ISSN 2320 - 2602

Volume 3, No.4, April 2014

International Journal of Advances in Computer Science and Technology

Available Online at http://warse.org/pdfs/2014/ijacst08342014.pdf

ENCRYPTING A BLOCK USING PARVALAY ALGORITHM

Sharad Jash[1], Suvajit Dutta[1], Tanumay Das[1], Debasish Patra[1], Dr.(Prof.) Pranam Paul[2]

[1]Student, MCA, Narula Institute of Technology, Agarpara, Kolkata-109, West Bengal, India, sharad99jash@gmail.com

[2]Student, MCA, Narula Institute of Technology, Agarpara, Kolkata-109, West Bengal, India, suvajit.brj@gmail.com

[3]Student, MCA, Narula Institute of Technology, Agarpara, Kolkata-109, West Bengal, India, debnit90@gmail.com

[4]Student, MCA, Narula Institute of Technology, Agarpara, Kolkata-109, West Bengal, India, tanumay.das@gmail.com

[5]Computer Application, Narula Institute of Technology, Agarpara, Kolkata-109, West Bengal, India, pranam.paul@gmail.com

ABSTRACT

Nowadays, the computer network has changed the mode of people's communication. People can easily transfer the various information through the network. However, because of the openness of the network, people have to take more and more attention on security and confidentiality of information. Data encryption is widely use to ensure security of the data. So encryption implies a way or procedure to hide some secrets by either applying some mathematical or logical functions to a plain text to produce a text that will be difficult for some outsider to decrypt.

Here we introduced a new algorithm based on symmetric key block encryption technique. In this algorithm encryption is done on binary file. Since this encryption is done on binary file so this algorithm is applicable for any data such as text as well as multimedia data.

KEYWORDS

Cryptography, Encryption, Decryption, Plain Text, Cipher Text, Cipher Block Chaining (CBC), Initialization Vector (IV), parabola.

1.INTRODUCTION

In modern world, security is a big issue and securing important data is very essential, so that the data can't be intercepted or misused for illegal purposes. For instance we can assume the situation where a bank manager is instructing his subordinates to credit an account, but in the mean while a hacker infer the message and he uses the information to debit the account instead of crediting it. This can be highly fatal and can cause too much destruction. So, different cryptographic methods are used by different organizations and government institutions to protect their data.

We introduce a block based symmetric key encryption algorithm. For encryption a key is to generate. Key length and bit stream is chosen at random. The algorithm emphasis on the properties of a conic section i.e. parabola and its directrix in order to encrypt a plain text. The word 'Parvalay' is a Sanskrit word whose English translation is 'parabola'. The value to be encrypted is shifted with the value of directrix which is further obtained by intersection of a line and parabola. To enhance the security of this algorithm the file is converted into a binary file and broken into segments of n-bits where n is the number of bits of encryption we are trying to achieve.

2. Cipher-block chaining (CBC)

IBM invented the cipher-block chaining (CBC) mode of operation in 1976.In CBC mode, each block of plaintext is XORed with the previous cipher text block before being encrypted. This way, each cipher text block depends on all plaintext blocks processed up to that point. To make each message unique, an initialization vector must be used in the first block. The CBC mode of encryption is depicted in the figure 1.

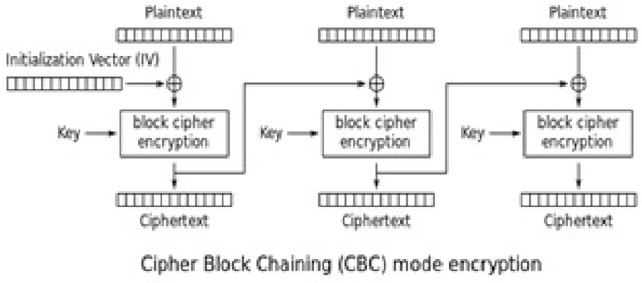


Figure 1: Cipher Block Chaining (CBC) mode encryption

CBC has been the most commonly used mode of operation. Its main drawbacks are that encryption is sequential (i.e., it cannot be parallelized), and that the message must be padded to a multiple of the cipher block size. One way to handle this last issue is through the method known as cipher text stealing. Note that a one-bit change in a plaintext or IV affects all following cipher text blocks.

If the first block has index 1, the mathematical formula for CBC encryption is

$$C_i - E_K(P_i \oplus C_{i-1}), C_0 - IV$$

The initialization vector for the algorithm is the slope of the line (say m) and the constant.

3. ALGORITHM

3.1.1 Structure Of Key

For encryption the key consists of the following sets of values:

Step 1: **n**: It is the number of bits over which decryption has to be carried out.

Step 2:**N1'**: Number of unchanged bits (if any).

Step 3: **N2'**: value of directrix from the equation x+a =0 where parabola intersects the line which further is shifted to produce decrypted value(s).

3.1.2 Sub Key Generation from Key

Step 1: calculate the value of directrix of parabola (say a).

Step 2: Again calculate the value of discriminant (say d) by solving the equation of parabola and line.

Step 3: Find the real roots of the equations. The two equations will give two values of x and y respectively.

Step 4: Find the actual sub key(say S_k) from the mathematical equation

$$S_k = \sqrt{(bx^2 - ax^2)} - (by^2 - ay^2)$$

(a)if S_k = = 0 then set $S_{k=}$((by-ax)+(ay-bx))/2
(b) else do nothing.

Where (ax,ay) and (bx,by) are the two roots obtained from the equations as shown in figure 2.

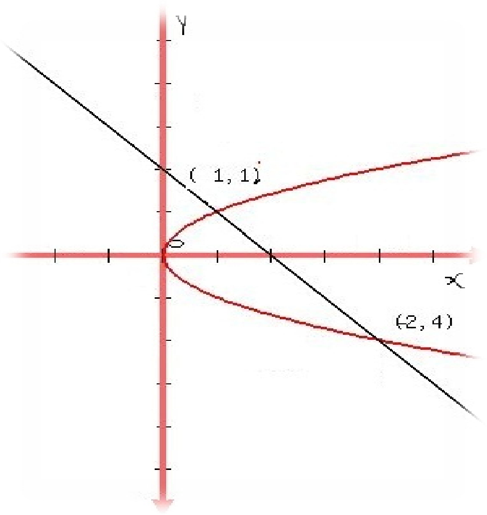


Figure 2: Intersection of a line and a parabola from where the key is to be evaluated.

3.2 Encryption Algorithm

Step 1: Convert the file to be encrypted into a binary file.

Step 2: Segment the binary into binary streams of n bits.

Step 3: Find the decimal equivalent of the segments formed and divide each segment s into two parts in the following manner $X = \frac{s}{10}$ and Y=s%10 $and\ the\ Intialisation\ vector(IV)\ i.e$ the value of slope(m) and constant c of a line.

Step 4: Find the value of a(value of directrix) from above values of x and y using parabola's equation $Y^2 = 4aX$ (Since we are assuming that the parabola is at the centre).

Step 5: Find the intersection value of the line and a parabola from equation y' = mx' + c and $Y^2 = 4aX$ say P (ax, ay) and q(bx,by) such that ax = (4*a) + / (2 * m*m) and bx = (4*a) - (4*$\sqrt{d}$) / (2 * m*m).

Where d is discriminant of value d = $b^2 - (4 * a * c)$

Step 6: Find the encrypted value and the sub-key from which the plain text is to altered i.e.,db = $\sqrt{(bx^2 - ax^2)} - (by^2 - ay^2)$.

if the value of db ≅ 0 then db = ((bx-ax)+(by-ay)) / 2.

Step 7: The encrypted value say C’ = X $\ll db$ and C” = Y $\ll db$.

Step 8: Check for any bits remaining of unordered size i.e. less than size of n.

Step 9: We convert the encrypted value C’ and C’’ into binary stream of n bits and write to the encrypted file.

3.3 Decryption Algorithm

Step 1: We read the binary streams in Key file in segments of n1’ number of bits.

Step 2: We take the size of encryption and size of unchanged bits from the key file.

Step 3: We read the binary streams in encrypted file in segments of n’ number of bits.

Step 4: We further divide the segment of binary streams into 2 halves of size of (n’/2) number of bits and we mark the decimal of the 1st half as x and 2nd half as y.

Step 5: Convert the decimal equivalent of two halves and altered the value in the following manner.

P’ = x $\gg$ k1 and P’’ = Y $\gg k2$

Where k1 and k2 are sub parts of a key.

Step 6: The binary streams thus formed are broken into segments of 8-bits binary stream.

Step 7: Decimal equivalent of the 8-bits blocks are written into the decrypted file.

4. Example

We consider a simple plain text – **Hello World**

4.1 Encryption

Step 1: Binary equivalent of this text is,

0100100001100101011011000110110001101111001000000 1
1101110110111101110010011011000110010 0
whose length is :88

Step2: We now take the key as follows, n=32 and Initialize IV values i.e. m=4 and c=7

We now segment the file in 32 bits segments we get two segments as follows

P1 = 0100100001100101

P2 = 0110110001101100

Step 3: Decimal equivalent are dP1=18533 and dP2= 27756 respectively.

Step 4 : Two coordinate pairs are: (0,7) and (0,7) using db = ((bx-ax)+(by-ay))/ 2.

Step 5: sub –key1 = 7.

Step 6: CIPHER VALUE (decimal):2372224

C’ – 00000000001001000011001010000000 And C’’ (decimal):3552768

C’’- 00000000001101100011011000000000 written to file.

Again, another segment of 32 bits is taken from the file and the process is repeated again i.e. 01101111001000000111011101101111 and two segments are

P1(28448) = 0110111100100000 and P2(30575) = 0111011101101111

We get the Cipher Value: C’– 00000000011011110010000000000000
and C’’ – 00000000011101110110111100000000.

Step 8: Again, we check the length of file it shows the remaining bits left for encryption i.e. 011100100110110001100100 since it is less than the size of n.

Step 9: These bits are left unchanged and written in the encrypted file maintaining an appropriate manner.

4.2 Decryption

Step 1: we first read the key file and first determines the size of encryption i.e. value of n =32.

Step 2: Then we need to find out if there is any unchanged bits or not but in this case we get Remaining Bits: 0010000011000.

Step 3: Now a data block of size n is to be retrieved from encrypted file until it is not empty

Step 4: So we get
C1(2372224):00000000001001000011001010000000
C2(3552768):00000000001101100011011000000000
C3(7282688):00000000011011110010000000000000
C4(7827200):00000000011101110110111100000000

Step 5: Number of Sub-Keys:4

Key1(7):0000000000000111
Key 2(7): 0000000000000111
Key 3(8): 0000000000001000
Key 4(8): 0000000000001000

Plain Text obtained using Sub-keys and Cipher Texts are

Step 6:

P1(18533)=0100100001100101
P2(27756)=011011000110110
P3(28448)=011011110010000
P4((30575) = 0111011101101111

We further Divide 30575) = 0111011101101111

Step 7: Entire Plain Text in Binary Form :

01001000011001010110110001101100011011110010000001
11011101101111011100100110110001100100
these entire block of size 8 bits to get the decimal equivalent character symbol.
I.e. **Hello world**

5. ANALYSIS

In this algorithm encryption is perform on binary data. All data which is under stable by the computer is finally converted into binary bits. So it can be implemented for any data type encryption process.

In this algorithm the length of the plane text is not restricted so it can be applicable for any large file.

5.1 Size Comparative Report

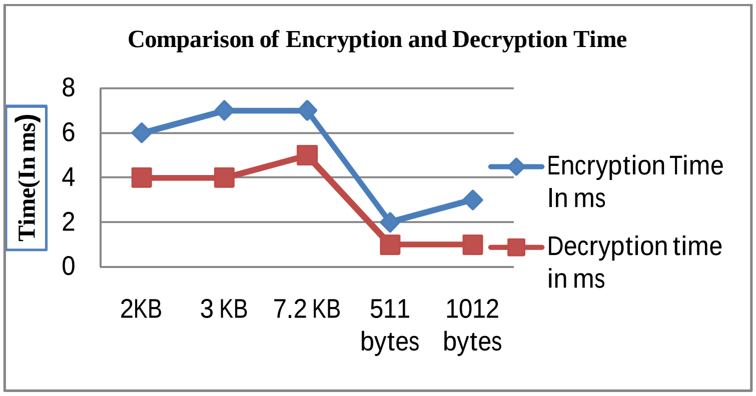
Files on which the algorithm is being implemented are given below in the table 1

Table 1 Set of Files with Size

Serial No	File Name	Encrypted File Size	Decrypted File Size
1	Prime.c	2KB	2KB
2	Input.txt	3 KB	3 KB
3	Pic1.bmp	7.2 KB	7.2 KB
4	Reverse.java	511 bytes	511 bytes
5	Index.html	1012 bytes	1012 bytes

5.2 Time Comparative Report

In Table 5.1.1, a comparison on basis of encryption time and decryption time with their file size has been shown here on the same set of files used in the following graph 1



Graph1 Comparison of Encryption and Decryption Time

6. CONCLUSION

Our conclusion towards the algorithm is that we have tested this algorithm and this algorithm worked correctly for the above set of values. From this we can assume that algorithm can correctly be implemented for very large values of m, c and obviously n . Since very large values can be used for encryption and decryption and large files can be encrypted using the same. Hence we say our algorithm is quite secured.

REFERENCES

[1] Pranam Paul, Saurabh Dutta, "A Private-Key Storage-Efficient Ciphering Protocol for Information communication Technology", National Seminar on Research Issues in Technical Education (RITE), March 08-09, 2006, National Institute of Technical Teachers' Training and Research, Kolkata, India.

[2] William Stallings- "Cryptography and network security principles and practices",4th edition - Published in 2006 By Pearson Education, Inc., publishing as Prentice Hal.

[3] Behrouz A.Farouzan –" Cryptography and Network Security"–ISBN: 9780073327532 Edition: Pub Date: 23-AUG-10

[4] Block cipher mode of operation - Source-http://en.wikipedia.org

[5] Pranam Paul, Saurabh Dutta, A K Bhattacharjee; **"**An Approach to ensure Security through Bit-level Encryption with possible Lossless compression**"; **Accepted and Published in "International Journal of Computer

Science and Network Security”, Vol. 08 No. 2, ISSN 1738 – 7906, IJCSNS, South Korea, pp 291 – 299.

[6] Symmetric key Cryptography using two-way updated – Generalized Vernam Cipher method:TTSJA algorithm, International Journal of Computer Applications(IJCA, USA), Vol 42, No. 1, March, Pg: 34 -39(2012).

[7] Symmetric Key Cryptography using Random Key generator : Asoke Nath, Saima Ghosh, Meheboob Alam Mallik: "Proceedings of International conference on security and management (SAM'10" held at Las Vegas, USA Jull 12-15, 2010), Vol-2, Page: 239-244(2010).

[8] Neeraj Khanna, Dripto Chatterjee, Asoke Nath and Joyshree Nath. Article: Bit Level Encryption Standard (BLES): Version-I.*International Journal of Computer Applications* 52(2):41-46,August2012.
Published by Foundation of Computer Science, New York, USA.