Volume 3, No. 1, January 2012
Journal of Global Research in Computer Science
ISSN-2229-371X
RESEARCH PAPER
Available Online at www.jgrcs.info

A NOVEL SCHEME FOR TEXT DATA ENCRYPTION

Monodeep Banerjee[1], Saptarshi Naskar[*2], Krishnendu Basuli[3], Samar Sen Sarma[4]
[1 and 2]Dept. of Computer Science, Sarsuna College
sapgrin@gmail.com
[3]Dept. of Computer Science, West Bengal State University
[4]Dept. of Computer Science & Engg., University of Calcutta

Abstract: Cryptography is used to make users convinced that their messages had not been tainted during transmission. However, it relied on mutual trust between the communicating parties. The application of cryptography is such as Key Agreement, Data Encryption, Data Decryption and Digital Signature. Our approach is quite different we have encrypted the text data by the efficient compression method. Were it is quite difficult to decrypt unless until we know the particular compression technique.

INTRODUCTION

The cryptography is compulsory because when a shared secret can be recognized between two communicating parties online by exchanging only public keys and public constants if any [1]. Any third party, who has access only to the exchanged public information, will not be able to calculate the shared secret unless it has access to the private key of any of the communicating parties [1]. The two diplomacy connecting in the internet can exchange data between them in some encrypted format that the private data can't be access by some third party until these two devices give the permission two exchanges of data [2,3].

The two most widespread uses in cryptography are, probably, to store data securely in a computer file or to transmit it across an insecure channel such as the Internet. In either scenario the fact that the document is encrypted does not prevent unauthorized people gaining entrée to it but, rather, ensures that they cannot comprehend what they see.

The information to be secret is often called the operation of disguising it is known as *encryption.* The encrypted plaintext is called the *cipher text* or *cryptogram* and the set of rules used to encrypt information plaintext is the *encryption algorithm.* Usually the operation of this algorithm depends on an *encryption key*, which is input to the algorithm together with the message [3]. In order that the recipient can obtain the message from the cryptogram there has to be a *decryption algorithm* which, when used with the appropriate *decryption key*, reproduces the plaintext from the cipher text [4].

Cryptography is the science of manipulative of cipher systems, whereas *cryptanalysis* is the name given to the process of deducing information about plaintext from the cipher text without being given the appropriate key. *Cryptology* is the collective term for both cryptography and cryptanalysis [5,6,7].

Cryptography can be divided into two parts:

a. Symmetric key cryptography or Private Key cryptography.
b. Asymmetric key cryptography or Public Key cryptography.

SALIENT FEATURES OF THE ALGORITHM

Usually the text data are encrypted by the different known methods. Where some key, either they are public or private, are used to encrypt the cipher text. But here we have used the data compression method to encrypt the text.

ALGORITHM: ENCODE

Begin:
Call procedure Encode.
Call procedure Decode.
Stop.
Procedure: Encode
Begin:
Step1 Set X: = 0
Step2 repeat step3 to step7 while X<8
Step3 Set Y: =0
Step4 repeat step5 to step6 while Y<8
Step5 B[X][Y]: =0
Step 6 Y: =Y+1
[End of inner while.]
Step7 X: =X+1
[End of outer while.]
Step 8 Set I:=0
Step 9 repeat step 10 to step 40 while C[I]≠'\0'
Step10 Set J: =I
Row =0
K=0
Step11 repeat step 12 to step 20 while C[J]≠' ' and C[J]≠'.' and K<8
Step12 If[C[J]≥65 and C[J]≤90] then,
Step 12(i) C[J]=C[J]+32
[End of if]
Step 13 ASCII: =C[J]
Step 14 Set COLOUMN: =7
Step 15 repeat step16 to step18 while ASCII>1
Step 16 B[ROW][COLOUMN]= ASCII%2
Step 17 ASCII =ASCII/2
Step 18 COLOUMN: =COLOUMN-1
[END OF WHILE]
Step 19 B [ROW] [COLUMN]=ASCII
STEP20 Set J=J+1
ROW=ROW+1, K=K+1
[End of while]

Step21 Set COLOMN:=7
Step22 Repeat Step 23 to Step29 While COLOMN ≥ 0
Step23 Set ROW1=0, ASCII=0
Step24 Repeat step 25 to step 26 while ROW1< ROW
Step25
ASCII=ASCII+(B[ROW1][COLOUMN]*pow(2,row1)
Step26 Set Row1=Row1+1
[End of the while]
Step27 If [ASCII≠(pow(2,Row)-1) and ASCII≠0) then,
Step27(i) A[Z=Z+1]=ASCII
[End of if]
Step28 Set COLOMN=COLOMN-1
[End of while]
Step29 Set COLOMN=7
ASCII=0
Step30 Repeat Step 32 to Step 33 while COLOMN≥3
Step31 ASCII=ASCII+(B[0][COLOMN]*Pow(2,7-COLOMN)))
Step32 COLOMN=COLOMN-1
[End of while]
Step33 A[Z=Z+1]=ASCII
Step34 If[K=8 and C[J]≠' ' or k=8 and C[J]='.'] then,
Step 34(i) A [Z=Z+1] =' '
Step 34(ii) A [Z=Z+1] =' '
[End of if]
Step34 Else if[C[J]=' ' or C[J]='.'] then,
A[Z=Z+1]=C[J]
Step35 Set i=j
Step36 Set i=i+j
[End of outer while]
Step37 A[Z]='\0'
STOP [END]

ALGORITHM: DECODE

Begin:
Step 1 Set I :=0
Step 2 Return step3 to Step8 while I<8
Step 3 Set J=0
Step 4 return step 5 to step 8 while J<8
Step 5 If [J=1 or j=2] then,
Step 5(i) B[I][J]=1
End of If
Step 6 Else B[I][J]=0
End of Else
Step 7 J=J+1
[End of while]
Step 8 I=I+1;
[End of while]
Step 9 set I:=0
Step 10 repeat step 11 to step while A[I] ≠ '\0'
Step 11 set J:=I
Step 12 repeat step 13 to step 14 while A[J] ≠ ' ' and
A[J] ≠' .' and
A[J] ≠ '\0 ' and
Step 13 ASCII =A[J-1]
Step 14 J:=J+1
[End of While]
Step 15 COLUMN:=7
Step 16 repeat step 17 to step 19 While ASCII > 1
Step 17 B[0][COLUMN] = ASCII % 2
Step 18 ASCII = ASCII / 2
Step 19 COLUMN := COLUMN-1
[End of While]
Step 20 B[0][COLUMN] = ASCII
Step 21 COLUMN :+7
Step 22 repeat step 23 to step while I ≤ (J-2)
Step 23 ASCII = A[I]
Step 24 repeat step 25 to step 29 While COLUMN ≥ 3
Step 25 If [ASCII % 2 = B[0][COLUMN]] then,
Step 25(i) break
[End of If]
Step 26 Else if [B[0][COLUMN] =1] then,
Step 26(i) set ROW :=0
Step 26(ii) repeat step 26 (ii) to step 26(iii) While ROW ≤ 8
Step 26(iii) B[ROW][COLUMN] = 1
Step 26(iv) ROW :=ROW+1
End of While
Step 27 K=Pow (2, (7-COLUMN))
Step 28 Flag :=1
[End of Else If]
Step 29 COLUMN:= COLUMN-1
[End of While]
Step 30 If [COL0UMN ≠ 2]
Step 30(i) ROW:= 0
Step 30(ii) repeat step 30(iii) to step 30(v) While ASCII>1
Step 30(iii) B[ROW][COLUMN] = ASCII / 2
Step 30(iv) ASCII := ASCII/2
Step 30(v) ROW :=ROW+1
[End of While]
Step 31 B[ROW][COLUMN]=ASCII
Step 32 COLUMN :=COLUMN-1
[End of If]
Step 33 I:=I+1
Step 34 ROW :=0
Step 35 repeat step 36 to step 42 While ROW ≤ 8
Step 36 COLUMN :=7
ASCII :=0
Step 37 repeat step 38 to step 39 While COLUMN ≥ 0
Step 38 ASCII=ASCII+ (B[ROW][COLUMN] * Pow(2,(7-COLOUMN)))
Step 39 COLUMN:=COLUMN -1
[End of While]
Step 40 If [FLAG=1] then,
Step 40(i) If [ASCII > 96+K] then,
Step 40(ii) C [Z:=Z+1]=ASCII
[End of If]
[End of If]
Step 41 Else
Step 41(i) If [ASCII > 96]
Step 41(ii) C[Z:=Z+1]=ASCII
[End of Else]
Step 42 ROW :=ROW+1
[End of While]
Step 43 I=J
Step 44 If [I≠ (J+1)]
Step 44(i) C [Z=Z+1] = A[I]
[End of If]
Step 45 I: =I+1
[End of while]
STOP[END]

CONCLUSION

Our approach is novel in the sense that no other algorithm exists that encrypts the text matter by the using of compression technique. The complexity of encryption is in

the order of $O(n^2)$. And the decryption is of the order of exponential. We are trying for the newer approach for the same.

REFERENCE

[1]. Barry Nance, Network Programming in C, TMH, 1998.

[2]. Allen L. Wyatt, Using Assemble Language, World Scientific, 2001.

[3]. Douglas V. Hall, Microprocessor and Interfacing Programming and Hardware, TMH, 1999.

[4]. S. V. Sathyanarayana, M. Aswatha Kumar and K. N. Hari Bhat, Symmetric Key Image Encryption Scheme with Key Sequences Derived from Random Sequence of Cyclic Elliptic Curve Points, International Journal of Network Security, Vol.12, No.3, pp.137-150, May 2011.

[5]. Goldberg, Ian, Mashatan, Atefeh, Stinson, Douglas R, On message recognition protocols: recoverability and explicit confirmation, International Journal of Applied Cryptography, Volume 2, Number 2, January 2010, pp. 100-120.

[6]. M. Abdalla, M. Bellare and G. Neven, Robust Encryption, Proceedings of the 7th Theory of Cryptography Conference (TCC 2010), Lecture Notes in Computer Science Vol. 5978, D. Micciancio ed, Springer-Verlag, 2010.

[7]. M. Bellare and A. Palacio Towards Plaintext-Aware Public-Key Encryption without Random Oracles, Advances in Cryptology - Asiacrypt 2004 Proceedings, Lecture Notes in Computer Science Vol. 3329, P. J. Lee ed, Springer-Verlag, 2004.

