NEW APPROACH IN SYMMETRIC BLOCK CIPHER SECURITY USING A NEW CUBICAL TECHNIQUE

[1]Ali M Alshahrani and Prof. Stuart Walker

Computer Science and Electronic Engineering,
University of Essex,
Wivenhoe Park, Colchester, Essex, UK, C04 3SQ

ABSTRACT

Cryptography is a security technique that must be applied in both communication sides to protect the data during its transmission through the network from all kinds of attack. On the sender side, the original data will be changed into different symbols or shapes by using a known key; this is called encryption. On the other communication side, the decryption process will be done and the data will be returned to its former shape by using the agreed key. The importance of cryptography is to fulfil the communication security requirements. Real time applications (RTA) are vulnerable for the moment because of their big size. However, some of the current algorithms are not really appropriate for use with these kinds of information. In this paper, a novel symmetric block cipher cryptography algorithm has been illustrated and discussed. The system uses an 8x8x8 cube, and each cell contains a pair of binary inputs. The cube can provide a huge number of combinations that can produce a very strong algorithm and a long key size. Due to the lightweight and fast technique used in this idea, it is expected to be extremely rapid compared to the majority of current algorithms, such as DES and AES.

KEYWORDS

Home of keys, Shared secret key, Encryption key, symmetric, block size.

1. INTRODUCTION

Voice, video, images, and text are examples of real time applications that need to be transmitted quickly with a high level of security. Nowadays, there are many applications that provide real time services, such as Skype and Tango. On the other hand, there are many suggested algorithms that can be used to protect these applications and to guarantee that unauthorized persons cannot access these services. Cryptography is one of the important techniques used to protect the data during its transmit from sender to receiver by changing the situation of the message into another shape. This process can be done in right way if both the communication sides know the key used to encrypt/decrypt the message. DES, AES and Blowfish are examples of algorithms that deal with the symmetric technique that is used to protect real time applications. Some of these, like DES, are no longer used, and some of them will be broken in the coming years. Therefore, an alternative algorithm must be implemented with a guarantee that it will fulfil all security requirements, such as integrity, confidentiality, authentication, and non-repudiation [1].

There are two different types of cryptography: symmetric and asymmetric techniques. In this paper, symmetric cryptography will be used. Symmetric key encryption, commonly called secret or conventional encryption, refers to the type of encryption where the keys of encryption and de-

cryption have the same values. Also, a symmetric key can contain a stream cipher and a block cipher; the block cipher will be used here. The concept of a block cipher is to divide the text into fairly bulky blocks, for instance, 128 bits, then encode every block individually[2][3].

The act of cracking or breaking a cipher involves extracting the plain text from the cipher text with no knowledge of the key and frequently with no encryption algorithm information. The strength of a cipher is evaluated according to the time required to break it, with the emphasis on the length of time it takes to break the cipher rather than whether it is possible to break it as, probably, all recognized ciphers except for some very few protocols, may be easily cracked or broken[6].

Strong ciphers are those that require an extended period of time to be broken; however, they are also likely to be more complicated and hard to apply. In contrast, weak ciphers tend to be broken quickly; nevertheless, they are generally quite simple and uncomplicated to use. Furthermore, it is important to take into consideration the fact that any cipher's use has some operating expenses concerning time and processing demands[6][7].

In recent years, a cube has been used to create some algorithms that can be used to encrypt a specific type of data (i.e. images). The main idea was to use the cube as a message container. Consequently, in this paper, an 8*8*8 cube is used to generate the key that will be used to encrypt the message.

2. LITERATURE REVIEW

The Rubik's cube was created in 1974 for entertainment purposes. In 1992, the cube was used in cryptography by writing and jumbling the message on the cube. It was very new technique in cryptography and it has given rise to further new suggested techniques[13].

As mentioned previously, recently, many cryptography solutions have been implemented that use a cube, and many of these algorithms were created for a specific type of data, that is, images. The majority of techniques wrote the messages on the cube and then scrambled the plain text to get the cipher text. However, the original message in this case can be obtained by anyone who can solve the cube if the arrangement faces of the cube are known. Another disadvantage is that writing the message on the cube will help to get the plain text by linguistic analysis. This can be done by looking for the most frequently used letter, which in the English language is 'E' as it is repeated around 12% [13][14].

A paper titled A Secure Image Encryption Algorithm based on Rubik's Principle [15] suggests two secret keys that are generated randomly and that take the same number of rows and columns (M x N) as the plain text image. The technique uses the concept of a Rubik's cube to shuffle the pixels and then the keys will be generated by applying the bitwise XoR operation into odd and then even rows and columns respectively, which encrypts the image. The main advantages of it are the extreme speed and the fact that two kinds of keys are used. On the other hand, it is easy to break because of the simple implementation. Another paper, namely, New Approach and Additional Security to Existing Cryptography Using Cubical Combinatorics [16], has a shared key that is generated from many aspects: arrangement of letters on cube slides, arrangement of faces, arrangement of colours, steps needed to reach a solved cube, and cube angle which is then exchanged using another existing technique. This algorithm is written with the original message on the cube faces. The Rubik's cube shuffles them randomly. After scrambling the cube, any known algorithm will be applied; the authors used SHA-1 Hash. The benefits of this algorithm are that it can be used to encrypt text and may be applied to other types of data, it is fast, and it can be combined with other current and tested algorithms. However, the SHA-1 algorithm used in this tech-

nique is no longer in use. Moreover, the technique that was used to generate the key is easy to guess and break. The message is written on the face, and the plain text may obtained by anybody who knows how to solve the cube, especially with the 2*2*2 cube used in their experiment.

a. Advanced Encryption Standard (AES):

The AES based on the block cipher symmetric encryption technique was created in 2001 and is used by the US government as a standard. AES is considered secure and fast so most applications use it. Its block data size is 128-bits, and it has three different key sizes: 128, 192 and 256 bits. These different keys are generated by different numbers of rounds, that is, 10, 12 and 16 respectively[6].

3. THE SUGGESTED ALGORITHM

The suggested algorithm is symmetric block cipher cryptography that is more suitable for use with multimedia compared to the asymmetric technique because it is faster. The strongest feature of the suggested algorithm is not the key size, although it is long, but rather is the very complex technique used to generate the key. The possible combinations in this project are divided into two huge numbers. One is related to the cube itself and the other one is the key length. The suggested algorithm uses the cube to generate the key in a specific way that is completely different from the other current algorithms. Cube 8*8*8 has a 3.6×10^{217}possible number of permutations. Moreover, the number of bits in each single cell can be 1, 2, 4 and 8 bits. 2-bits will be used in this paper and can be represented as follows:

$M: ([1;8] \cap \mathcal{N})^3 \rightarrow \{0,1\}^2$.

The key will not be exchanged. The secret shared key will be exchanged instead. The master cube $8 \times 8 \times 8$ is a three-dimensional object limited by six square faces or sides (front, back, up, down, right, and left faces).

In this paper, each face consists of 128 bits, and the cube's centre facets contains of four faces with a total of 512 bits. The whole cube has 384 cells, and each cell has 2-bits so the maximum key length considered here is 768 or 1024 bits. The method used to mix up the home keys contains complex rounds that will be used to create the E_K. Actually, the rounds are divided into two groups, and the total number of rounds is 10. The first group states the first four rounds which are achieved by shifting the odd rows/columns by even numbers and vice versa. The second group carries out the remaining rounds by XoR-ing each pair of faces, and stores the result in a different face. Using fewer rounds means spending less time so all or some of the rounds can be applied.
The method of reading the cube means it is very important to specify the key length. Here, two different ways are considered. The first method uses a face by face process as a key, starting with 128-bits for the first face and ending with 768-bits. The second reading mechanism of the cube to obtain the 1024 bits is to read the cube's centre facets in horizontal and vertical directions. Each face's cells can be shifted in 12 different directions as follows:

No.	Direction	Name
1	→	Rightwards
2	←	Leftwards
3	↑	Upwards
4	↓	Downwards
5	↗	North East
6	↙	South West
7	↖	North West
8	↘	South East
9	↴	Right Down (apply in the same face)
10	↱	Right Up (apply in the same face)
11	↳	Left Down (apply in the same face)
12	↰	Left Up (apply in the same face)

Table 1: cube movement directions

Encryption Algorithm:

Encryption Algorithm: secret shared key to mix-up the key's home and apply the selected rounds.

Require 1: X is a selected block size and could be any values between 128 and 1024 bits.
Require 2: Shared Secret Key (SSK) to mix-up the key's home. $SSK = \{n \in Z\hat{} + \mid 1 \leq n \leq 7\}$.
Require 3: Pre-define place to hide the SSK.

1. Select the plain text (PT) size, which must be equal to the key length (K).
2. Agree shared secret key (SSK).
3. The keys' home cube will be jumbled up by the SSK and applied to the selected rounds (R).
4. The result of point 3 above will be the key (EK).
    5. The generated key will be XoR-ed with the plain text. EK: $K \oplus P + R \rightarrow CT$
    6. The result of (5) operation will be the ciphered text: CT
    7. Then, the shared secret key in point 2 above will be injected into the ciphered text in a pre-known place: $C + SSK$
8. The result of 7 will be sent to the receiver: $CT + SSK \rightarrow Receiver$.

The figure below illustrates the algorithm.

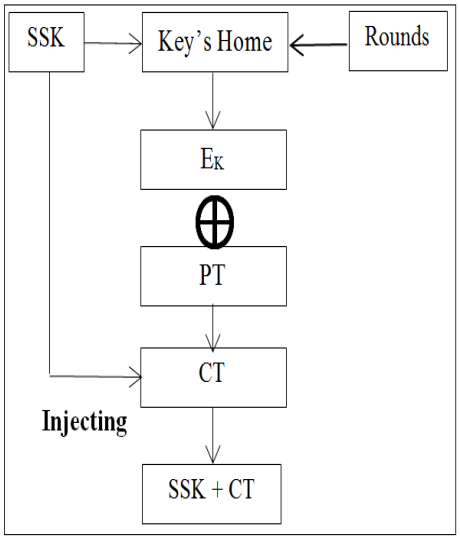

Figure 1: Encryption process.

Decryption Algorithm:

Decryption algorithm:

Require 1: Pre-defined place to hide the SSK

The decryption process will be done in the receiver side as the following steps:

1- The receiver will receive $CT + SSK$.
2- Then, the receiver will extract the SSK that has been hidden in a pre-defined place. As a result, the receiver will separate the SSK from CT.
3- The SSK that is obtained from the previous point will be used to shuffle the home of the keys.
4- The K will be produced.
5- $K \oplus C \rightarrow PT$

The figure shows the decryption process.

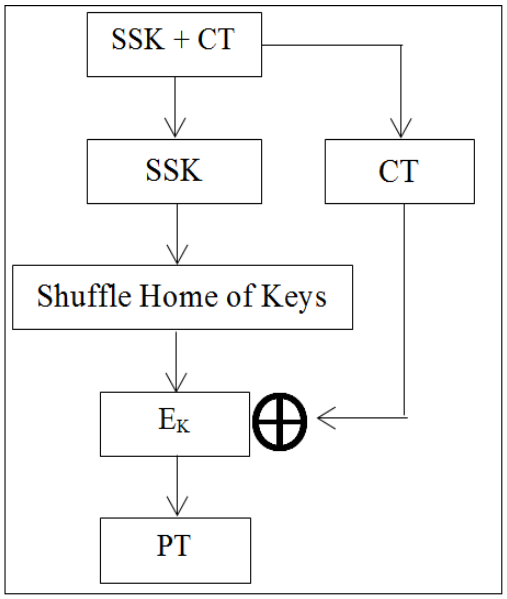

Figure 2: Decryption process.

4. RESULT AND EVALUATION

Using Java, the suggested algorithm has been tested in two different packet /key sizes of 128, 256 and 512-bits. The tested file size is 10,000 bits. Only the first group of rounds has been applied, and the result was shown in the below chart. According to the result, the time required to encrypt and decrypt the message was less than 7.0 milliseconds with the longest suggested encryption key. On the other hand, only 2.1 milli-seconds the key size of 128 bits take to do the same process in the same encrypted file.

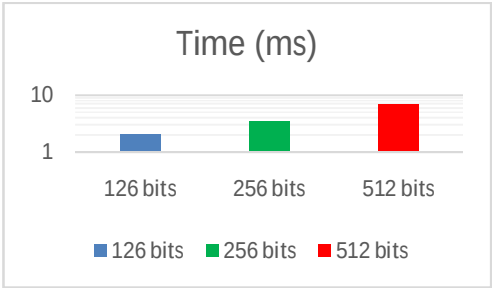

Figure 3: The experiment result

From the above result, it is clear that the time needed to encrypt and decrypt the files in the suggested algorithm referred to as ITU-T in its G114 recommendations was very reasonable. Moreover, it takes around a third of the time needed by AES to encrypt the files.

5. CONCLUSION

In this paper, a novel cubic symmetric block cipher technique is discussed. The greatest advantages of this technique are the length of the key, the technique used to generate the key, and that the suggested algorithm and tools are very lightweight and so are inexpensive. Moreover, the algorithm can be used with all different kinds of data because it will not cause any specific delay. Two keys are used to produce this strong algorithm. The first one is a shared key that will be used to scramble the home keys to get the key that will be XoR-ed with the plain text. Compared to the AES, less time is required to encrypt and decrypt the tested message. However, the delay will be affected by the number of rounds that will be used to generate the key so it is highly recommended that fewer rounds be used.

REFERENCES

[1] Stinson D.,(2003), " Cryptography Theory and Practice", CRC Press Inc., NY, USA.
[2] E.Cole, R. Krutz and J.W.Conley,(2005), " Network Security Bible", Wiley Publishing Inc,.
[3] A.D Elbayoumy, Sch. of Eng. Design & Technol., Bradford Univ, “QoS control using an end-point CPU capability detector in a secure VoIP system”, 10th IEEE Symposium on Computers and Communications (ISCC 2005).
[4] J.Evans, and C.Filsfils, Deploying IP and MPLS QoS for Multiservice Networks: Theory and Practice. Francisco: Morgan Kaufmann, 2007.
[5] A.C Rumin and E.C. uy, “Establishing How Many VoIP Calls a Wireless LAN Can Support without Performance Degradation,” 2nd ACM international workshop on Wireless Multimedia networking and performance modelling, pp. 61-65, Oct. 2006.
[6] Stallings W., (2009),“Cryptography and Network Security Principles and Practices”, Fourth Edition; Pearson Education; Prentice Hall.
[7] Yehuda L., Jonathan K., (2007).’ Introduction to Modern Cryptography’, CRC Press.
[8] K,R.Lars, ‘The Block Cipher Companion’, Springer. ISBN 9783642173417, (2011).

[9] H.Wang ; H.Zheng ; B.Hu ; H.Tang .'' Improved Lightweight Encryption Algorithm Based on Optimized S-Box', Computational and Information Sciences (ICCIS), 2013 Fifth International Conference. Page(s): 734 - 737
[10] T.Sharma. ; R. Thilagavathy, 'Performance analysis of advanced encryption standard for low power and area applications', Information & Communication Technologies (ICT), 2013 IEEE Conference, 2013 , Page(s): 967 – 972.
[11] S.Verma, R.Choubey, R.soni, 'An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security', International Journal of Emerging Technology and Advanced Engineering Web-site: www.ijetae.com (ISSN 2250-2459, Volume 2, Issue 7, July 2012).
[12] Gollmann, D.(2006), 'Computer Security', second edition, John Wiley and Sons.
[13] Douglas W., (1992), 'Cryptologia', "Rubik's Cube" As A Transposition Device', pp. 250-256.
[14] Extended essay, 'cryptography and Rubik's Cube: An Investigative Analysis', 2008.
[15] K.Loukhaoukha, J.Chouinard, A. Berdai, 'A Secure Image Encryption Algorithm Based on the Rubik's Cube Principle', Journal of Electrical and Computer Engineering 01/2012; DOI:10.1155/2012/173931.
[16] P Elayaraja, M Sivakumar, 'New Approach and Additional Security to Existing Cryptography Using Cubical Combinatorics', Master of Computer Applications; Dhanalakshmi College of Engineering, Chen-nai.
[17] S.Kilaru, Y.Kanukuntla, A.Firdouse, M.Bushra & S.chava, 'Effective and Key Sensitive Security Algorithm For An Image Processing Using Robust Rubik Encryption & Decryption Process', ISSN (Print): 2278-8948, Volume-2, Issue-5, 2013.