AN INTEGRATED BLOCK AND STREAM CIPHER APPROACH FOR KEY ENHANCEMENT

[1]MANIKANDAN.G, [2]MANIKANDAN.R,[3] RAJENDIRAN.P, [4]KRISHNAN.G, [5]SUNDARGANESH.G

[1]Assistant Professor, School of Computing, SASTRA University, Thanjavur-613401, Tamil Nadu, India.
[2]Assistant Professor, School of Computing, SASTRA University, Thanjavur-613401, Tamil Nadu, India.
[3]Assistant Professor, School of Computing, SASTRA University, Thanjavur-613401, Tamil Nadu, India.
[4]Student, School of Computing, SASTRA University, Thanjavur-613401, Tamil Nadu, India.
[5]Student, School of Computing, SASTRA University, Thanjavur-613401, Tamil Nadu, India.

ABSTRACT

In the world of digital data transaction, in order to transact the data in a more secured manner the need for a cryptographic algorithm cannot be compensated by others. There are huge numbers of cryptographic algorithms which makes the system resistant from the attacks of intruders and eavesdroppers.RivestCipher4 and Blow fish are the cryptographic algorithms which are very well known for their performance, simplicity, strong key generation. In this paper, we propose a method of combining block and stream cipher for increasing the key strength so that it will be very hard for the intruder to break the key and intruder will have no idea about the key formation from the combination of block and stream cipher. So it will lead to increased key complexity which obviously results the intruder nothing else than confusion and frustration.

Keywords: *Blowfish, Cryptography, Encryption, Key strength, Key complexity, Rc4 algorithm.*

1. INTRODUCTION

Cryptography is a well known and widely used technique that manipulate information in order to crypt their existence. More specifically, cryptography protects information by transforming it into an unreadable format [1]. The original text is transformed into a scramble equivalent text called cipher text and this process is called as "Encryption". This is achieved via an Encryption Algorithm. Only those who possess a secret key can decrypt the cipher text into plaintext. Simply it scrambles a message so it cannot be understood.

Cryptography deals with protecting information by encoding or transformation of data [1].There are two types of cryptographic schemes available on the basis of key.

- *Symmetric key Cryptography*: This is the cryptographic scheme which uses a common key for enciphering and deciphering the message.
- *Asymmetric or Public Key Cryptography*: This type of cryptographic scheme uses two keys for encryption and decryption called Public key and Private Keys.

We can also classify symmetric key cryptography into two types on the basis of their operations as

- *Stream Ciphers*: It is a symmetric key cipher where stream of plaintext are mixed with a random cipher bit stream (key stream), typically by any logical operation. In the case of stream cipher one byte is usually encrypted at a particular time.
- *Block Ciphers*: It is also a symmetric key cipher which operates over a fixed-length group of bits. It usually takes particular bit block of plaintext as input, and produces a corresponding n-bit output block of cipher text.

2. EXSISTING SYSTEM

2.1 Rc4 Algorithm

RC4 is a synchronous stream cipher designed to satisfy both security and efficiency for lightweight algorithms, dedicated to hardware environments where the usage of resources are restricted. In general, Stream ciphers like RC4 have fragileness on the key size. If the key size is too short then the attacker can easily obtain the key by using one of the existing key recovery algorithms, so a need for re-keying becomes vital [10]. Generally, re-keying is done by using the internal state in each packet to reinitialize the large internal state. The newly initialized internal state is state dependant; therefore more variation and randomness can be achieved for the internal state. The value of internal state must be unique, and must not be used twice or more although the messages to be encrypted are different. Stream cipher is the important class of encryption and they encrypt each digit of plain text one at a time using a simple enciphering transformation. Rc4 is most widely used stream cipher nowadays due to its simplicity and high efficiency .Rc4 usually has varying key size and key size of stream cipher based on a 256 byte internal state and two one byte indexes I and J. The operation of Rc4 comprises of two phases namely key scheduling algorithm and pseudo random generation [2].

The key-scheduling algorithm consists of following steps. In order to generate a key we should start the permutation in the Array S. Here the term Key length can be defined as the number of bytes in the key which usually is of a range 1 ≤ key length ≤ 256.

```
for (i=0;i<=255;i++)
S[i] = I;
j = 0;
for (i=0;i <=255;i++)
{
j = (S[i] + i + key[i mod key length]) mod
256;
swap (S[i] , S[j]);
}
```

The Pseudo random generation algorithm changes the state and particular outputs a byte over the key stream. In each iteration, the PRGA increments *i*, adds the value of S pointed by *i* to *j*. [3]

```
i = 0;j = 0;
while
{
   i = (i + 1) mod 256;
   j = (j + S[i]) mod 256;
   swap( S[i] , S[j] );
   K = S[(S[i] + S[j]) mod 256];
}
```

2.2 Blowfish Algorithm

Blowfish, a symmetric block cipher and a Feistel network which follows simple Enciphering and Deciphering functions of 16 times each. The strength of the Blowfish algorithm relies on its sub-key generation and its basic confusion and diffusion based design. [2].Blowfish cipher uses 18 each of 32-bit Permutation arrays precisely known as P-Boxes and 4 Substitution boxes referred as S-Box each of 32 bit size and having 256 entries each. It uses a Feistel cipher which is a general method of transforming a function into an another function by using the concept of permutation, diffusion, confusion [3]. The working of blowfish cipher can be illustrated as follows,

It splits the 64 bit block into two equal blocks having 32 bit size each. Left block is XORed with first Sub array P1 and thus obtained result is fed in to a function called F-function. Inside the F-function substitution operations are carried out which in turn converts 32 bit blocks in to another 32 bit blocks. Thus resulted 32bit entries are XORed with the Right half and the result obtained is swapped as the left half for the next round. The Fiestal Structure of Blowfish Algorithm with 16 rounds of encryption is shown in the following figure1.

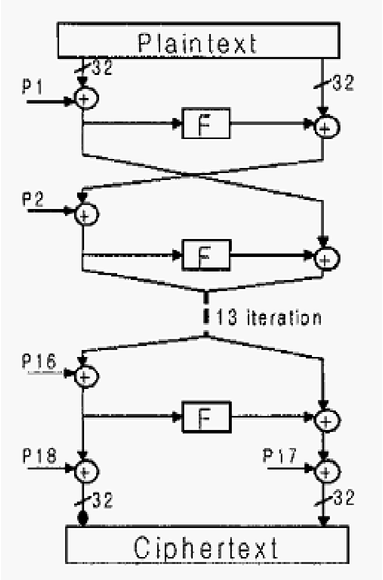


Figure 1: Fiestal structure of Blowfish Cipher

The transformation operations that actually happen inside an F-function are XOR Operation, ADD Operation and few table look up operations. These operations are carried out between four S-Boxes and as a result of all manipulations finally 32 bit entries are transformed into another 32 bit entry.

3. PROPOSED SYSTEM

In this paper, we propose a new system for integrating the Block and the Stream cipher for generating a fresh key using blowfish algorithm from the actual key provided by the user. The original key is supplied as a plaintext to the blowfish algorithm and it produces a cipher text will be taken as fresh and it is supplied as the key for RC4 algorithm. Thus obtained is hard to crack because it involves addition of the complexity from the block cipher to the key for a stream cipher For our publication purpose, we proposed the Blowfish and RC4 integration for the generation of strong key. The following block diagram illustrates our approach of integrating block and stream cipher using Blowfish algorithm and Rivest Cipher 4 algorithm.

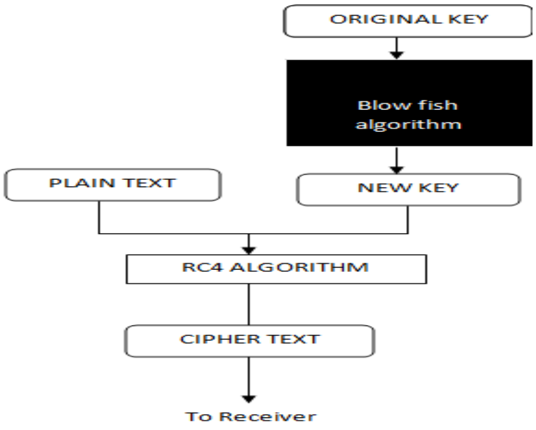


Figure 2: Block diagram of Encryption Phase

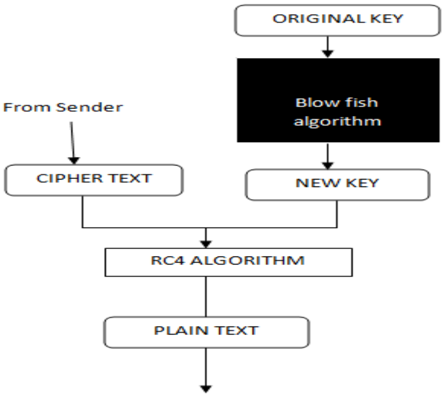


Figure 3: Block diagram of Decryption Phase

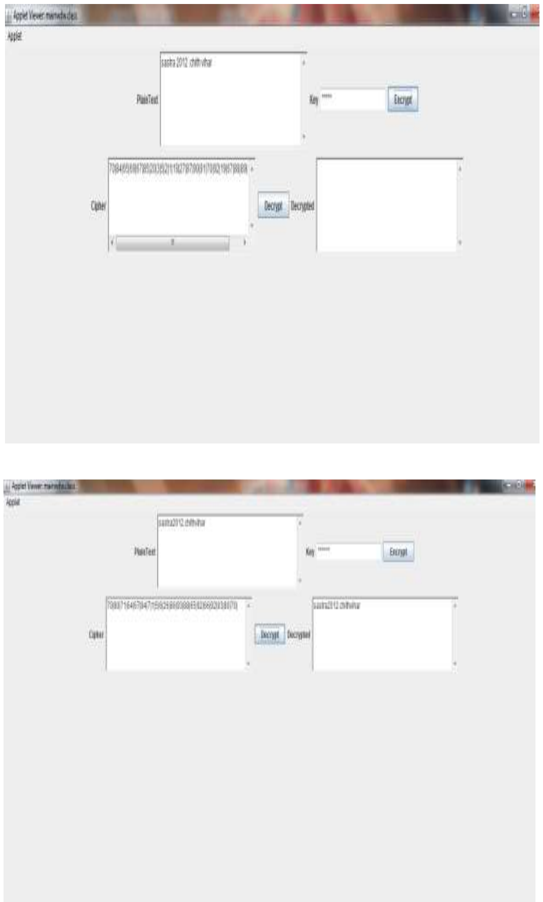
The significance of the proposed systems is as follows:

- Rc4 is enhanced in terms of key strength or its complexity. The newly formed key which is generated will not be known even to sender and receiver.
- In general, Key strength which is one of the salient feature of block cipher which is missing in all stream ciphers. This scenario will come to an end by the usage of blowfish which is a block cipher and it will increase the key strength that is ever provided by other stream ciphers.
- Since blow fish execution is independent of the original RC4 algorithm, there is no chance for affecting the data security of RC4 algorithm.

- We propose this blowfish in order to increase the key strength so that it will be suitable in the mere future to hold any algorithm like RC4 which lags in their key strength.

6. SIMULATION AND RESULTS

For the purpose of simulating the blowfish and RC4 algorithm we used Java which is known for its platform independency and better GUI features. We developed, tested and executed using JDK 1.6 in core 2 duo processor. We adopted JCreator1.6 for IDE purposes. The following figures illustrates about the encryption and decryption phase of Rc4 algorithms. In the Key textfield, the entered key is none other than the ciphertext derived from the Blowfish algorithm.



PlainText: sastra 2012 chithvihar
Key: *****
Cipher: [illegible]

PlainText: sastra2012 chithvihar
Key: ******
Cipher: [illegible]
Decrypted: sastra2012 chithvihar

Figure 4:Rc4 Encryption & Decryption

7. FUTURE ENHANCEMENTS

The integration of the blowfish and Rc4 can be easily modified to accept any encryption algorithm which is framed in future. Just by adding or removing another module in the main function and also by increasing or decreasing the hash limit, any number of iteration can be included or reduced in the particular algorithm or other in particular they are used to increase the key length especially in the case of RC4[9]. Though the system is designed for stream cipher but the modules can be used in block cipher also in which key length is already in high complexity. By adding a new authentication between the sender and receiver sockets, the system can also be improved to work as secure WAP and WLAN.

8. CONCLUSION

Key strength of RC4 has the main concern of security weakness , so that they can be easily breached, the tool that we developed is used to increase the complexity of the key .The intruder will have no idea about the existence of the particular block cipher in our case we have blow fish which is the strongest block cipher and before his realization of existence of block cipher for key generation particular data will be transferred to the receiver. Thus the integration of block and stream cipher that is being implemented will increase the key complexity without affecting the performance up to a maximum level that ever offered by the existing stream ciphers.

REFERENCES:

[1] B. Schneier, Applied Cryptography, 2nd edition, New York:Wiley, 1996.

[2] G. Gong, K. C. Gupta, M. Hell, and Y. Nawaz, “Towards a general RC4-like keystream generator,” in *SKLOIS Conf. Inf. Security and Cryptol., CISC 2005 LNCS 3822*. New York: Springer-Verlag, 2005,pp. 162–174.

[3] S. Paul and B. Preneel, “On the (in)security of stream ciphers based on arrays and

modular addition," in *Adv. Cryptol.—Asiacrypt 2006*, *LNCS4284*. New York: Springer-Verlag, 2006, pp. 69–83.

[4] S. Fluhrer, I. Mantin, and A. Shamir, "Weaknesses in the key scheduling algorithm of RC4," in *Sel. Areas in Cryptogr.*, *SAC 2001*, *LNCS2259*. New York: Springer-Verlag, 2001, pp. 1–24.

[5] S. Fluhrer and D. McGrew, "Statistical analysis of the alleged RC4 keystream generator," in *Fast Software Encrypt.*, *FSE 2000 LNCS1978*. New York: Springer-Verlag, 2000, pp. 19–30.

[6] S. Paul and B. Preneel, "A new weakness in the RC4 keystream generator,"in *Fast Software Encrypt.*, *FSE 2004, LNCS 3017*. New York: Springer-Verlag, 2004, pp. 245–259.IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 53, NO. 9, SEPTEMBER 2007 3255

[7] I. Mantin, "Predicting and distinguishing attacks on RC4 keystream generator," in *Adv. Cryptol. — Eurocrypt 2005*, *LNCS 3494*. New York: Springer-Verlag, 2005, pp. 491–506.

[8] I. Mantin, "A practical attack on the fixed RC4 in the wep mode," in *Adv. Cryptol. — Asiacrypt 2005*, *LNCS 3788*. New York: Springer-Verlag, 2005, pp. 395–411.

[9] R. J. Jenkins, Jr., *ISAAC Fast Software Encrypt.*, *FSE 1996*, *LNCS1039*. New York: Springer-Verlag, 1996, pp. 41–49.

[10] G.Manikandan,R.Manikandan,G.SundarGanesh,A New Approach for Generating Strong Key in RC4 Algorithm.,JATIT Vol.24,No.2,2011