

Enhanced OneTime Pad Cipher with MoreArithmetic and Logical Operations with Flexible Key Generation Algorithm

S.G.Srikantaswamy¹ and Dr.H.D.Phaneendra²

¹Research scholar, CSE, National Institute of Engineering, Mysore, Karnataka, India
sg_srikantaswamy@yahoo.com

²Professor and Research Guide, CSE, NIE, Mysore, Karnataka, India
hdphanee@yahoo.com

ABSTRACT

The process of exchanging information is called Communication. The basic Communication system involves transmitter, receiver and the channel. The data transmitted by the sender reaches receiver through the channel. The unauthorized parties (cracker,hacker, eavesdropper, or attacker) should not be able to access the information at the channel. Therefore transmitting data securely from the sender to the receiver is a very important aspect. A cryptographic system is unconditionally secure if the cipher text produced by the system does not contain enough information to determine uniquely the corresponding plaintext, no matter how much cipher text is available. A cryptographic system is said to be computationally secure if the cost of breaking the cipher exceeds the value of the encrypted information and the time required to break the cipher exceeds the useful lifetime of the content. One time pad system can be called as unconditionally secure algorithm, if the keys (pad) used are truly random in nature. In this paper, we are demonstrating that one-time pad can be used as an efficient encryption scheme by involving arithmetic and logical operations. Here we proposed a new key generation technique, to generate a key of any length just by providing a seed value, to encrypt the message. The problem generating key value has been solved by the use of key generation algorithm.

KEYWORDS

Cipher, Plaintext, Cipher text,Confusion,Diffusion,Encryption,Decryption,Eavesdropper, Cryptanalysis.

1. INTRODUCTION

In one-time pad encryption scheme, random key that is equal in length to the plaintext message to be encrypted, with no repetition is used. The plaintext character is exclusive-or bitwise with the key, to produce cipher text output.

$C=P\oplus K$, Where P=Plaintext, K=Key, and C=Cipher text

Thus the ciphertext is generated by performing the bitwise Exclusive-or of the plaintext with the key. Because of the properties of Exclusive-or, decryption simply involves the same bitwise Exclusive-or operation. The main risk of the scheme lies with the pad used. Also the use of only exclusive-or operation to generate ciphertext makes the system simple and involves less time for computations. More complex the operations used more the time required to decrypt and more secure the algorithm. The one time pad system has been modified by using the concepts of 9's complement operation. The 9's complement version

of the plaintext value is processed with the key value to produce cipher text [1]. Another variation of the one time pad involves the process of taking the 2's complement of the binary number obtained by adding plaintext with the key and subtracting 26 if the complement is greater than 26, else retain the complemented sum.[2]. One time pad system can also be used for providing authentication services [3]. Modified version of one time pad security system with random key generation technique is suggested in [4]. One time pad system concept can also be used to secure short messaging service [5]. A Practical implementation of one time pad system from hardware to software has been discussed in [6]. The ciphertext produced by one time pad contains no information about the plaintext, there is simply no way to break the code[7]. Assuming an eavesdropper can not get access to the one time used to encrypt the message, this scheme is perfectly secure, but the given cipher text should be equal to the plaintext message in length [8]. The main drawback of the one time pad encryption scheme is that the key is required to be as long as the message. This means that it is necessary to store the long keys securely and processed and it is a tedious task and highly problematic [9]. Consider two messages m and m' which are encrypted using the same key K . A person who obtains $C = m \oplus K$ and $C' = m' \oplus K$ can compute

$$C \oplus C' = (m \oplus K) \oplus (m' \oplus K)$$

$C \oplus C' = m \oplus m'$. Thus it gives some hint about the plaintext message [9].

The performance of the algorithm also plays a major role. It depends on the method of key generation, storage and usage. Response time plays a major role in designing the throughput of the algorithm. Response time is defined as the time interval between a user's request and the system's response [10]. The combined effect of encryption and compression operations can be used to achieve data security and to minimize resource requirements [11]. The features of symmetric and asymmetric algorithms can be combined to achieve appreciable level of security [12]. In this paper, we have devised a method for modifying the one time pad scheme. In one time pad it is very difficult to generate the keys which are truly random. It is very difficult to generate, store and process the keys which are as long as the messages to be encrypted. It is very tedious to generate and store long key values as it requires large amount of memory and system resources. Hence it is highly essential to devise an algorithm which works in parallel to one time pad and supplies the key required for encryption. In this paper we have devised a method for generating keys separately at the sending and receiving ends. The algorithm requires only the seed value. If the entity A wants to communicate with B, then A can select a random seed value and can generate the keys. The generated key values can be used to encrypt the plaintext information. Now A can send the seed value to entity B, by encrypting the seed value using a secret key value which had already been shared by entity A and B in some previous communications. Now receiver B can decrypt the encrypted value of the seed value and can generate the keys required for decrypting the encrypted message. In this method, the key values are shifted left before using it with the plaintext to make it more random. This shift operation is employed to introduce confusion properties. The left shifted values of the key values are then used with plaintext characters to produce ciphertext output. The proposed system is designed in such a way as to exhibit both confusion and diffusion properties to thwart the efforts of cryptanalysis. Thus the proposed system is computationally secure. The main theme behind this paper is to generate keys separately by a key generation algorithm and to use more arithmetic and logical operations in processing plaintext and key values to produce final ciphertext, to hide the statistical relationship between the ciphertext and the plaintext. The arrangement of the paper is as shown below. Section 1 gives Introduction; Section 2 describes

encryption and decryption details. Section 3 contains encryption and decryption algorithms. Section 4 describes key generation algorithm.

Section 5 involves results and discussions. Section 6 gives out the features of the proposed system. Section 7 gives conclusions drawn from the above analysis. Section 8 contains references.

2. ENCRYPTION AND DECRYPTION DETAILS

Consider the plaintext message "TRY". Replace the characters in the plaintext message by their ASCII (Decimal) Equivalent values for the purpose of calculations and computations. Let the plaintext characters be designated as P1, P2 and P3. (i.e. P1=T, P2=R, and P3=Y). Now using a suitable seed value, generate the key values K1, K2 and K3. The length of the plaintext characters and the key should be equal. Now the key values are left shifted by one position. The plaintext is Exclusive-or with the key values to produce the partial ciphertext. The above result is added with the doubled basic key values. ($2 \times K1$, $2 \times K2$ and $2 \times K3$ respectively). Now take the 10's complement of the above sum. Then subject the 10's complemented sum to modulo 127 arithmetic operations. The resultant remainders generated after modulo operations are treated as the resultant cipher text output.

3. ENCRYPTION AND DECRYPTION ALGORITHMS

3.1 Encryption Algorithm

- i) Start
- ii) Input the plaintext
- iii) Convert the plaintext characters into ASCII (Decimal) Equivalent values.
- iv) Generate the key values by providing the proper seed value to the key generation algorithm
- v) Left shift the key values by one position.
- vi) Exclusive-or the left shifted key values with the plaintext.
- vii) Multiply the key values by 2 and add the output of step 5 with the doubled Key values.
- viii) Take the 10's complement of the sum produced by step 6
- ix) Take modulo 127 of the result produced by step 7
- x) Store the remainders generated by step 8 as the resultant cipher text Characters
- xi) Encrypt the key seed value with secret key and transmit it with the resultant cipher text to the receiver.
- xii) Stop

3.2 Decryption Algorithm

- i) Start
- ii) Input the cipher text and encrypted key seed value
- iii) Decrypt the key seed value and generate the key values using key generation algorithm
- iv) Apply the above steps on the ciphertext in reverse direction
- v) Retrieve the plaintext message
- vi) Stop

4. KEY GENERATION ALGORITHM

- i) Let the seed value be K
- ii) Now let us assume that the key for encrypting first plaintext character of the plaintext be K_1 . Now key for encrypting first character be K_1 and let $K_1=K$
- iii) Now for generating second key say K_2
- iv) $K_2=2*K_1$
- v) The next key value is equal to the thrice the value of the Previous key value
- vi) Now $K_3=3*K_2$;
- vii) $K_4=4*K_3$; $K_5=5*K_4$ and so on
- viii) In general, the required Key Value is:
- ix) End

$K_n = n * K_{n-1}$

5. RESULTS AND DISCUSSIONS

1. Let us consider the Plaintext “**TRY**”.
2. Replace the plaintext character by ASCII decimal equivalent values ($P_1=T, P_2=R, P_3=Y$),
Then $T=84, R=82, Y=89$
3. Let the key seed value be i.e. $K=50$
4. Now Let $K_1=K=25$
 $K_2=2*K_1=2*25=50$
 $K_3=3*K_2=3*50=150$
5. Therefore $P_1=84, P_2=82, P_3=89$
6. $K_1=25, K_2=50, K_3=150$
7. Left shift K_1, K_2 and K_3 by one Position.
8. Let $K_{11}=K_1 \ll 1$
 $K_{22}=K_2 \ll 1$
 $K_{33}=K_3 \ll 1$, Then the values of K_{11}, K_{22} and K_{33} would be as shown below
 $K_{11}=50$
 $K_{22}=100$
 $K_{33}=300$
9. Now exclusive or P_1, P_2 and P_3 with K_{11}, K_{22} and K_{33} relatively
 $P_{11}=P_1 \oplus K_{11}$
 $P_{12}=P_2 \oplus K_{22}$
 $P_{33}=P_3 \oplus K_{33}$,
10. By Performing the XOR operations, we get P_{11}, P_{22} and P_{33} as detailed below.
 $P_{11}=102$
 $P_{22}=54$
 $P_{33}=373$
11. Now multiply K_1 by 2 and add with P_{11} to get P_{111} .
 $P_{111}=K_1 * 2 + P_{11}$
12. Multiply K_2 by 2 and add with P_{22} to get P_{222}
 $P_{222}=K_2 * 2 + P_{22}$
13. Multiply K_3 by 2 and add with P_{33} to get P_{333}
 $P_{333}=K_3 * 2 + P_{33}$

14. The Numerical values of P111 , P222 and P333 after performing the above calculations are as follows.
P111=152
P222=154
P333=673
15. Now ,Take the 10's complement of P111,P222 and P333.
Let P111=10's complement of p111,
P222=10's complement of p222
and P333=10's complement of p333.
16. Then ,We get P1111=848
P2222=846
P3333=327
17. Now by performing the modulus operations on p1111, p2222 and p33333 as shown below, we get
P1111%127=86
P2222%127 =84
P3333%127=73
18. Considering the remainders as the corresponding ciphertexts C1, C2 and C3, we get
C1=P1111%127=86
C2=P2222%127=84
C3=P3333%127=73
19. Now by, replacing the ASCII (Decimal) values by corresponding ASCII character, we get the final Ciphertext character string as detailed below.
20. The cipher text string is: V T I
21. The Decryption is performed exactly in the reverse direction as that of the encryption,to obtain the Plain text output.The Plaintext output obtained after the decrypting the Plaintext is **TRY**.

6. FEATURES OF THE PROPOSED SCHEME

- i) Variable Key Length
- ii) Effective Security
- iii) Flexible Key Generation Technique
- iv) Less Coding Complexity
- v) Inherent Confusion and Diffusion properties
- vi) Less Complexity in analysis

7. CONCLUSION

Security of information is a very important requirement in data communications. Designing an unconditionally secure algorithm is very difficult task. This is because with the use of high speed computers any cipher system can be cracked. Therefore it is very much desirable to design computationally secure algorithms. One time pad is unbreakable if keys are truly random. The main computational difficulty lies with the generation of one time pad and also memory requirement and processing speed also play a very crucial role. By providing a appropriate key seed value different key values can be generated. The proposed scheme can be further improved by using modular arithmetic operations. The algorithm can be employed to plaintext message of any length.

REFERENCES

- [1] SrinivasanNagaraj,KishoreBhamidipati, MRamachandra : Formal Method of Encryption Using 9's Complement –International Journal of Computer Applications (0975-8887),volume 8-No.5.october 2010
- [2] SharadPatil , Ajay Kumar:Effective Secure Encryption Scheme(One Time Pad) using Complement Approach-International Journal of Computer Science & Communication, Vol.1, No.1, January-June 2010, pp.229-233
- [3] Raman Kumar,Roma Jindal ,Abhinav Gupta , SagarBhalla, HarshitArora: A Secure Authentication System-Using Enhanced One Time Pad Technique –IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.2, February 2011
- [4] SharadPatil ,ManojDevare , Ajay Kumar :Modified One Time Pad Data Security Scheme :Random Key Generation Approach –International Journal of Computer Science and Security (IJCSS) , Volume (3):issue(2)
- [5] N.J.Croft and M.S.Olivier:Using an approximated One-Time Pad to Secure ShortMessaging service(SMS)-SATNAC 2005 Proceedings
- [6] Jeff Connelly : A Practical Implementation of a One-time Pad Cryptosystem-CPE 456 , June 11,2008
- [7] William Stallings : Cryptography and Network Security Principles and practices ThirdEdition - Pearson Education
- [8] Bruce Schneier :Applied Cryptography Second Edition-john wiley& sons, Inc.
- [9] Jonathan Katz , Yehuda Lindell : Introduction to Modern Cryptography,Chapman&Hall/CRC Taylor & Francis Group
- [10] Raj Jain : The art of Computer Systems Performance analysis - John Wiley & Sons , Inc.
- [11] Ajitsingh and RimpleGilhotra : Data Security using private key Encryption system based on arithmetic coding- International Journal of Network Security & its applications(IJNSA),vol.3,No.3,May 2011
- [12] ShaikRasool, G.sridhar ,K.Hemanth Kumar , P.Ravi Kumar :Enhanced Secure algorithm for Message Communication-International journal of Network security &its applications (IJNSA), vol.3.No.5,sept 2011
