	3
7	0	1	1	1	1	4
8	0	0	0	1	1	2

And then:

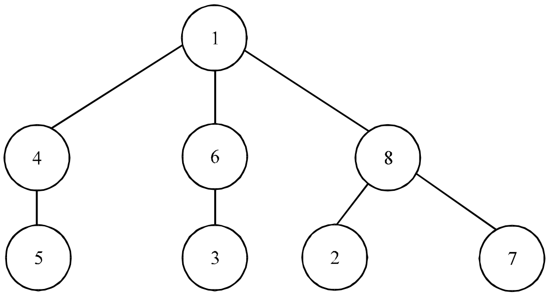

Fig. 2: Tree T

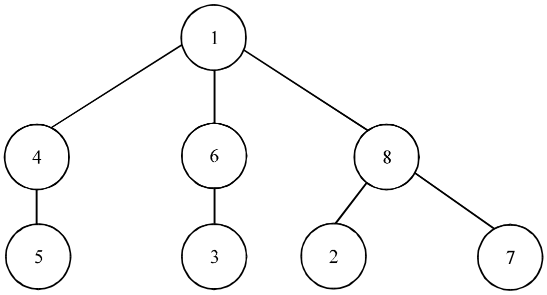

Fig. 3: Tree T for matrix M_2

	2	3	5	7	Sum
2	0	1	1	1	3
3	0	1	0	1	2
5	0	1	0	1	2
7	0	1	1	1	3

Finally, the tree T will be as follows (Fig 2). As the processing of creating the ciphertext C, we will take p and q as and 11, respectivly. So, φ (n) = φ (p·q) = 40. Thus, we will chose the public key e = 7 gcd (7, 40) = 1 as required. So, C will be as follows:

$$(1)^7 \equiv 1 \bmod 55, \quad (4)^7 \equiv 49 \bmod 55,$$
$$(6)^7 \equiv 41 \bmod 55, \quad (8)^7 \equiv 2 \bmod 55,$$
$$(5)^7 \equiv 25 \bmod 55, \quad (3)^7 \equiv 42 \bmod 55,$$
$$(2)^7 \equiv 18 \bmod 55, \quad (7)^7 \equiv 28 \bmod 55$$

For decryption process, firstly, we have to compute the private d where d (7) = 1 mod 40, since, 7^{-1} mod 40 is 23, so, d = 23. Therefore, the elements of the tree will be as follows:

$$(1)^{23} \equiv 1 \bmod 55, \quad (49)^{23} \equiv 4 \bmod 55,$$
$$(41)^{23} \equiv 6 \bmod 55, \quad (2)^{23} \equiv 8 \bmod 55,$$
$$(25)^{23} \equiv 5 \bmod 55, \quad (42)^{23} \equiv 3 \bmod 55,$$
$$(18)^{23} \equiv 2 \bmod 55, \quad (28)^{23} \equiv 7 \bmod 55$$

That's, we have the following tree again (Fig. 3). And then by properties we will get the matrix M_2. Finally using the Table 1 and 2 to get the plain text M.

CONCLUSION

In this study, we introduce an algorithm to encrypt and decrypt a plain text through public channels using encoding tables and some properties of graph theory which is used to convert a matrix to a tree. This algorithm use the concepts of public key cryptosystem that is it is based in the most widely hard mathematical problem which is integer factorization problem.

RECOMMENDATIONS

There are many improvements that can be done as future research such as do some changes to make the size of the matrix less than the one in the proposed algorithm to make the research easer. As another suggest for improving the proposed algorithm is using another hard mathematical problem such as discrete logarithm problem or Computational Diffie-Hellman problem. These problems may be effect on the difficulty of the proposed algorithm.

REFERENCES

Al Etaiwi, W.M., 2014. Encryption algorithm using graph theory. J. Sci. Res. Rep., 3: 2519-2527.

Chartrand, G. and L. Lesniak, 2016. Graphs and Digraphs. 6th Edn., CRC Press, Boca Raton, Florida, USA., ISBN: 9781498735766, Pages: 628.

Diffie, W. and M. Hellman, 1976. New directions in cryptography. IEEE. Trans. Inf. Theor., 22: 644-654.

Galbraith, S.D., 2012. Mathematics of Public Key Cryptography. Cambridge University Press, Cambridge, England, UK., ISBN:978-1-107-01392-6, Pages: 616.

Gross, J. and J. Yellen, 2004. Handbook of Graph Theory. CRC Press, Florida, USA.,.

Ustimenko, V.A., 2002. Graphs with special arcs and cryptography. Acta Appl. Math., 74: 117-153.

Wroblewska, A., 2008. On some properties of graph based public keys. Albanian J. Math., 2: 229-234.

Xiao, Y., F.H. Li and H. Chen, 2011. Handbook of Security and Networks. 6th Edn., World Scientific Publishing, Singapore, ISBN:9789814273039, Pages: 551.

Yamuna, M., M. Gogia, A. Sikka and M.D.J.H. Khan, 2012. Encryption using graph theory and linear algebra. Intl. J. Comput. Appl., 5: 102-107.

Yan, S.Y., 2002. Number Theory for Computing. 2nd Edn., Springer, Berlin, Germany, ISBN:9783540430728, Pages: 435.