ISTANBUL UNIVERSITY –
JOURNAL OF ELECTRICAL & ELECTRONICS ENGINEERING

YEAR : 2004 (1057-1062)
VOLUME : 4
NUMBER : 1

A NEW APPROACH TO STREAM CIPHER: UNSYSTEMATIC CIPHER

[1]Oğuzhan TAŞ [2]Bilal ALATAŞ [3]Erhan AKIN

[1, 2, 3] Department of Computer Engineering, Faculty of Engineering, Firat University,
23119, ELAZIĞ

[1]E-mail : oguzhantas@firat.edu.tr [2]E-mail : balatas@firat.edu.tr [3]E-mail : eakin@firat.edu.tr

ABSTRACT

The encryption technique proposed in this study encrypts / decrypts in binary form. For each character in the plaintext, a randomly generated number (r), called added number, makes the cipher of the character in 2^r different ways. In addition to this process, the character in the plaintext is converted into binary form and divided into parts according to the values of the added number. Different from the One Time Pad Cipher, proposed as perfect cipher in the literature, different ciphertexts are obtained from the same key. This is one of the advantages of the proposed technique. Besides, its more structural simplicity in comparison with the other stream cipher algorithms makes its development in hardware and software easy.

Keywords: *Cryptography, Stream Cipher, Unsystematic Cipher.*

1. INTRODUCTION

The science of cryptography is used in many fields such as military and financial for confidentiality, integrity, privacy, and authentication for years. Processes in cryptography consist of five basic components. These are encryption function, decryption function, encryption key, ciphertext, and plaintext. Encryption functions produce different ciphertexts according to different values of keys. Consequently, an attacker who tries to break cipher cannot obtain the plaintext without the key. In cryptography, encryption function, decryption function, and ciphertext are not concealed while key and plaintext are concealed [1, 2, 3, 4]. Encryption function and decryption function have to be secreted when encrypting without key. When encryption function is known by the people different from the members of the group using this type of encryption, all of the members will have to reach an agreement on a new encryption function [1]. If the group uses the technique of encryption with key, by changing the key without changing encryption / decryption function, secure communication will again be provided. In cryptography, besides symmetric key systems [17, 18] in which the same key is used for encryption and decryption, asymmetric systems in which different keys are used for encryption and decryption have also been developed [1, 3, 4, 19]. In asymmetric systems, key distribution in communication of remote people is not a problem because of the use of different keys. However, in symmetric systems, encryption and decryption are quickly performed in spite of the problem in key distribution [15]. Presently, in security protocols such as SSL (Secure Electronic Transaction) [21], SET (Secure Socket Layer) [20], symmetric key systems are used. The technique proposed in this study uses symmetric key system.

Received Date : 30.04.2003
Accepted Date: 30.12.2003

In cryptography, there are two main types of symmetric algorithms called block cipher and stream cipher [14]. Block cipher, encrypts / decrypts by blocking data. Usually, 64-bit or longer length is selected for block length. In stream cipher, encryption and decryption are performed on one bit or one byte (sometimes 32-bit word) of plaintext and ciphertext. In block cipher, always the same block of ciphertext is obtained from the same block of plaintext by using the same key. Some examples of block cipher algorithms are DES [6], RC5 [7], SAFER [8], Blowfish [9], TEA [16], and FEAL [10]. In stream cipher, one bit or byte in the same plaintext is encrypted as different bit or byte in every time. Examples to this type of algorithms are RC4 [1], A5/1 [12], ORYX [5], and SEAL [11]. Block cipher is preferred more in software applications while stream cipher is more preferred in hardware applications because of its bit by bit process. Proposed encryption technique is developed for not only hardware applications but also for rapid software applications.

A cryptographic mode is a combination of the basic cipher, some sort of feedback, and some simple operations. Because the security is a function of the underlying cipher and not the mode, the operations are simple. Even more strongly, the cipher mode should not compromise the security of the underlying algorithm. There are many types of cryptographic modes such as ECB (Electronic Codebook Mode), CBC (Cipher Block Chaining), PCBC (Propagating Cipher Block Chaining Mode).

2. UNSYSTEMATIC CIPHER

The unsystematic cipher technique is an encryption technique which operates bit by bit. In this proposed technique, the same key (secret key, single key) is used for encryption and decryption. Below notations will be used throughout the paper:

P : Plaintext
C : Ciphertext
E() : Encryption Function
D() : Decryption Function
K : Key
N : Number of the characters in the plaintext
r : Added number
S : Number of bits in the ciphertext
y : Bits of the characters of the ciphertext
x : Bits of the characters of the plaintext
n : Number of bits used for each character in the plaintext
Character(): The function that returns the character equivalent of ASCII value

2.1. ENCRYPTION

In this encryption technique, first, the number of bits (n) used for each character in the plaintext is selected. This number may be selected as 8 or more, because the characters in the ASCII table are expressed by minimum 8 bits. Henceforth, the value of n for the examples will be 8 as default. After the selection of the value of n, a random added number (r) for the characters of the plaintext is selected. This number must be selected from the interval $1\leq r\leq n$. Let the characters in the plaintext be expressed as $P = \{p1, p2, p3, pn\}$, $\forall pi \in \{Character(1\ldots255)\}$, $1\leq i\leq N$, and let the bits of each character of the plaintext be expressed as $C = \{x1, x2, x3,\ldots, xn\}$ and $\forall xj \in \{0,1\}$, $1\leq j\leq n$. Consequently, the obtained bits of the ciphertext will be $\{y1, y2, y3,\ldots, ys\}$ and $\forall yt \in \{0,1\}$, $1\leq t\leq S$.

After the selection of the added number (r), binary form of the character of the plaintext is divided into r equal parts. When the characters are not divided exactly by the r values, (the situation in not being able to be divided into equal parts) the biggest parts will be placed in the beginning and the other equal parts will be placed in the end. For example, let the added number be 6. When dividing the 8-bit plaintext into 6 parts, first two bits will constitute the first part, the following two bits will be the other part, and remaining each 4 bit will constitute new parts. If the character in the plaintext is “F” (F = Character(70)), then the binary form will be 01000110. If the added number (r) is selected as 6, then character will be divided into parts as shown below:

01 → 1. part
00 → 2. part
0 → 3. part
1 → 4. part
1 → 5. part
0 → 6. part

Figure 1. Dividing of the bits of the plaintext

This process can be formulated. Let k be the remainder and q the quotient in the process of n/r. Then,

$$n = q * r + k \quad (1)$$

$$k = n \bmod r \quad (2)$$

$$k * (q + 1) \quad (3)$$

$$(r - k) * q \quad (4)$$

(2) shows that q+1 bits will be in k parts, and q bits will be in r-k parts. (1) is the addition of (3) and (4). The above formulations are valid for all values of n and r. For example if n = 8 and r = 6;
$k = 8 \bmod 6 = 2$
$q = 8 / 6 = 1$
$r - k = 6 - 2 = 4$

From here, it can be seen that two parts will involve two bits and following four parts will involve only one bit. After the process of division, the bits of the added number in binary form are added to the end of the binary form of each character. If the added number is 5, 2^5 different bit arrays can be added to the end of the binary form of the character. Thus, the character of "F" in the plaintext can be encrypted in 2^5 = 32 different ways for this added number. When using large values for the added number, the complexity of the encryption is increased and more secure results are obtained. In the other step, bit by bit change is performed in the parts according to the binary equivalent of added number. For example, if the bit in the binary equivalent of added number is "0" then involved part will be the same without any change. If it is "1" then the bits of the involved part will be complemented, that is the number 0s in the part will be converted into 1s and the 1s will be converted into 0s. For example, let the added number for the example in Figure 1 be selected as 6 and the value 101011. The first bit in this binary number is "1" and this means that the value of "01" in the first part will be "10". Then, because second bit of the number is "0", the value of "00" in the second part will remain the same. Because the other bit is "1", the value of "0" in the third part will be "1" and this process is continued in the same way. The values that the added numbers may take and the encryption of the plaintext according to these values can be clarified as shown in figures below:

When the added number r = 1, the character itself and the complementation of all its bits may be used to constitute the decrypted "F":

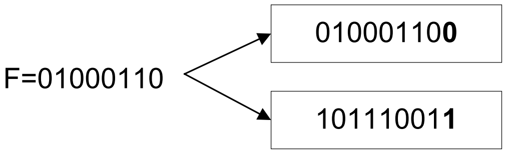


Figure 2. Plaintext and ciphertext with $r = 1$

In Figure 2, the length of one character becomes 9-bit with the added number. The last bit is the value of the added number. The bits remain the same in the situation of addition of 0, and complementation of all bits is performed in the part in the situation of addition of 1.

When r = 2, the binary form of the character is divided into two parts and the added number can have four different values (2^2=4). If the value is 00, all bits remain the same. If the value is 01, the bits of the first part remain the same and the bits of the other part are complemented. The bits of the first part are complemented and the bits of the second part remain the same if the value is 10, and all bits are complemented if the value is 11. Thus, the length of one character becomes 10-bit with the added number. The plaintext and the obtained different ciphertext according to these values are shown in Figure 3.

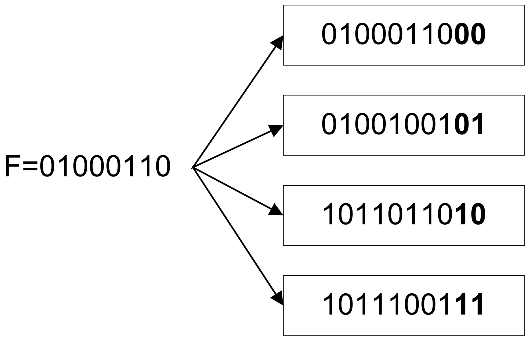


Figure 3. Plaintext and ciphertext with $r = 2$

When r = 3, the bits of the characters are divided into three parts as explained above and can be encrypted in 8 different ways (2^3=8) according to the values of the added number. The added numbers can have the values 000, 001, 010, 011, 100, 101, 110, and 111. In this situation, the character “F” can be expressed as 11 bits and encrypted in 8 different ways as shown in Figure 4.

This process can be generalized and different encrypted characters are obtained from the same character according to the r values.

In order to increase the confidentiality, unsystematic stream cipher may be used together with other cipher algorithms. The ciphertext of the other algorithms may be the plaintext of the proposed technique, or the ciphertext of the proposed technique may be the plaintext of the other algorithms. However, while this process increases the confidentiality, it increases the computational complexity.

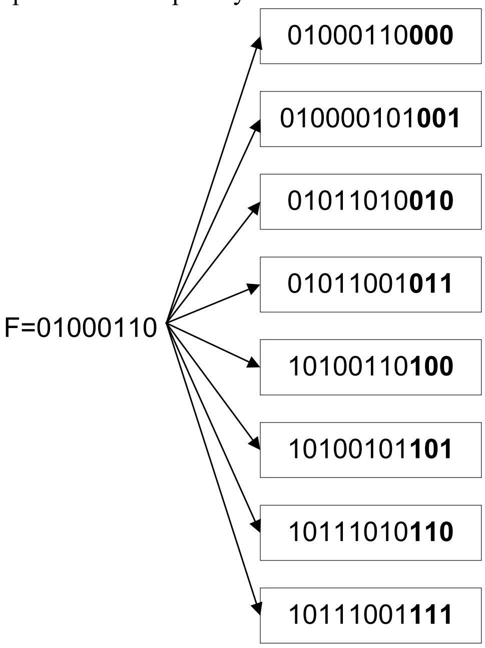


Figure 4. Plaintext (F) and ciphertext with r = 3

2.2. DECRYPTION

The process of decryption is performed by the same key used in the encryption. Because the numbers in the key show the number of bits added to the end of each character (added numbers) one by one, decryption can easily be performed by using this information.

In addition to value of the keys, the value of n must also be known for the process of decryption . If the value of n is selected large, the ciphertext will be more complicated and thus the security will increase. Although each character in ASCII table is expressed with 8 bits, if n is selected as 10, each character will be expressed with 10 bits and the length of the ciphertext will increase accordingly. After the value of n is known, the values in the key are sequentially added to the value of n. By this process, the number of bits that encrypts each character is understood. Then, bit arrays encrypted for each character from the ciphertext are sequentially taken and resolved. For example, let n = 8 and key be

54537142345614237564353223653152…

Let the ciphertext be,

101111010100111100010101011100011…

When decrypting, for resolving the ciphertext of the first character, the value of n, 8, and the first number of the key, 5 is added. The obtained value, 13 shows that the first character of the plaintext is encrypted with 13 bits. For the following character, the value of n and the second value of the key, 4 is added and it is understood that the bits in the interval from 14th bit to 26th bit are the ciphertext of the second character. This process is continued and the expression of the number of bits and bit arrays of all characters in the ciphertext are understood.

The bit array for the first character is found as 1011110101001 and the last 5 bits (added number) are analyzed. After this process, by using the equation (3) and (4) the 8 bits are divided into five parts as (2, 2, 2, 1, 1). Then, the inverse process of the encryption made before is performed according to the obtained values of added number, 01001. Because the first bit of the added number is “0”, the value “10” remains the same. Second bit is “1” and thus the value “11” is complemented and the value “00” is obtained. When this process is continued, the first character of the plain text 10001100 = î is found.

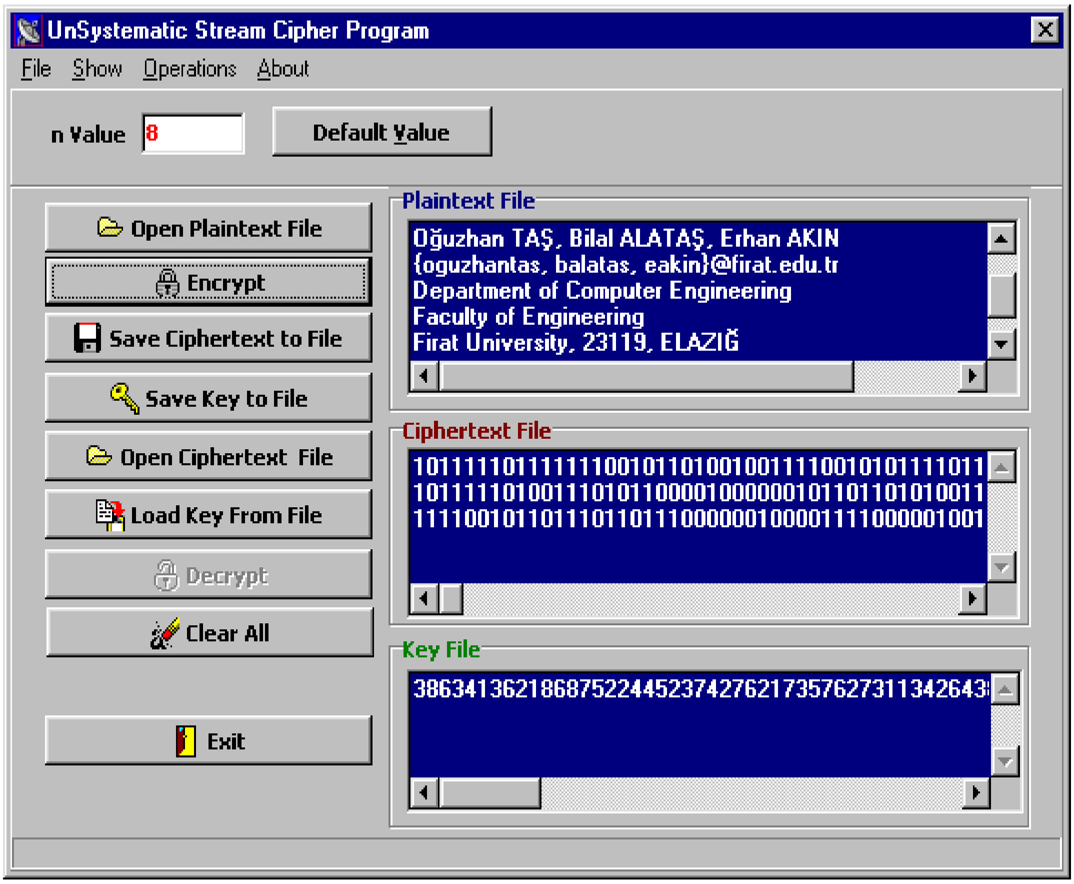
2.3. KEY

The key used for encryption and decryption is composed of the arrangement of the added numbers used for each character. Consequently, the length of the key is equal to the number of the characters in the plaintext. The longer the length of the key, the more complicated the process is. In this situation, the solution becomes harder and the security increases as in all cipher algorithms.

In the technique of One Time Pad, proposed as perfect cipher in the literature [13], encryption is performed by set of the keys which are not repeated and maximum security is obtained. However, this technique has two main disadvantages. It needs a complete synchronization between the sender and receiver, and needs generation of many unique keys. A control mechanism must check the keys every time to see whether these have been used or not. Besides, a database must be used for the storage of the keys used before [22]. Although these are problems for the technique of One Time Pad, they are not problems for the proposed technique, because even if the key is repeated, the contents of ciphertext will change.

3. APPLICATION

In order to apply the proposed cipher algorithm, a software has been developed with Borland Delphi. The program encrypts the plaintext and decrypts the ciphertext specified by the user using the values of the keys. The user can select any plaintext file saved before or write any plaintext himself/herself for encryption online. In the software, the value of *n* can easily be changed and ciphertext is also changed according to this value. Decryption is carried out by using the saved key and ciphertext. This software can be used for secure communications in Internet such as secure e-mail sending and receiving, and secure chat. Besides, it can also be securely used to store the data. The screenshot of the software is shown in Figure 5.



n Value 8

Plaintext File

```
Oğuzhan TAŞ, Bilal ALATAŞ, Erhan AKIN
{oguzhantas, balatas, eakin}@firat.edu.tr
Department of Computer Engineering
Faculty of Engineering
Firat University, 23119, ELAZIĞ
```

Ciphertext File

```
1011111011111111001011010010011110010101111011
101111101001110101100001000000101101101010011
1111001011011101101110000000100001111000001001
```

Key File

```
386341362186875224452374276217357627311342643
```

Figure 5. Screenshot of the developed software for the unsystematic stream cipher

4. CONCLUSION

In this study, a new stream cipher technique has been developed. The technique is similar to technique of One Time Pad, proposed as perfect cipher in the literature, but the new technique is more advantageous. Different ciphertexts are obtained from the same key in the new technique while the same ciphertext is obtained from the same key in the technique of One Time Pad. Besides, in the new technique, the length of the ciphertext will always be longer than the plaintext. Random generation of the added numbers makes the breaking of the process of the encryption quite hard. Changing the bits of the character according to the added numbers increase the security of the process of encryption still more.

REFERENCES

1. B. Schneier, Applied Cryptography 2nd Edition, John Willey & Sons Inc, New York, 1996.
2. A. Menezes, Van Oorschot O., Vanstone S., Handbook of Applied Cryptography, CRC Press, 1997.
3. W. Stallings, Network Security Essentials Applications and Standards, Prentice Hall, New Jersey, 2000.
4. D.R. Stinson, Cryptography Theory and Practice, CRC Press, 1995.
5. D. Wagner, L.Simpson, E.Dawson, J.Kelsey, W.Millan, B. Schneier, Cryptanalysis of ORYX, Fifth Annual Workshop on Selected Areas in Cryptography, Springer Verlag, 1998.
6. ANSI X3.106, “American National Standard for Information Systems – Data Encryption Standard- Modes of Operation”, American National Standard Institute, 1983.
7. “The RC5 Encryption Algorithm”, B. Prencel, Fast Software Encryption, Second International Workshop (LNCS 1008) 86-96 Springer-Verlag, 1995.
8. “SAFER K-64: One year later”, B.Preneel, editor, Fast Software Encryption, Second International Workshop (LNCS 1008), 212–241, Springer-Verlag, 1995.
9. B. Schneier, “Description of a new variable-length key, 64-bit block cipher (Blowfish)”, R. Anderson, editor, Fast Software Encryption, Cambridge Security Work-shop (LNCS 809), 191–204, Springer-Verlag, 1994.
10. S. Miyaguchi, “The FEAL cipher family”, Advances in Cryptology–CRYPTO ’90 (LNCS 537), 627–638, 1991.
11. P. Rogaway, D. Coppersmith, “A Sofware Optimized Encryption Algorithm”, Journal of Cryptology, 273-287, 1998.
12. Golic J. Dj., Cryptanalysis of Alleged A5 Stream Cipher , Proceedings of Eurocrypt 97, Springer LNCS 1233, 239-255, 1997.
13. D. Kahn, “The Code Breakers- The Comprehensive History of Secret Communication from Ancient Times to the Internet” , Revised and Updated Edtion, Scribner, USA, 1996.
14. M.J.B. Robsaw, “Stream Ciphers”, RSA Laboratories Technical Report, 1995.
15. J. Kahanek, “Protecting Business Application with Encryption Symmetric and Asymmetric”, 2000.
16. D.J.Wheeler, R.M.Needham, “TEA, a Tiny Encryption Algorithm”, Cambridge University, England.
17. C.M.Adams, “Simple and Effective Key Scheduling fo Symmetric Ciphers”, Workshop on selected Areas in Cryptography Workshop Record, Kingston, Ontario, pp. 129-133, 5-6 May 1994.
18. C.M. Adams “Symmetric Cryptographic System for data encryption”, U.S Patent 5,511,123, 1996.
19. Rivest, Shamir, ve Adleman. “A method for obtaining digital signatures and public-key cryptosystems”. Comm. ACM, 120-126, 1978.
20. MasterCard Inc. SET Secure Electronic Transaction Specification, Book 1: Business Description, MasterCard Inc., May 1997.
21. A. O. Freier, P. Karlton, and P. C. Kocher. The SSL Protocol Version 3, Netscape Communications Corp., available from http://home.netscape.com/eng/ssl3, 1996.
22. C.P.Pfleeger, "Security in Computing" Second Edition, Prentice Hall, 1997.