The implementation of RC4\(^+\) and Variably Modified Permutation Composition algorithms in the three-pass protocol scheme for data security

M A Budiman, D Rachmawati, and R A Badegil

Departemen Ilmu Komputer, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Sumatera Utara, Jl. Alumni No. 9-A, Kampus USU, Medan 20155, Indonesia

Email: mandrib@usu.ac.id, dian.rachmawati@usu.ac.id, rahmaasdarina@gmail.com

Abstract. The science and art that is used to take care of message security by encoding messages into obfuscated forms is called cryptography. In cryptography, there are two types of algorithms, which are symmetric and asymmetric algorithms. The RC4\(^+\) algorithm and the Variably Modified Permutation Composition (VMPC) algorithm are examples of symmetric algorithms in the form of stream ciphers where the process of encryption and decryption is done by using the same key. To achieve the security objectives, these two algorithms are then used and applied into the three-pass protocol scheme. The three-pass protocol is a cryptographic scheme for sending and receiving messages that allows one party to securely send messages to a second party without the need to exchange the keys. As a result, this research succeeded in securing text file with (*.docx) dan (*.pdf) and can restore the data back as before by combining the RC4\(^+\) algorithm and the Variably Modified Permutation Composition (VMPC) algorithm in the three-pass protocol scheme. The theoretical complexity of these two algorithms is \(\theta(n)\) both for encryption and decryption. The actual running time is directly proportional to the length of the character of a file.

1. Introduction

One example of communication at this time is to exchange messages to others in digital manner. However, not all of the things communicated are in public domain; there are times when the message being communicated is confidential. Information has become such a valuable asset so it needs more specific efforts to keep its security. This has certainly attracted the intruders to find and use any existing weaknesses of the predefined system information configuration [1].

Encryption is the process of transforming information into a form that no longer understandable by any human beings. Encryption can be a workable solution to prevent eavesdropping of data that we send or receive. When an "eavesdropper" steals an encrypted data, it can only see meaningless symbols because the encrypted data does not reflect the actual data content [1].

It is only when the sender and recipient have agreed to use a certain encryption method and encryption key then encryption can be done. The encryption key must be maintained so that no outsiders know it. The problem that then arises is how to tell the recipient about the key so he or she can decrypt the message. The key agreement that will be used can be achieved through more secure channels of communication, for example by meeting face-to-face. But what if this safer
communication path is not possible? The next question is how to send the key to keep it confidential. If the key can be known by someone else then that person can unpack the message we will send [1].

The Three-Pass Protocol in messaging is a communication mode that allows one party to securely send messages to a second party without the need to exchange or distribute encryption keys. The Three-Pass Protocol was first developed by Adi Shamir in 1980. It is called the Three-Pass Protocol because the sender and recipient exchange three times to encrypt the message.

RC4 stream cipher was proposed by Maitra et al. in Indocrypt 2008. This stream cipher is designed to address all the flaws reported on alleged RC4 stream ciphers. In RC4 design specifications, the authors use 8-bit design parameters called pad [2]. The RC4 algorithm is one type of RC4 algorithm invented by Ronald Rivest in 1987. The RC4 algorithm is one of the key symmetric algorithms in the form of a stream cipher that encrypts and decrypts in one byte using the same key.

The Three-Pass Protocol scheme is not an algorithm, but merely a digital communication mode. Therefore, the Three-Pass Protocol requires an algorithm to support the process of running the digital communication procedure. Based on this, the authors intend to combine the RC4’ algorithm and the Variably Modified Permutation Composition (VMPC) algorithm in the Three-Pass Protocol scheme for data security so that users do not need to exchange keys.

2. Methods
In this study, sender uses RC4’ algorithm and recipient uses Variably Modified Permutation Composition (VMPC) algorithm. The sender and recipient will send the message using the Three-Pass Protocol scheme.

2.1. RC4’ Algorithm
The RC4’ algorithm is one type of RC4 algorithm invented by Ronald Rivest in 1987. The RC4 algorithm is one of the key symmetric algorithms in the form of a stream cipher that encrypts and decrypts in one byte using the same key. The RC4 Cipher algorithm uses variables whose key length is from 1 to 256 bits which is used to initialize a 256 byte table where this table will be used for the ciphertext formulation process.

The RC4’ algorithm uses a structure like RC4 by adding some operations to strengthen the cipher. This structure tries to exploit good points on RC4 then provides some additional features for better security limits [4]. The RC4’ algorithm has two main parts, the Key Scheduling Algorithm (KSA) and the Pseudo-Random Generation Algorithm (PRGA).

2.1.1. Key Scheduling Algorithm (KSA)
The structure of the RC4’ algorithm is the same as the RC4 Cipher algorithm. Both algorithms have a Key Scheduling Algorithm (KSA). The KSA of the RC4’ algorithm as follows.

\[
\text{for } i \text{ from } 0 \text{ to } 255 \\
S[i] := i
\]
endfor

j := 0
for i from 0 to 255
    j := (j + S[i] + key[i mod keylength]) mod 256
    swap values of S[i] and S[j]
endfor

At this stage i and j are the initial variables that are 0 and S are all possible 256 permutations.

2.1.2. Pseudo-Random Generation Algorithm (PRGA)

Systematic PRGA algorithm RC4 is written as follows:

while random:
    i := i + 1
    a := S[i]
    j := j + a

    Swap S[i] and S[j]  
    (b := S[j]; S[i] := b; S[j] := a)

    c := S[i] =< 5 ⊕ j>>3] + S[j] =< 5 ⊕ i>>3

    output (S[a+b] + S[c ⊕ 0xAA]) ⊕ S[j+b]
endwhile

In the Pseudo Random Generation Algorithm (PRGA) phase of the RC4 algorithm, there are i and j as 8-bit array indexes, S is all possible 256 permutations, ⊕ is exclusive OR, << is left shift, and >> is the right shift.

2.2. Variably Modified Permutation Composition (VMPC) Algorithm

Variably Modified Permutation Composition (VMPC) is one variation of the RC4 algorithm in the form of stream cipher. The inventor of this cryptographic system was Bartosz Zoltak published in 2004. The VMPC Key Scheduling Algorithm changes the cryptographic key and vector initialization (optional) to the permutation of 256 P elements and initializes the variable s. The VMPC cryptographic system works by generating the keystream with a keystream generator and then performing the encryption algorithm with the key and plaintext. This algorithm has two main parts, namely Key Scheduling Algorithm (KSA) and Pseudo-Random Generation Algorithm (PRGA).

VMPC Stream Cipher (PRGA) is executed after VMPC Key Scheduling Algorithm is executed first. Note that initialization vector is optional. However, with the initialization vector, the resulting permutation gets more random. Similar to a typical stream cipher scheme, the VMPC Stream Cipher generates a keystream with a keystream generator and an encryption algorithm with the key and plaintext is performed [5]. The resulting values must be XOR with plaintext to get ciphertext [6]. VMPC algorithm has two main parts, namely Key Scheduling Algorithm (KSA) and Pseudo-Random Generation Algorithm (PRGA).

2.2.1. Key Scheduling Algorithm (KSA)

In the Key Scheduling Algorithm (KSA) there are two differences with the KSA RC4 algorithm that occurs three times on the array 256, and there is an additional option for initialization of the vector [7]. Here is the Key Scheduling Algorithm (KSA) of the Variably Modified Permutation Composition (VMPC) algorithm.

1. s = 0
2. for n from 0 to 255: P[n] = n
3. for m from 0 to 767: repeat steps 4 to 6:
   4. n = m modulo 256
5. \( s = P[(s + P[n] + \text{Key}[m \text{ modulo keylength}]) \text{ modulo } 256] \)
6. \( \text{Temp} = P[n] \)
   \( P[n] = P[s] \)
   \( P[s] = \text{Temp} \)

There is \( P \) as a 256-byte permutation storage array, \( s \) which is an 8-bit variable, as well as \( n \) and \( m \) which is an 8-bit variable.

2.2.2. Pseudo-Random Generation Algorithm (PRGA)

Here's the scheme of the Pseudo-Random Generation Algorithm (PRGA) on the VMPC algorithm:

1. \( n = 0 \)
2. Repeat the steps 3-6 along \( L \):
   3. \( s = P[(s + P[n]) \text{ modulo } 256] \)
   4. \( \text{Keystream} = P[(P[P[s]] + 1) \text{ modulo } 256] \)
   5. \( \text{Temp} = P[n] \)
      \( P[n] = P[s] \)
      \( P[s] = \text{Temp} \)
   6. \( n = (n + 1) \text{ modulo } 256 \)

At the PRGA stage there is \( P \) as a 256-byte permutation storage table initialized by VMPC KSA, \( s \) is an 8-bit variable initialized by VMPC KSA, \( n \) and \( m \) which is an 8-bit variable, and \( L \) is the desired plaintext length in bytes.

2.3. Three-Pass Protocol

The Three-Pass Protocol in messaging is a framework that allows one party to securely send messages to a second party without the need to exchange or distribute encryption keys. The Three-Pass Protocol was first developed by Adi Shamir in about 1980. It is called the Three-Pass Protocol because the sender and recipient exchange three stages to encrypt the message. The basic concept of the Three-Pass Protocol is that each party has a private encryption key and a personal decryption key. Both parties use their own key to encrypt and to decrypt the message so that the sent message can reach the recipient safely without fear of key leakage.

Information:

- : Sender encryption key using RC4
- : Receiver encryption using VMPC
- : Sender decryption key using RC4
- : Receiver decryption key using VMPC
Figure 1. The Conceptual Diagram of Three-Pass Protocol

In Figure 1 it is shown that the Three-Pass Protocol has several stages to be able to deliver the message from the sender to the recipient. Here are the steps:

a. Asdarina (message sender) has an RC4+ (K_A) key. Asdarina encrypts the message with her own key and sends an encrypted message (C1) to Dareen (the recipient of the message).

b. Dareen (the recipient of the message) has a VMPC (K_B) key. Dareen encrypts the C1 message with his private key and sends the encrypted message again (C2) to Asdarina.

c. Asdarina who has received C2 decrypts the message using her own private key RC4+ key and sends the decrypted message (C3) back to Dareen.

d. Dareen who has received C3 also decrypts the message with his private key, so now he obtains the original message (P) sent by Asdarina.

3. Result and Discussion

3.1. The Encryption Result of RC4+ Algorithm and Variably Modified Permutation Composition (VMPC) Algorithm

The author will perform encryption testing with the following example:

• The plaintext to be encrypted is "darin".
• The RC4+ key to use is "ari".
• The VMPC key to use is "comm".

The results of encryption done by the sender using RC4+ algorithm is ciphertext 1. The first ciphertext (ciphertext 1) generated by the RC4+ algorithm is "Ó 5HE". And then, this ciphertext 1 will be sent to receiver. After the message is received, the message will be re-encrypted using the Variably Modified Permutation Composition (VMPC) algorithm. The result of the encryption done by the recipient using the VMPC algorithm is ciphertext 2. The second ciphertext (ciphertext 2) generated by the VMPC algorithm is "6A1E".

3.2. The Decryption Result of RC4+ Algorithm and Variably Modified Permutation Composition (VMPC) Algorithm

The author will perform decryption testing with the following example:

• The ciphertext to be decrypted is "6A1E".
• The RC4+ key to use is "ari".
• The VMPC key to use is "comm".
Next, ciphertext 2 will be sent back to the sender to be decrypted into ciphertext 3. The result of decryption done by sender using RC4 algorithm is ciphertext 3. The third ciphertext (ciphertext 3) generated by the RC4 algorithm is "dľ3Ģq". And then, this ciphertext 3 will be sent to receiver. After the message is received, the message will be decrypted again using the Variably Modified Permutation Composition (VMPC) algorithm. The decryption test results by the receiver using the Variably Modified Permutation Composition (VMPC) algorithm is plaintext. The original plaintext generated by the algorithm is "darin".

3.3. Process Time Encryption and Decryption

The average test result of encryption process using RC4 algorithm with the number of characters 50, 500, and 2000 characters are 16.0417 ms, 17.5223 ms, and 33.491 ms. The average test result of encryption process using Variably Modified Permutation Composition (VMPC) algorithm with the number of characters 50, 500, and 2000 characters are 2.7997 ms, 11.3174 ms, and 23.6566 ms. The results can be illustrated with the graphs shown in Figure 2 and Figure 3.

![Character Length Chart against RC4 Algorithm Encryption Process Time](image1)

**Figure 2. Character Length Chart against RC4 Algorithm Encryption Process Time**

![Character Length Chart against VMPC Algorithm Encryption Process Time](image2)

**Figure 3. Character Length Chart against VMPC Algorithm Encryption Process Time**

The average test results of decryption process using RC4 algorithm with the number of characters 50, 500, and 2000 characters are 1.7791 ms, 8.536 ms, and 19.5899 ms. And the average test result of decryption process using Variably Modified Permutation Composition (VMPC) algorithm with the
The number of characters 50, 500, and 2000 characters is 0.5751 ms, 4,3364 ms, and 11,3222 ms. The results can be illustrated with the graphs shown in Figure 4 and Figure 5.

![Character Length Chart against Time of Decryption Process of RC4 Algorithm](image1)

**Figure 4. Character Length Chart of Process Time Decryption RC4 Algorithm**

![Character Length Chart against Time of Decryption Process of VMPC Algorithm](image2)

**Figure 5. Character Length Chart of Process Time Decryption VMPC Algorithm**

4. Conclusions

The conclusions of this work are as follows. Firstly, by using Three-Pass Protocol encryption and decryption process using RC4 algorithm and Variably Modified Permutation Composition (VMPC) algorithm we can return the message to its original form. Secondly, the complexity of the RC4 algorithm comprising the process of the Key Scheduling Algorithm (KSA) and the Pseudo-Random Generation Algorithm (PRGA) yields a value of \( \theta(n) \). Thirdly, the complexity of the Variably Modified Permutation Composition (VMPC) algorithm, which consists of the process of Key Scheduling Algorithm (KSA) and Pseudo-Random Generation Algorithm (PRGA) also yields a value of \( \theta(n) \). Fourthly, the time of encryption process using RC4 algorithm with length 50, 500, and 2000 characters are 16,0417 ms, 17,5223 ms, and 33,4491 ms. The time of encryption process using VMPC algorithm with length 50, 500, and 2000 characters are 2,7997 ms, 11,3174 ms, and 23,6566 ms. The test results are shown with linearly shaped graph which means that the length of the plaintext character is directly proportional to the process time. The longer the plaintext characters, the more time the process takes the two algorithms.
5. References

[1] Anonim 2003 *Memahami Model Enkripsi dan Security Data* Andi Semarang Wahana Komputer Yogyakarta

[2] Banik Subhadeep and Sonu Jha 2015 *Some Security Results of The RC4+ Stream Cipher Security and Communication Network* 8(18) ISSN : 1939-0114 pp 4061-4072

[3] Chayanie, Noer Inda 2017 Implementasi Algoritma Zig-zag Cipher dan Algoritma RC4+ Cipher dalam Skema Super Enkripsi untuk Pengamanan File Skripsi Universitas Sumatera Utara

[4] Paul Goutam, Subhamoy Maitra 2012 *RC4 Stream Cipher and Its Variants* London: CRC Press Taylor & Francis Group

[5] Bastari A T A 2011 Analisis dan Implementasi VMPC (Variably Modified Permutation Composition) Stream Cipher sebagai Mac (Message Authentication Code), Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung

[6] Zoltak, B 2004 *VMPC One-Way Function and Stream Cipher* : Delhi India

[7] Karahan M 2015 *New Attacks on RC4A and VMPC* Thesis of The Graduate School of Engineering and Science Bilkent University