International Journal of Latest Trends in Engineering and Technology
Vol.(10)Issue(2), pp.335-339
DOI: http://dx.doi.org/10.21172/1.102.54
e-ISSN:2278-621X

SYMMETRIC KEY ENCRYPTION ALGORITHM USING SPIDER WEB (ESP) VERSION-1

Aashijit Mukhopadhyay[1], Soumick Chatterjee[2] , Asoke Nath[3]

Abstract- The authors in this paper have introduced a totally new concept of utilizing the spider web structure to store plain text data bytes temporarily and encrypt them. Spider Web [1] is utilized as 2-dimensional data structure in which the plain text information will be stored to complete the encryption process. The encryption process begins with the storage of plain text data bytes in the spider web structure. The pre-generated random data-insects are made to follow a rectilinear path towards the web structure in a constant circular motion around the axis perpendicular to the plane of the web. These random data-insects all on the web and tend to get stuck to the sticky cells and change their characteristics in turn encrypting them. These encrypted characters after several rounds of reaction with random data-insects are then extracted. The random data-insects are re-encrypted using the enlarged key string and finally they are again made to react with the encrypted data bytes to produce a final string which can quite easily and flamboyantly travel through the insecure medium of transmission without the fear of being deciphered by some unknown brute attacker.
Keywords –Spider Web, Symmetric Key Cryptography, Archimedean Spiral, Data Diffusion, Data-insects.

1. INTRODUCTION

With the advent of faster computers, security of every individual in the webbed network tends to be at stake. A considerable part of the world's population is attached to every day internet usage which includes passing secret and delicate messages, along with banking details and other financial transactions. The authors in this paper have found a totally new method of applying theories of cryptography to encrypt information between a given sender receiver pair. The concept of Spider Web have been utilized the main basis of the total process.

Spider Web structure have been used in this algorithm as the temporary data storage mechanism before performing the encryption process. Spider web structure has been built on a simple 2-d linear data structure. The spider web has been built using concepts from the Archimedean Spiral. The centre of the spider web structure is decided and the web skeleton is created by drawing lines from the centre to the ends of the 2-dimentional matrix. Above this skeletal structure the spirals will be constructed in order to develop the final web. Starting from the centre of the structure, the spiral construction moves around the structure several times until reaching the end point of the matrix. The cells where the spiral and the previously constructed skeleton intersect are considered the correct place to attract the data-insects to get stuck. So, those cells are filled with plain text data bytes. During the first stage of encryption, the pre-generated random data-insects are made to move forward towards the spider web following a rectilinear path for this version of algorithm. The data-insects are in a constant circular motion about an axis that is perpendicular to the plan of the web structure. When the insects get stuck at a particular intersection of the web, they change the characteristics of the cell contents and thus encrypt the particular information byte. After several thousand rounds of this encryption, all the plain text characters are encrypted using the first round of random data-insects. The encrypted information is extracted. The inputted key is first converted into a large string with the help of several non-reversible arithmetic operations. The random data-insects that have been used for encryption are now made to react with this large string to produce characters that lies in the range of Unicode character set. These characters are finally made to react with the previously encrypted characters to produce the final encrypted text. This group of encrypted characters is finally padded with the encrypted random data-insects to form the final encrypted textual matter ready to be transmitted through an un-reliable and insecure channel.

Both the encryption and the decryption algorithms have been tested against a large data set of known plain text characters, and it has been observed that both statistical attack and brute force attack could not break any message that will be transmitted using this algorithm.

[1] Department of Computer Science, St. Xavier's College (Autonomous), Kolkata, West Bengal, India
[2] Data and Knowledge Engineering, Institute of Technical and Business Information Systems, Faculty of Computer Science and Department of Biomedical Magnetic Resonance, Institute of Physics, Faculty of Natural Sciences, Otto-von-Guericke University, Magdeburg, Germany
[3] Department of Computer Science, St. Xavier's College (Autonomous), Kolkata, West Bengal, India

2. LITERATURE SURVEY

The main concept of the paper comes from the idea of building the spider web and trapping insects into it [1]. The spider releases a sticky thread that is blown away with the wind. If the breeze carried the silken line to a spot where it sticks the first bridge is formed. The spider cautiously crosses along the thin line reinforcing it with a second line. She enforces the line until it is strong enough. After the first horizontal line the spider makes a loose thread and constructs with a second thread a Y-shaped line. These are the first three radii of the web. Then a frame is constructed to attach the other radii to. After all the radii are completed the spider start to make the circular threads. At first non-sticky construction threads a made. The distance between the threads is so wide that the spider can span the width with her legs. Finally the sticky thread is woven between the circular thread. While attaching the sticky thread to the radii the construction thread is removed by the spider. Then web is completed with non sticky radii and sticky circular threads and the spider can rest and sit in the centre of the web with her head down. After a night of hunting the web becomes worn out. The spider removes the silk in the morning by eating it, only leaving the first bridge line. After a daytime rest the spider constructs a new web in the evening. If the catch was low and the web is not heavily damaged the web may stay during the day and be reused after minor repairing. There are a lot of variations on this type of orb web. Spiders can leave a hole in the centre, leave out one sector or only make one sector (like a piece of a pie) and the extreme form is a single line web. The web shown in Fig. 1 is made by the orb web spiders Araneusdiadematus [1].

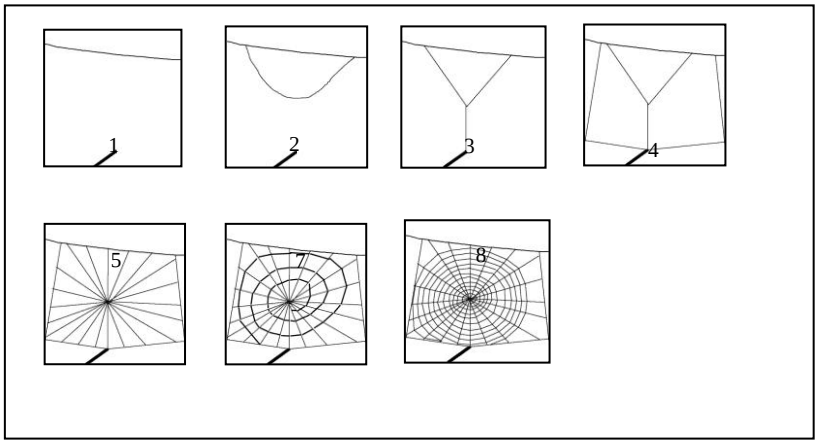


Fig. 1: (1-8) Steps to build a web by an orb web spider.

3. PROPOSED ALGORITHM AND REQUIRED EXPLANATION

Step 1 Input the plain text in the form of a text file.

Step 2 Input Secret Password for a given sender-receiver pair that has been pre-decided, extract characters and multiply by a factor determining the power of encryption and store it in a variable number_of_times.

Step 3 Linked_Listi = Plain_Text_Bytei ;i=i+1

Step 4 Select Centre_Web[x,y] values from the random byte file. The size of the matrix is max_size [xm,ym], such that x<=xmand y<=ym. Draw the line [(x1, y1), (x2, y2)] from [x,y] to [xm,ym] such that D = √{(xm-x)2 + (ym-y)2} is minimum. Keeping this line as the primary line, the other lines are drawn from [x,y] to [xm,ym] such that the angular distance from this line remains random.

Step 5 Atwo dimensional data layer (data_layer[xm][ym]) is utilized to generate the final spider web data structure. Then the Archimedean Spiral is drawn keeping its center at Centre_Web[x,y]. The equation governing the construction of the Archimedean Spiral is: The radius r (t) and the angle t are proportional for the Archimedean spiral. The equation is: Polar equation: r (t) = at ['a' is constant]. From this follows, Parameter form: x (t) = at cos (t), y (t) = at sin (t), Central equation: $x^2+y^2 = a^2$ [arc tan (y/x)] ^{2}. If x1=<x(t)<=x2 and y1=<y(t)<=y2 then data_layer[x][y]=1 or else data_layer[x][y]=0.

Step 6 If data_layer[xi][yi]==1 then data_layer[xi][yi] = Linked_Listiand i=i+1.

Step 7 Now, the data layer is ready for the first stage of encryption procedure to take place.

Step 8 The data layer and the previously extracted information about the number of times of encryption are passed on the first stage spider web encryption module. This module utilizes the random bytes previously generated in the form of ordered pairs to perform the encryption.

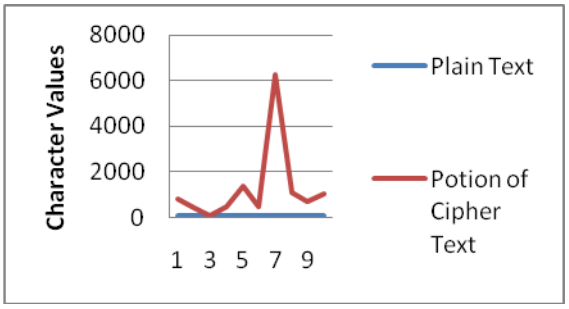
Step 9 Random_Data_Insectsi[x,y]is a list of ordered pairssuch that 1=<x<=255 and 1=<y<=255.Motion_Insectsi[x,y] is stored such that (x-a)2+(y-b)2=r2. Here 2*r<=ymand 2*r<=xm. For each Motion_Insectsi[x,y] such that x=xi and y=yi and data_layer[xi][yi] > 0 then Encri[x][y]=Random_Data_Insectsi[x][x] ^ data_layer[xi][yi]; i=i+1 until i< size of Motion_Insects.

Step 10 Repeat Step 9 until counter>=number_of_times

Step 11 The encrypted plain text bytes are extracted from the spider web data layer. The utilized random bytes and the random data-insects used for the first stage of encryption are also extracted and segregated into two different strings for the next stage of encryption. The previously generated key is now used to encrypt the segregated random data-insects. But before that this key is stretched using arithmetic operations like multiplication to generate a large string.
Step 12 This string is part by part extracted to react with the Random_Data_Insectsi[x,y]bytes according to the digit length of the Random_Data_Insectsi[x,y].The extracted string portions(String_Keyi[x,y]) are made to react with the Random_Data_Insectsi[x,y] such that Encr_Keyi = String_Keyi[x,y] ^ Random_Data_Insectsi[x,y]; i=i+1 .Characters in the Encr_Keyi are no more just in ASCII character set range, but many of the characters have been in the range of the UNICODE character set.
Step 13 These characters are finally utilized to re-encrypt the Encri.
Step 14 Final_Messagei = Encri ^Encr_Keyi ; i=i+1
Step 15 The final message (Final_Messagei) size for this version of the algorithm is generally around to 7 KB. This message can then be forwarded freely through the insecure channel without the fear of being deciphered.

4. RESULTS AND DISCUSSION

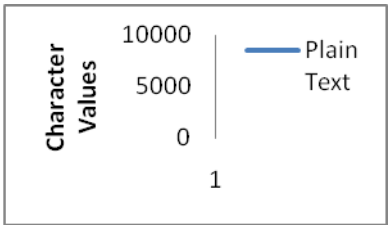
The results are taken on the basis of some inputs containing redundant bytes as plain text which can prove its strength over statistical attacks.
Plain Text 1:
Text Input: AAAAAAAAAA
Password: abcd




```
----------------------------------------
Final Message to be Transmitted :
----------------------------------------
769,389,64,457,1357,467,6266,1059,707,997
```

Fig. 2: Portion of the final transmittable encrypted text generated

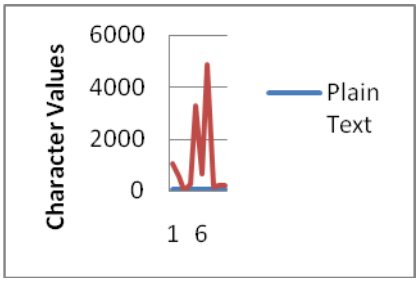
Plain Text 2:
Text Input: AAAAAAAAAB
Password: abcd




```
----------------------------------
Final Message to be Transmitted :
----------------------------------
679,19,6,1147,6489,497,3598,117,857
```

Fig. 3: Portion of the final transmittable encrypted text generated

Plain Text 3:
Text Input: 1111111111
Password: abcd




```
----------------------------------------
Final Message to be Transmitted :
----------------------------------------
1081,549,24,289,3305,679,4874,181,241,205
```

Fig. 4: Portion of the final transmittable encrypted text generated

Plain Text 4:
Text Input : AA
Password: abcd

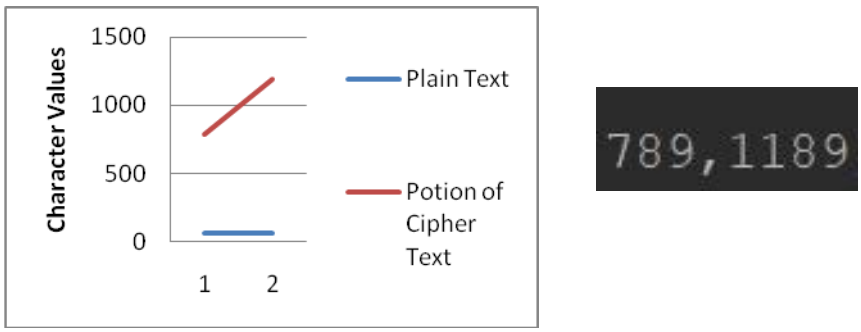


Fig. 5: Portion of the final transmittable encrypted text generated

Plain Text 4:
Text Input : AB
Password: abcd

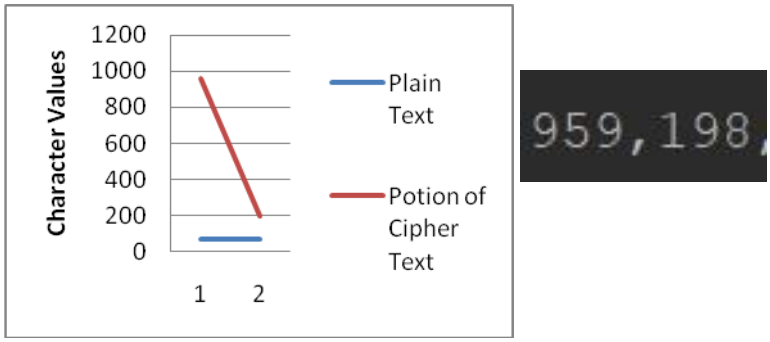


Fig. 6: Portion of the final transmittable encrypted text generated

Small portions of the encrypted texts were taken under study to find the nature of the randomness present in the finally transmittable message which led to the following results. This is part of the encryption process where the encryption text has been "1111111111" and the password has been "abcd". The x-axis of the graph in Fig. 6 represents the consecutive characters taken under consideration and the y-axis represents the encrypted Unicode character values taken under consideration.

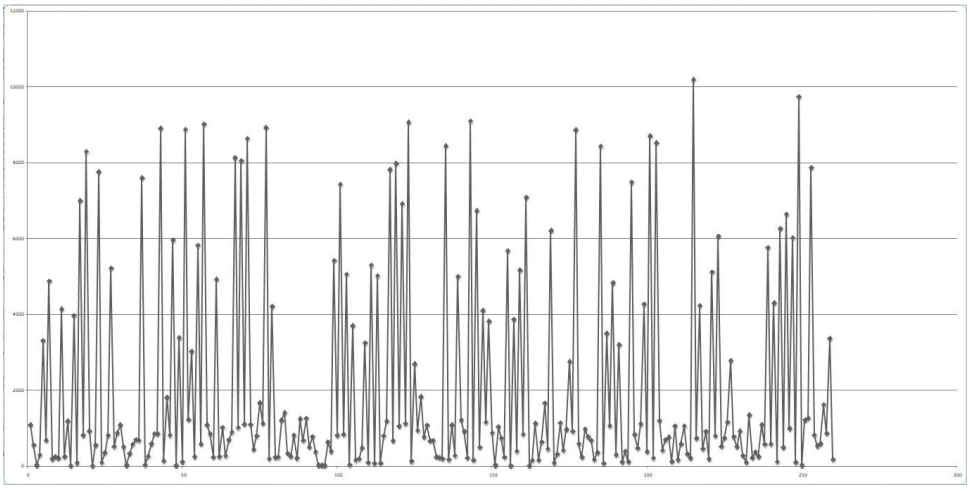


Fig. 6: Graph representing the study performed to find the randomness of any trivial message transmitted using a trivial password

5. CONCLUSION AND fUTURE SCOPE

The algorithm have been tested for considerable amount of results and it has proven its strength to cope with the growing attacks performed on encrypted messages to entrap the user's privacy. Although the large output size against a considerably smaller input plain text might be a hindrance in sending large encrypted message over the internet, the security of the procedure have never been compromised at any stages of the encryption procedure. The size of the output texts shall be considered as one of the main targets on the next version of the algorithm

6. REFERENCES

[1] The book of the spider, Paul Hillyard, ISBN 0 679-40881-9

[2] Spider silk - Structure, properties and spinning, D. Saravanan, Journal of textile and apparel, technology and management, volume 5, Issue 1, winter 2006

[3] Studies on structure and properties of nephila-spider silk dragline, Raju SeenivasanRengasamy, Manjeet Jassal and Chidambaram Rameshkumar, AUTEX Research Journal, Vol. 5, No1, March 2005

[4] Spider Silk Fibers Spun from Soluble Recombinant Silk Produced in Mammalian Cells, AnthoulaLazaris, et al.,The Journal of Experimental Biology 202, 3295–3303 (1999) 3295

[5] The mechanical design of spider silks: from fibroin sequence to mechanical function, J. M. Gosline, P. A. Guerette, C. S. Ortlepp and K. N. Savage, The Journal of Experimental Biology 202, 3295–3303 (1999)