Indian Journal of Science and Technology, *Vol 6(12), 5607–5611, December 2013*

Cryptography: a Mathematical Approach

P. M. Durai Raj Vincent[1*], Syed Amber Iqbal[2], Karan Bhagat[2] and Kamal Kant Kushwaha[2]

[1]Assistant Professor (Senior), SITE, VIT University; pmvincent@vit.ac.in
[2]III B.Tech(IT), SITE, VIT University;
syedamber91@yahoo.com, karan.bhagat2010@vit.ac.in, kushwaha.kamalkant@yahoo.com

Abstract

Networking is a way of connecting geographically distributed devices so as to communicate and share resources such as printers, files etc. Nowadays, Networking is extensively used in banking sector, e–commerce, social networking sites, news groups, sharing of software, sharing of valuable information among others and so many more. The unauthorized access to these networks is restrained by the administrator who apply and use various provisions and policies which is termed as network securities. Network security refers to any activity designed to protect network. Network security is an extremely valuable and procreative amenity. A threat exists whenever the security is perceived to get breached and vulnerability likely to be exploited. To enhance and strengthen a safeguard mechanism in protecting information multiple layers of security are implemented in our algorithm instead of adopting a single layer which at times may fall through.

Keywords: Transposition, Randomization, Quadratic Encryption, Hash Code, Cryptography.

1. Introduction

To maintain the integrity of network security there are certain encryption and decryption algorithms that are followed and cryptography is used as explained [1, 2]. For successful transmission of a message we have to ensure that the secrecy of message is maintained. For this purpose a key is generated which is used to create cipher texts that is being sent to the destination [3, 1]. The key is symmetric so that the sender and the receiver will use the same key [4]. To increase the complexity of the encryption algorithm multiple layers of security are used in algorithm. Sender will verify the received message in order to check for any alteration using the hash function [5]. The algorithm is built on five modules and it is practically impossible to break it [6].

1.1 Key Generation

Primarily an odd number is used to generate the Pythagorean triplets [7]. A Pythagorean triplet is selected and a set of mathematical operations are performed and a key is generated [8]. We obtain a sum of the Pythagorean Triplet which is then sent to the destination.

1.2 Key Finder

The Pythagorean triplet can be re–derived from the sum which is obtained from the sender [9].

1.3 Cryptographic Algorithm

The generated key is used in the encryption algorithm which comprises of three layers: Transposition [10], Randomization and Quadratic Encryption sequentially. Decryp-tion process involves obtaining the original message back using Quadratic Decryption, De–randomization and Transposition sequentially. To ensure authentication, digital signature is used [11].

1.4 Hash Code Generation

Conversion of the message to its equivalent hash code is done using a hash function [12]. Hash Code is generated for each message.

1.5 Hash Code Verification

Integrity of the message is preserved by comparing received hash code and generated hash code [12]. If both the received and generated hash code is equal then the message is unaltered.

*Corresponding author:
P. M. Durai Raj Vincent (pmvincent@vit.ac.in)

2. Algorithm

2.1 Key Generation

An odd number ‘n’ is selected and pairs of co–prime numbers are found out whose sum is ‘n’. Let us select a pair x_i, y_i and we perform the following operations to obtain a_i, b_i, c_i which are the Pythagorean Triples.

$$a_i = x_i^2 - y_i^2$$

$$b_i = 2\, x_i * y_i$$

$$c_i = x_i^2 + y_i^2$$

when a_i, b_i, c_i are obtained then we obtain the sum of these variables and store the sum in the variable d_i:

$$d_i = a_i + b_i + c_i$$

Keys used for encryption of the message are (a_i, b_i, c_i). d_i thus obtained is sent to the sender.

2.2 Key Finder

We obtain all the odd factors of the received d_k. For each odd factor f_k of d_k we find the co–prime numbers. Let us consider x_k, y_k as the pairs whose sum is f_k. We perform the following set of operations:

$$a_k = x_k^2 - y_k^2$$

$$b_k = 2\, x_k * y_k$$

$$c_k = x_k^2 + y_k^2$$

Obtained variables are added and stored in a new variable d_l.

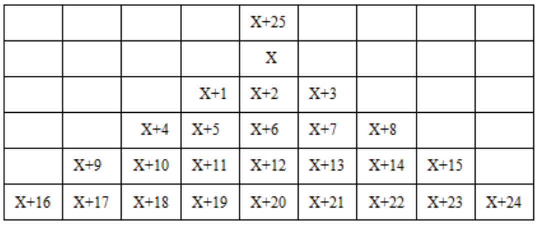
$$d_l = a_k + b_k + c_k$$

If the obtained d_l is equal to d_k then we can conclude that (a_k , b_k , c_k) is the desired triplet. For decryption key to be used is (a_k, b_k, c_k).

2.3 Encryption

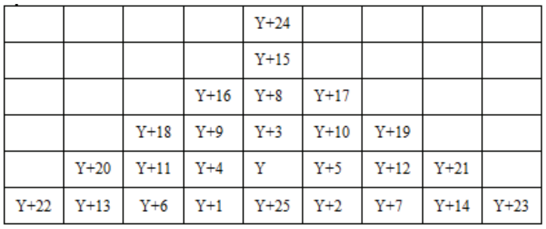
Keys used for encryption are (x, y, z) where x = a_i % 26, y = b_i % 26, z = c_i.

1) Layer I–Triangular Transposition: We use the pair of keys (x, y) from the set of keys (x, y, z). (Figure 1 and Figure 2) all the letters from message M are numbered from 1 to 26 i.e. A = 1, B = 2,…Z = 26. Arrange these numbers in a triangular manner. Numbers in the triangle depend on the value of x and y and the arrangement is specified in the diagram. This key is kept fixed for a particular session.



				X+25				
				X				
			X+1	X+2	X+3			
		X+4	X+5	X+6	X+7	X+8		
	X+9	X+10	X+11	X+12	X+13	X+14	X+15	
X+16	X+17	X+18	X+19	X+20	X+21	X+22	X+23	X+24

Figure 1. Encryption Triangle 1.



				Y+24				
				Y+15				
			Y+16	Y+8	Y+17			
		Y+18	Y+9	Y+3	Y+10	Y+19		
	Y+20	Y+11	Y+4	Y	Y+5	Y+12	Y+21	
Y+22	Y+13	Y+6	Y+1	Y+25	Y+2	Y+7	Y+14	Y+23

Figure 2. Encryption Triangle 2.

Replace the number in the first triangle with the number corresponding its position in the second triangle. We will name this number as M_{i1}.

1) Layer II–Randomization: Using z as the key for this encryption and applying the following equation:-

$$M_{i2} = M_{i1} * z + \text{Random (z)}$$

M_{i2} is a random real number obtained which may differ for same letters of the message.

2) Layer III–Quadratic Encryption: A quadratic equation is used for encryption using the set of keys (a_i, b_i, c_i) and M_{i2} obtained as a result of Layer II Encryption method:

$$E_i = a_i * (M_{i2})^2 + b_i * (M_{i2}) + c_i$$

E_i is the decimal number obtained for a letter in the message. We will obtain a set {E_1, E_2, E_3,…, E_n} where n is the length of the message M and this set is the cipher text. The encrypted set of cipher text is sent to the receiver

2.4 Decryption

The receiver will use the key finder to obtain the keys (p, q, r) where p = a_k % 26, q = b_k% 26, r=c_k. Using these keys he will apply the decryption algorithm.

1) Layer I–Quadratic Decryption: A set of cipher text obtained is {E_1, E_2,…, E_n}. For decryption of a letter let us take E_k from the set. Apply the following quadratic equation to obtain M_{k2} which will be one of the roots of the equation:

$$a_k * (x)^2 + b_k * (x) + r - E_k=0$$

The roots obtained will be real roots since the encryption procedure deals with the real numbers. Out of the 2 roots obtained, one root will be positive and the other root will be negative. The absolute value of the negative root will be higher than the positive root since:

Sum of the roots = – (q/p)
Product of the roots = $(r - E_k)/p$
where $p > 0$, $q > 0$, $r > 0$ and $(r - E_k) < 0$

We will discard the negative root and take M_{k2} equal to the positive root.

2) Layer II–De–randomization: Using the key r and M_{k2} obtained we will obtain M_{k1}. Where $M_{K2} = N * r + \text{Random}(r)$ where $1 <= N <= 26$

Use the following pattern to de–randomize:

$A = 1 * r + \text{Random}(r)$ where $(1 * r) <= A < (2 * r)$
$B = 2 * r + \text{Random}(r)$ where $(2 * r) <= B < (3 * r)$
…
…
$Z = 26 * r + \text{Random}(r)$ where $(26 * r) <= Z < (27 * r)$

Obtain the value of N for which M_{k2} lies in one of the above intervals. M_{k1} is the decimal equivalent of the obtained alphabet.

3) Layer III–Triangular Transposition: We use the pair of keys (p, q) from the set of keys (p, q, r). (Figure 3 and Figure 4) these keys will remain the same for a particular session. Use the following arrangements:

				Q+24				
				Q+15				
			Q+16	Q+8	Q+17			
		Q+18	Q+9	Q+3	Q+10	Q+19		
	Q+20	Q+11	Q+4	Q	Q+5	Q+12	Q+21	
Q+22	Q+13	Q+6	Q+1	Q+25	Q+2	Q+7	Q+14	Q+23

Figure 3. Encryption Triangle 3.

				P+25				
				P				
			P+1	P+2	X+3			
		P+4	P+5	P+6	P+7 ˙	P+8		
	P+9	P+10	P+11	P+12	P+13	P+14	P+15	
P+16	P+17	P+18	P+19	P+20	P+21	P+22	P+23	P+24

Figure 4. Encryption Triangle 4.

Replacing the number in the first triangle with the number corresponding its position in the second triangle. The number thus obtain is converted back to its alphabetic equivalent.

The above process is described for E_k from a set of cipher text $\{E_1, E_2, …, E_n\}$. We need to repeat the process n times i.e. length of the message for obtaining plain text from the cipher text.

3. Hash Code Generation

For each letter in the original message, obtain the equivalent decimal number N_i and perform the following operation:

$C_i = a^* (N_i)^2 + b^* (N_i) + c$; where a , b ,c are the keys used for encryption.

Let C be the summation of $\{C_1, C_2, C_3, …, C_n\}$ decimal numbers and E be the summation of $\{E_1, E_2, E_3, …, E_n\}$ decimal numbers of the cipher text. Then,

Hash Code = Log (C + E)/Log (d) where d = a + b + c.

4. Hash Code Verification

Hash Code, $\{E_1, E_2, …., E_n\}$ decimal cipher for each alphabet in the message and the summation of keys is received by the receiver. Receiver is required to obtain E the summation of $\{E_1, E_2, E_3 …, E_n\}$, decrypt the message and obtain the value of C the summation of $(C_1, C_2, C_3, …, C_n)$. Then Hash Code is calculated as follows:

Calculated Hash Code = Log (C+E)/Log (d), where a, b, c are the keys and d= a + b + c.

If received Hash Code is equal to calculated Hash Code, then we can conclude that the message received is the correct message without any alteration. And hence proves the integrity of the message.

4.1 Example

Let us consider an odd number 679. We have many coprime triplets and we select a co-prime triplet (459683, 1353, 459685) and calculate sum of these co–prime numbers. The value is 3445312 which is to be sent to the receiver. The key to be used for encryption is (3, 4, 458685).

Now let the message to be sent be "hello" and its decimal equivalent is (8, 5, 12, 12, 15) The encrypted value by triangular transposition will be (10, 9, 21, 21, 1).These values then undergo randomization and quadratic encryption to produce cipher code corresponding to each alphabet in the message.

$$h = 8.3283041363310531E18$$

$$e = 6.9462663457208003E18$$

$$l = 2.2334643574764695E18$$

$$l = 3.9915536922842993E18$$

$$o = 5.0704122924857432E16$$

For the given 3 digit decimal odd number, there is large decimal output corresponding to each letter and for similar letters there is a large variation.

5. Graphical Analysis

Analysis of Complexity by varying the 3rd key keeping k_1 and k_2 constant for each set of input. Consider the 3 case.

Case 1: Let the 3rd key be a 2 digit number ($1 < r < 100$) (Table 1)

Case 2: Let the 3rd key be a 3 digit number ($100 < r < 1000$) (Table 2)

Case 3: Let the 3rd key be a 4 digit number ($1000 < r < 10000$) (Table 3)

In the graphs below, (Figure 5, Figure 6, Figure 7 and Figure 8) the y axis corresponds to the output of encryption process and the x axis corresponds to the key used for randomization. The output shown below is an example taking $k_1 = 1$ and $k_2 = 1$

Case 1:

Table 1.

Key 3	Output
10	4042
30	37086
50	118730
70	193230
90	315372

Case 2:

Table 2.

Key 3	Output
100	470010
300	3395106
500	1.0841056E7
700	2.2920856E7
900	3.0443706E7

Case 3:

Table 3.

Key 3	Output
1000	4.562427E7
3000	3.3432594E8
5000	1.11159394E9
7000	2.26048648E9
9000	3.19514115E9

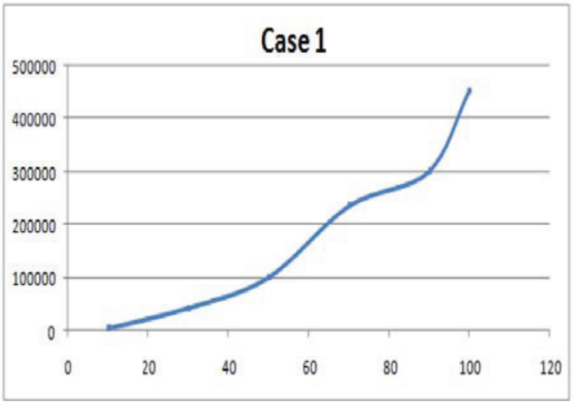


Figure 5. Graph for Table 1.

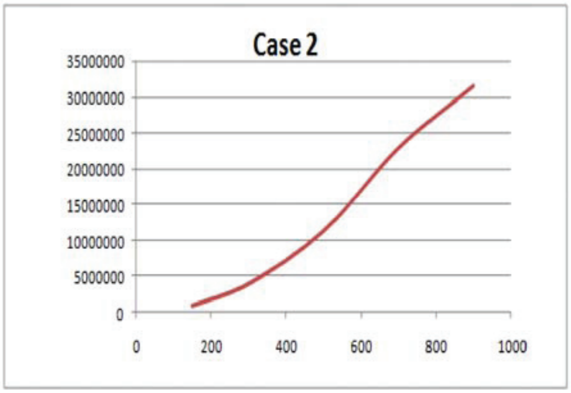


Figure 6. Graph for Table 2.

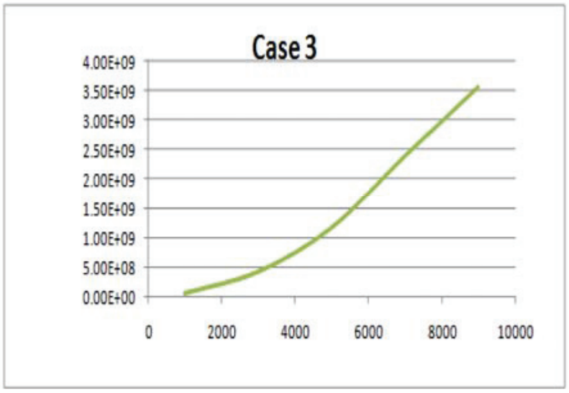


Figure 7. Graph for Table 3.

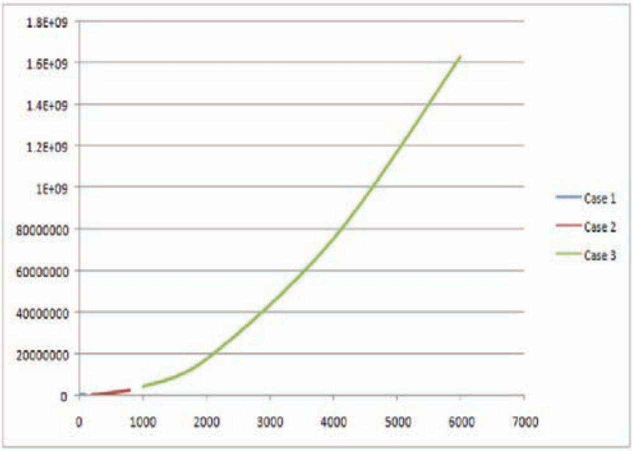


Figure 8. Combined graph for Table 1, 2, 3.

6. Conclusion

Our algorithm has high complexity and deals with random numbers. Applying the brute force attack is difficult to perform as the decimal value of encrypted message is very large. Time taken for decrypting the message by the special receiver is less and the process is very efficient since simple methods are employed. Use of symmetric key insures authentication and confidentiality. Hash Code will validate the message received and provide message integrity. Algorithm absolutely satisfies the necessities of a good encryption algorithm for providing secure communication.

7. References

1. Stallings, W. (2006). Cryptography and Network Security: Principles and Practice, 4th ed. Englewood Cliffs, NJ: Prentice Hall.
2. Sakalli, M.T.; Bulus, E.; Buyuksaraqoglu, F., "Cryptography education for students," Information Technology Based Higher Education and Training, ITHET 2004. Proceedings of the Fifth International Conference on, vol., no., pp.621, 626, 31 May-2 June 2004.
3. Ma Ji, "Fractal theory based on the pseudo-attacks on encryption model," Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on, vol.9, no., pp.659,662, 9–11 July 2010.
4. Ayushi," A Symmetric Key Cryptographic Algorithm", International Journal of Computer Applications (0975–8887) Volume 1 – No. 15.
5. Shannon C.E. Communication Theory of Secret Systems // Bell Syst. Tech.Jour., 1949, V.28, pp.658–715.
6. Vignesh, R.S.; Sudharssun, S.; Kumar, K.J.J., "Limitations of Quantum & the Versatility of Classical Cryptography: A Comparative Study," Environmental and Computer Science, 2009. ICECS '09. Second International Conference on , vol., no., pp.333,337, 28–30 Dec. 2009.
7. Swartzlander, Diane, "Pythagorean Triples" (2007). MAT Exam Expository Papers. Paper 42.
8. Ciobanu, G.; Rusu, D., "Pythagoras: An Interactive Environment for Plane Geometry," Intelligent Computer Communication and Processing, 2007 IEEE International Conference on , vol., no., pp.283,286, 6–8 Sept. 2007.
9. Blinn, J.F., "How to solve a quadratic equation?," Computer Graphics and Applications, IEEE , vol.25, no.6, pp.76,79, Nov–Dec. 2005
10. Data Encryption Standard (DES), Nat'l Institute of Standards and Tech., Federal Information Processing Standards Publication 46–3, 25 Oct. 199; http://csrc.nist.gov/sryptval/des.htm
11. Kaur, R.; Kaur, A., "Digital Signature," Computing Sciences (ICCS), 2012 International Conference on, vol., no., pp.295,301, 14–15 Sept. 2012.
12. Thulasimani Lakshmanan, Madheswaran Muthusamy," A Novel Secure Hash Algorithm for Public Key Digital Signature Schemes", The International Arab Journal of Information Technology, Vol. 9, No. 3, May 2012.