STEGANOGRAPHY BASED ASYMMETRIC KEY CRYPTOSYSTEM USING TRELLIS CODED GENETIC ALGORITHM RANDOM KEY GENERATOR

Anusha T and Venkatesan R

Department of Computer Science and Engineering, PSG College of Technology, Coimbatore

ABSTRACT

This paper focuses on generating a random bit sequence using Trellis coded Genetic Algorithm (TCGA) with boolean function as source of input. Randomness of the generated bit sequence is tested using the methods proposed by National Institute of Standards and Technology (NIST) to be used as random key for cryptographic applications. Generated random key is transmitted using Blue Pixel Least Significant bit (BPLSB) steganographic technique. The extracted random key is then used for image encryption and decryption for asymmetric key cryptosystem.

KEYWORDS

Genetic Algorithm, Image Encryption, Random Key, Steganography, Trellis Code

1. INTRODUCTION

Random keys for any cryptographic application can be generated either in deterministic or non-deterministic way. Keys generated non-deterministically are considered as true random key and deterministically is considered as pseudo random key[12]. In this work, it is generated using Boolean Functions and Trellis Coded Genetic Algorithm (TCGA) random key generator. Trellis codes are generated using state machine which takes 8-bits as input and generates 32 bits as output. Generated random key is in risk of being detected when the state machine gets exposed. To avoid it genetic algorithm is applied over the Trellis Code (TC) to increase the complexity. It is transmitted using BPLSB steganographic technique, which is then extracted to be further used for any cryptosystem. In this paper, we have used the generated random key for Elgamal Cryptosystem for image encryption and decryption [12].

2. LITERATURE REVIEW

2.1. TRUE RANDOM NUMBER GENERATOR

Chaotic optic waveform is sampled in an all-optical domain and then random bit streams are generated in comparison to a threshold level [8].Laser chaos are used as a non-deterministic random bit source, by making use of large intensity fluctuation[9].

2.2. GENETIC ALGORITHM CROSS OVER OPERATORS

Genetic algorithm can be used for key generation. Two chromosomes are randomly selected and crossover operator is applied over randomly selected positions to generate a random key[1].Pseudo random binary sequence is generated on the basis of current date using millisecond function [2].Multistage algorithm combines various techniques with genetic programming making cryptanalysis difficult[4].Crossover operator of genetic algorithm[10] is used for shuffling the generated bit sequence. We have used single point crossover, two point crossover and uniform crossover chosen randomly while generation of random key.

2.3. TESTS FOR RANDOMNESS

2.3.1. NIST RANDOMNESS TEST

Randomness of the bits generated can be tested using the following tests proposed by NIST [3].

a. **FREQUENCY TEST-** The purpose of this test is to determine whether the number of ones and zeroes in a sequence is approximately the same as would be expected for a truly random sequence.

b. **FREQUENCY BLOCK TEST**-The purpose of this test is to determine whether the frequency of ones in an M-bit block is approximately M/2 as would be expected under an assumption of randomness.

c. **LONGEST RUNS TEST**-The purpose of this test is to determine whether the number of ones and zeros of various lengths is as expected for a random sequence.

d. **SPECTRAL TEST**-The purpose of this test is to detect periodic features in the tested sequence that would indicate a deviation from the assumption of randomness.

e. **NON-OVERLAPPING TEMPLATE TEST**-The purpose of this test is to detect generators that produce too many occurences of a given non-periodic pattern.

f. **OVERLAPPING TEMPLATE TEST**-The focus of the Overlapping Template Matching test is the number of occurences of pre-specified target strings.

g. **APPROXIMATE ENTROPY TEST**-The purpose of this test is to compare the frequency of overlapping blocks of two consecutive/adjacent lengths against the expected result for a random sequence

2.3.2. HYPOTHESIS TEST

Hypothesis testing [14] refers to the predefined formal procedures that are used by statisticians to accept or reject the hypothesis. Generated code is considered as random key if it has high entropy value. By stating Null and alternate hypothesis as

H0:TCGA code is random
H1:TCGA code is not random
Correctness of detecting whether the generated sequence is random done as follows:
True Positive: Generated code is random and is detected as random
True Negative: Generated code is random and it is detected as non-random
False Positive: Generated code is not random and it is detected as random

False Negative: Generated code is not random and it is detected as not random

If the generated code has p-value greater than 0.01, then it is considered as random key.

2.4. BENCHMARK AND METRICS FOR RANDOMNESS

Metrics that are considered for randomness are entropy, and probability valu (p-value).Sequence with entropy value and p-value greater than 0.5 and 0.01[3] respectively is considered as random.

2.5. GENERATION OF STEGO IMAGE

The message gets embedded in the Least Significant Bit(LSB) of the image at random positions based on generated Pseudo Random Numbers (PRN)[7].Encrypted message is embedded into an image using pseudo random numbers in LSB of image [11].Watermark is embedded into the bin of selected group of pixels[13].

2.6. ELGAMAL CRYPTOSYSTEM

There are various cryptographic applications such as Digital Signature, Elliptic Curve Cryptography and Elgamal Cryptosystem. In this paper, we have used generated random key for image encryption and decryption using Elgamal Cryptosystem[13]. Elgamal Cryptographic System is the public key cryptographic system. Random Integer X acts as private key and {q, a, Y} as public key for user A.Y is calculated using the formula:

$$Y = a^X \bmod q \quad (1)$$

where a is the primitive root of Q.

Encryption is carried out using the formula

$$C1=a^k \bmod q \quad (2)$$

$$C_2=K*M \bmod q \quad (3)$$

Where C_1 and C_2 are the resultant integers of encryption, M is the message to be encrypted and the one-time K is calculated as $K=Y^k \bmod q$ where k is the random integer such that $1 \leq k \leq q-1$

Decryption is carried out after recovering the key by computing

$$K=(C_1)^X \bmod q \quad (4)$$

$$M=(C_2K^{-1}) \bmod q \quad (5)$$

3. PROPOSED SYSTEM

3.1. DESIGN

Our proposed system uses min terms of a boolean function in generating random sequence of bit. Trellis coded encoder takes 8-bit input and outputs 32-bit random key that can be used in a cryptosystem. In this system 8-bits are selected from the bit positions randomly in the generated random sequence and the random key is generated using TCGA technique. The machine begins in a state and processes the bits of the input sequence one at a time. As it processes each input bit, it outputs four bits and changes state. If the input is a 0, it traverses through an arc coming from the current state and outputs the four bits with which that arc is labeled. If the input bit is a 1, it traverses through another arc and outputs that arc's four bit label. Thus an 8-bit message will be transformed into a 32 bit code word after encoding[6]. In this paper, random bit sequence is splitted into parental chromosomes each of 8 bits and are mated to form new chromosomes and generated bit sequence is shuffled to form another 32-bit random key which

is more random than the one previously generated. Generated TCGA code is tested for randomness to be used for any cryptographic application. The key is securely transmitted by embedding using BPLSB technique. It is then extracted and fed as input for Key Pair Generator. Key pair is generated for every 4 bit of the random key. Image is encrypted using public key and decrypted using private key. In this paper, we have used Elgamal cryptosystem for key pair generation, image encryption and image decryption.

3.2. GENERATION OF TRELLIS CODE

A simple random key generation system is designed by using state machine that generates Trellis Code. The machine begins in a state and processes the bits of the input sequence one at a time. As it processes each input bit, it outputs four bits and changes state. If the input is a 0, it traverses through an arc coming from the current state and outputs four bits with which that arc is labeled. If the input bit is a 1, it traverses through another arc and outputs that arc's four bit label. Thus an 8-bit message will be transformed into a 32 bit code word after encoding[5].

3.3. ROLE OF GENETIC ALGORITHM

The risk of exposure of state machine is overcome by making use of any of the following crossover operators selected randomly and applied over the 32-bit generated Trellis code to generate TCGA random key:

- Single Point Crossover
- Two Point Crossover
- Uniform Crossover

3.4. TESTS FOR RANDOMNESS

In this paper, we have used the following NIST tests for testing the randomness of the generated sequence:

- Frequency Test
- Frequency Block Test
- Longest Runs Test
- Spectral Test
- Non-Overlapping Template Matching Test
- Overlapping Template Matching Test
- Approximate Entropy Test

Also hypothesis testing is performed.

3.5. ROLE OF STEGANOGRAPHY

3.5.1. EMBEDDER

Steganographic system embeds the encoded message bits into the cover work. In this work, the message is embedded in the Least Significant Bit (LSB) of the selected Blue pixels of an RGB image.

3.5.2. EXTRACTOR

Extraction of 32 bit random key from a stego image is done by examining the pixel values selected during embedding process. Least Significant Bit (LSB) of 32 blue pixel values are extracted to generate key pair.

3.5.3. GENERATION OF PUBLIC KEY/PRIVATE KEY PAIR

Random key is fed as input to key pair generator to generate {private key, public key} pair .For every four bits, corresponding key pair is generated to be used in image encryption process.

3.5.4. IMAGE ENCRYPTION

Every pixel of an image is encrypted using public key to generate two encrypted files.

3.5.5. IMAGE DECRYPTION

Two encrypted files are used to get the original image by using corresponding private key for every pixel.

4. DISCUSSION

4.1. LENGTH OF RANDOM KEY

Since the length of key generated in this method is 32 bits, it is predictable by the attacker by brute force approach. It is predictable in $O(1)$,$O(2^n)$ and $O(2^{n-1})$ in the best, worst and average cases respectively, where n is the length of key in bits. Since the key is hidden within the steganographic image using stego key,32 bits key length is considered sufficient.

4.2. IMPERCEPTIBILITY

Since only the LSB of Blue pixels are changed, the embedded random key is imperceptible in the watermarked work.

4.3. SECURITY

The key hidden within the image can be detected when only the algorithm and stego key is known to the detector. Both the stego-key and the algorithm should be kept secret. Generated random key is tested for randomness using frequency block Test, longest runs test, overlapping template Test, non-overlapping template test, overlapping template test and approximate entropy Test. In all these tests calculated p-value is tested for it to be greater than or equal to 0.01 to consider the sequence of bits to be random. The user retrieves the embedded 32 bits and uses for any cryptographic application.The security of the stego-key depends on its own size as well as the size of the cover work.

4.4. ACCEPTANCE OF HYPOTHESIS

Null Hypothesis is accepted for all tests.

4.5. SELECTION OF CROSSOVER

Selection of Crossover is random and so for a single boolean function, many different random keys can be generated, which is difficult for the hacker to determine the key even if the boolean function is compromised.

4.6. RESULT

Table 1: Generation of Random Key from Boolean Function using Trellis Coded Genetic Algorithm

Sl.No.	Boolean Function	Key No.	TCGA Key	Entropy
1	Ab+Bc+c	K1	01001111100000011110111111101111	0.9284
		K2	00000101110010111110111111101111	0.9284
		K3	00000011110011011110111111101111	0.9284
2	A+BC	K4	00110001000010001011111011111101	0.9972
		K5	00100001000110001011111011111101	0.9972
		K6	00000001001110001011110111111110	0.9972

Table 2: NIST Tests for randomness

Test No.	Tests
Test1	Frequency Test
Test2	Frequency Block Test
Test3	Longest Runs Test
Test4	Spectral Test
Test5	Non-Overlapping Template Test
Test6	Overlapping Template Test
Test7	Approximate Entropy Test

Table 3:p-value for NIST Tests for Randomness

Key No.	Test1	Test2	Test3	Test4	Test5	Test6	Test7
K1	0.0771	0.0362	0.0103	0.0516	0.3442	0.0916	0.0959
K2	0.0771	0.1730	0.0204	0.0516	0.3442	0.1293	0.0227
K3	0.0771	0.0362	0.0184	0.0516	0.3442	0.0916	0.2054
K4	0.7237	0.1730	0.0204	0.7456	0.0408	0.0976	0.5877
K5	0.7237	0.1730	0.0204	0.7456	0.1184	0.4996	0.5877
K6	0.7237	0.0821	0.0195	0.7456	0.3442	0.3680	0.2447

Table 4: Hypothesis Testing

Key No.	Test1	Test2	Test3	Test4	Test5	Test6	Test7
K1	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive
K2	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive
K3	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive
K4	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive
K5	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive
K6	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive	True Positive

5. CONCLUSIONS

Random key can be generated by using TCGA by using boolean function as source of input and it can be used for image encryption and decryption using Elgamal Cryptosystem. Further research can be carried on using TCGA by using image, video, audio as source of input and using the key for any cryptographic application.

REFERENCES

[1] Aarthi Soni, Suyash Agrawal, "Key Generation using Genetic Algorithm for Image Encryption", International Journal of Computer Science and Mobile Computing ,Vol 2,Issue 6,June 2013.

[2] Aarthi Soni, Suyash Agarwal, " Using Genetic Algorithm for Symmetric Key Generation in Image Encryption", International Journal of Advanced Research in Computer Engineering & Technology(IJARCET),Volume 1 ,Issue 10,December 2012.

[3] Andrew Rukhin,Juan Soto,James Nechvatal,Miles Smid,Elaine Barker,Stefan Leigh,Mark Levenson,Mark Vangel,David Banks,Alan Heckert,James Dray,San Vo, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications",April 2010

[4] S.Devi, Palanisamy, "Multi-level Encryption using SDES Key Generation Technique with Genetic Algorithm", International Journal of Engineering and Computer Science", ISSN:2319-7242,Volume-3 Issue-8 ,August 2014

[5] Ingemar.J.Cox, Matthew.L.Miller, Jeffrey.A.Bloom, Jessica Fredrich and Tonkalker, "Digital Watermarking and Steganography",Margan Kaufmann Publishers,New York,2008.

[6] Masahito Kayasi, Shun Watanabe, "Uniform Random Number Generation from Markov Chains-Non Asymptotic and Asymptotic Analysis", IEEE Transactions on Information Theory, Vol 62,No.4, April 2016.

[7] Nasir Memon, Ping Wah Wong, "A Buyer-Seller Watermarking protocol", IEEE Transactions on Image Processing, Vol 10,No.4, April 2001.

[8] PuLi,YuanyuanSun,XianglianLiu,XiaogangYi, Jianguo Zhang, XiaominGuo, YanqiangGuo,Yuncai Wang,"Fully Photonics-based physical random bit generator",July 2016.

[9] PuLi,YuanyuanSun,XianglianLiu,XiaogangYi,JianguoZhang,XiaominGuo,YanqiangGuo,Yuncai Wang,"Brownian motion properties of ptoelectronic random bit generators based on laser chaos",Article in Optics Express 24(14):15822-15833 • July 2016.

[10] S.N.Sivanandam,S.N.Deepa, " Principles of Soft Computing",Wiley India

[11] Unik Lokhande, A.K.Gulve, "Steganography using Cryptography and Pseudo Random Numbers", International Journal of Computer Applications, Volume 96,-No 19,June 2014.

[12] William Stallings, "Cryptography and Network Security-Principles and Practice",Fifth Edition,Pearson.

[13] Tionrai Zhong, Yong Xiang, Iynkaran Natgunanathan, Song Guo,WanleiZhou, GlebBeliakov, "Robust histogram based method for Image Watermarking", IEEE Transactions on circuits and systems for video technology",Vol 25,No.5,May 2015

[14] https://math.tutorvista.com/statistics/hypothesis-testing.html

AUTHORS

Ms.T.Anusha completed her B.E degree in Computer Science and Engineering from Dr.Sivanthi Aditanar College of Engineering, Tiruchendur and M.E degree in Computer Science and Engineering from Manonmaniam Sundaranar University, Tirunelveli. She is currently working as an Assistant Professor(Sr.G) in the Department of Computer Science and Engineering in PSG College of Technology, Coimbatore

Dr.R.Venkatesan completed his B.E degree in Mechanical Engineering from Coimbatore Institute of Technnology, M.E degree in Industrial Engineering from PSG College of Technology, M.S in Computer Science from University of Michigan and PhD in Information and Communication Engineerin g from PSG College of Technology, Coimbatore. He is currently working as Professor & Head in the Department of Computer Science and Engineering.