New Way for Encryption Data Using Hourglass

A. Hamid Mehdi
Department of Computer, Andimeshk Branch, Islamic Azad University, Andimeshk, Iran
hamidmehdi@gmail.com

Abstract

Nowadays, there are many encryption algorithms to protect information. Data confidentiality is one of the most important functions of encryption algorithms, it means when the transferring data between different systems is vague for unauthorized systems or people. Moreover Encryption algorithms must maintain data integrity and provide availability for information. New encryption methods cause the attackers cannot simply access to the information and do not allow discovering the relationship between information and the encrypted one. Therefore availability can be difficult for them. Existing complexities make their longevity and effectiveness increase (Mandal, 2012).In This Article, It has been tried to present an encryption method which has the characteristic of encryption algorithms and also has some unique complexities which are not easily detectable and efficient.

Keywords: Encryption, decryption, symmetric algorithm, hourglass algorithm.

I. Introduction

Encryption algorithms are divided into two parts (Shanta, 2012) the first public-key algorithms such as RSA. The Second private key, this part is divided into two categories in turn. The first sub branch is stream ciphers and the second sub branch is blocks ciphers. Renowned algorithms such as DES (Shanta, 2012), 3DES (Pavithra et al. 2012), AES (Thakur et al. 2011), Blowfish (Agrawal et al. 2010), EABC (Mehdi, 2013) can be mentioned as some instances for Block ciphers. Encryption algorithms have advantages and disadvantages but what is of great important is, they are beneficial. The Main differences of encryption algorithms are computational algorithm in computation time, memory consumption and output bytes.

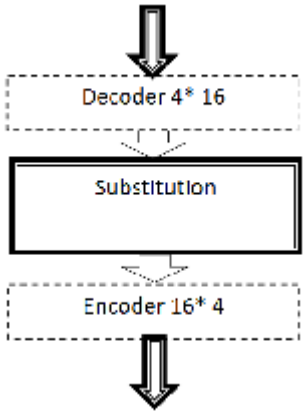

Figure.1. substitution 4 bits (Stalling, 2005)

Public key algorithms are known as asymmetric algorithms and private key algorithms are known as symmetric algorithms. Symmetric encryption algorithms are used mainly from Feistel method. These methods combine substitution and permutation simple method, then methods are combined and will achieve complex and safe algorithm. The primitive principle of these algorithms is shown in figure 1.

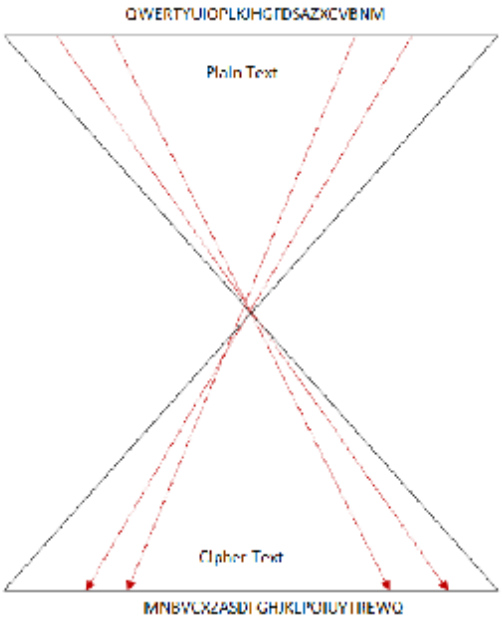

Figure.2. Proposed Algorithm

These methods have two important specifications to eliminate attacks based on statistical analysis. Diffusion and confusion, Diffusion tries to diffuse plain text in all of cipher text as changing in one bit of plain text causes changing to a large extent of bits in cipher text. It is obtained from the combination of substitution and permutation bits in the plain text. Confusion tries to increase the complexity between cipher text and plain text so that attackers do not achieve plain text with structural analysis of cipher text (Mandal, 2012).

Something which causes to use encryption algorithm in a wide range is the inner complexity of algorithm which does not easily break. According to mentioned phrases, Article tries to peresent an algorithm that has enough complexity and also it can be used easily.

The remainder of this paper is organized as follows. Section II presents the method used to encryption (EDH). Section III describes the sample used in our experiments. In Section IV, article presents algorithm codes. Concluding remarks are made in Section V.

I. Proposed Algorithm

Proposed algorithm, which is called encryption algorithm using hourglass (EDH), is shown in figure 2. This method is based on Feistel and has two parts, plain text and cipher text.

Algorithm is designed by a circle, top semicircle is Plain text and another semicircle is cipher text. Algorithm includes 26 circles for any characters in alphabet. Any top semicircles are for character in plain text and any bottom semicircles are a character in cipher text.

The algorithm has two Keyes. The first key to achieve plain text is called FPART and a key for achieve cipher text is called SPART. At the First stage, the first character must be indicated in plain text via FPART and must find its position then must find peer to peer position in cipher text. Now, the following characters in FPart and SPart must be permuted in terms of following conditions.

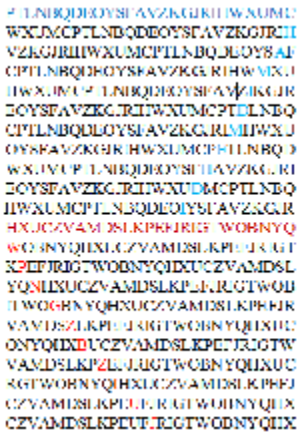

Figure.3. Algorithm Structure - Encryption/Decryption

If position character in plain text is odd, permutation must be continued while reaching character to the end of FPART but in even cases permutation continues while reaching character to the first position in FPART and also in SPART if the position of calculated character was odd in plain text, permutation must be continued while character reaches to the first in SPART but in even cases permutation continues while character reaches the second position in SPART. For any characters in plain text is generated a character in cipher text and also an expression is generated and adds for increasing the complexity and redundancy to all of cipher text in order to make algorithm difficult. Now, we illustrate algorithm.

Il. lustrate the algorithm

Assume FPART and SPART are below values:

FPART=PTLNBQDEOYSFAVZKGJRIHWXUMC
SPART= HXUCZVAMDSLKPEFJRIGTWOBNYQ

And the plain text is HAMIDMEHDI. Now, must be achieved cipher text for it with proposed algorithm. Algorithm explains is as follows:

1- Finding the first character position of plain text in FPART:
FPART=PTLNBQDEOYSFAVZKGJRI**H**WXUMC

2- Finding peer to peer the first character position of plain text in SPART:
SPART= HXUCZVAMDSLKPEFJRIGT**W**OBNYQ

3- Permutation FPART and SPART while given character reaches to the first in FPART and SPART.
FPART=**H**WXUMCPTLNBQDEOYSFAVZKGJRI
SPART=**W**OBNYQHXUCZVAMDSLKPEFJRIGT

4- Now, when character position increase, in FPART must be permutation continued while character reaches the last position then target character must insert at row number position. Also this operation must do by SPART but reverse.

All of the steps are shown in figure 3. In figure 3 blue texts in the first row are FPART and red texts in the first row are SPART and all of green texts are extra text. From the second row to last from right which is defined with blue color, shows plain text and from the second row to last from left with red color is cipher text. Finally cipher text of HAMIDMEHDI is WPNGZBZIUJ. Therefore when there are FPART and SPART and cipher text, the plain text can be decoded and discovered.

II. Algorithm pseudo Codes

This algorithm includes two main functions encryption and decryption which are designed by C#.Net.

a. Encrypt Code

First function is encryption and it gets plain text as input and output of this function is cipher text. Encryption code is as follows:

```
string FPART = MDSLKPEFJYQHXUCZVARIGTWOBN";
string SPART= PYSHWTLNBQDEOXUMCFAVZKGJRI";
string Encrypt(string plainText){
char[] temp ;
char[] temp1 ;
int target;
for (int i = 0; i < plaintext.Length; i++){
Console.WriteLine("{0}  {1}", new string(FPART),new string(SPART));
target = Array.IndexOf (SPART, plaintext[i]);
// permute FPART
for (int j = target; j < 26; j++) temp[j - target] = FPART[j];
for (int j = 0; j < target; j++) temp[26 - target + j] = FPART[j];
temp.CopyTo(FPART, 0);
//---------permute SPART
if (i % 2 == 0){target++;}
for (int j = target; j < 26; j++) temp[j - target] = SPART[j];
for (int j = 0; j < target; j++) temp[26 - target + j] = SPART[j];
temp.CopyTo(SPART, 0);}
INSERT CHARACTER IN ROW NUMBER Position FROM RIGHT
```

b. Decrypt Code

Second function is Decryption and it gets cipher text as input and output of this function is plain text. Decryption code is as follows:

```
char[]FPART= FPART.ToCharArray();
char[] FPART1 = FPART.ToCharArray();
char[] SPART = sAlphabet.ToCharArray();
char[] temp ;
char[] temp1;
int target;
for (int i = 0; i < ciphertext.Length; i++){
Console.WriteLine("{0}  {1}", new string(FPART), new string(SPART));
target = Array.IndexOf (FPART, ciphertext[i]);
plaintext[i] = SPART[target];
// permute FPART
for (int j = target; j < 26; j++) temp[j - target] = FPART[j];
for (int j = 0; j < target; j++) temp[26 - target + j] = FPART[j];
temp.CopyTo(FPART, 0);
// permute SPART
 if (i % 2 == 0){target++; }
for (int j = target; j < 26; j++) temp[j - target] = SPART[j];
```

```
for (int j = 0; j < target; j++) temp[26 - target + j] = SPART[j];
temp.CopyTo(SPART, 0);
INSERT CHARACTER IN ROW NUMBER Position FROM LEFT
```

The most important advantage of EDH algorithm is changeability of methods for using in different forms. For example in many organization can be used while they cannot understand their algorithms because all of the functions in algorithm can be changed easily.
According to the expressed result, the most important advantage of this algorithm is changeability with less cost and its complexity. One of the advantages is redundancy in cipher text and paying attention to this point just cipher text is sent therefore the decryption of cipher is not easy. Also one another important advantage of this algorithm is the Keyes changeability, that is possible easily, therefore this point causes complexity to increase.

III. Conclusion

Nowadays, we live in the world that information faces to different dangers, robberies, misuses, etc. One of the most important methods to prevention these dangers is cryptography. This is important that any organizations must have a cryptography algorithm for themselves so that they become sure from information when transferring.
In this paper it has been tried to present an encryption algorithm which is user-friendly and has difficult encryption. This algorithm is quick in changing and redundancy and has wide range of permutation, that all of them help the complexity and finally increasing its safety.

References

i. Shashi Mehrotra Seth, Rajan Mishra on " Comparative Analysis Of Encryption Algorithms For Data communication " in IJCST Vol. 2, Issue 2, June 2011 I ,pp. 292-294

ii. William Stalling, Cryptography and network Security, 4th Edition, Prentice –Hall, 2005

iii. Jawahar Thakur, Nagesh Kumar, "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis, "in International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, Volume 1, Issue 2, December 2011), pp.6-12

iv. Shanta, yoti Vashishtha on, "Evaluating the performance of Symmetric Key Algorithms: AES (Advanced Encryption Standard) and DES (Data Encryption Standard) in IJCEM International Journal of Computational Engineering & Management, Vol. 15 Issue 4, July 2012 ,pp.43-49

v. Himani Agrawal and Monisha Sharma "Implementation and analysis various symmetric Cryptosystems "in Indian Journal of Science and Technology in Vol. 3 No.12 (Dec 2010) ISSN: 0974- 846, p.1173-1176

vi. S.Pavithra, Mrs. E. Ramadevi "STUDY AND PERFORMANCE ANALYSIS OF CRYPTOGRAPHY ALGORITHMS " International Journal of Advanced Research in Computer Engineering & Technology Volume 1, Issue 5, July 2012 14, pp.82-86

vii. Pratap Chnadra Mandal, "Superiority of Blowfish Algorithm", International Journal of Advenced Research in Computer Science and Software Engineering 2(9), Volume 2, Issue 9, September 2012, pp. 196-201

viii. Mehdi Hamid (2013) " Checking EABC performance in comparison others cryptography algorithms " International *Journal of Computer Science and Network Solutions*, Vol(1),pp 51-57