ISSN 2321-807X
Volume 13 Number10
Journal of Advances in chemistry

Table Substitution Box Method for Increasing Security in Interval Splitting Arithmetic Coding

E.Wiselin Kiruba[a],DR.K.Ramar[b]
aAsst Professor
Department of Computer Science and Engineering,
University VOC college of Engineering, Tuticorin, Tamilnadu, India-628008
e-mail:wisekiruba@yahoo.co.in
b Professor&Principal
Department of Computer Science and Engineering, Einstein college of Engineering,
Tirunelveli, Tamilnadu, India- 627012
e-mail:kramar.einstein@gmail.

ABSTRACT

Amalgamation of compression and security is indispensable in the field of multimedia applications. A novel approach to enhance security with compression is discussed in this research paper. In secure arithmetic coder (SAC), security is provided by input and output permutation methods and compression is done by interval splittingarithmetic coding. Permutation in SAC is susceptible to attacks.Encryption issues associated with SAC is dealt in this research method. The aim of this proposedmethod is to encrypt the data first by Table Substitution Box (T-box) and then to compress by Interval Splitting Arithmetic Coder (ISAC). This method incorporatesdynamic T-box in orderto provide better security.T-box is a method,constituting elements based on the random output of Pseudo Random Generator (PRNG), which gets the input fromSecure Hash Algorithm-256(SHA-256) message digest. The current scheme is created, based on the key, which is known to the encoder and decoder. Further, T-boxes are created by using the previous message digest as a key. Existing interval splitting arithmetic coding of SAC is applied for compression of text data. Interval splitting finds a relative position to split the intervals and this in turn brings out compression. The result divulges that permutation replaced by T-box method provides enhanced security than SAC. Data is not revealed when permutation is replaced by T-box method. Security exploration reveals that the data remains secure to cipher text attacks,known plain text attacksand chosen plaintext attacks. This approach results in increased securityto IntervalISAC.Additionally the compression ratio is compared by transferring the outcome of T-box to traditional arithmetic coding.The comparison proved that there is a minor reduction in compression ratio in ISAC than arithmetic coding.However the security provided by ISAC overcomes the issues of compression ratio in arithmetic coding.

KEY WORDS: Encryption, compression, sender,decoder,key,messagedigest

1.INTRODUCTION

The digitalized world has increased the motivation to do research in the fields of compression and security, put together.Nowadays, the world is focusing on transmitting a large amount of datafor the rapid transmission of files. So, compressionhas become essential. Besides, to ensure the confidentialityof the data, security is needed. Therefore, compression together with security provides quicker and safer transmission. Encryption and compression are done simultaneously and as a result the processing time is reduced and execution is done faster[13].Moreover, combining compression and securityis helpful in the application of multimedia [1]. For the past few years, encrypting and compressing data have been a serious issue [14].Complexity increases as a consequence of combiningthe traditional secure algorithm with the compression method [1].The key problem faced by the current methods of encryption and compression is reduced speed, high cost and delayed execution [13].Huffman and arithmetic coding are a better choiceto achieve security, using compression with less coding efficiency [1]. Arithmetic coding is well known for its worthwhile, high coding efficiency and its application is widely used in many compression tools, such as jpeg2000 and h.264 standards [8].Nevertheless,stand-alone arithmetic coding cannot provide efficient encryption [19-21].This paves a new path for the modification of arithmetic coding.Wen etal hasdeveloped binary interval splitting arithmetic coding by splitting intervals,corresponding to a symbol withthe agreed key of the sender and receiver.Kim et al hasadded permutation to binary Interval Slitting Arithmetic Coding and incorporated security to the encoder. Jiantao et al has illustrated susceptibilityof thechosen ciphertext attack in secure arithmetic coding. Data is at risk while decoding the encrypted data alongwith compression [11].The above mentioned literature surveys deal with the impact on security issues of SAC.

The aimof this paper is to enhance security of ISACusing aT-box. To achieve this,the entire messages are broken into blocks of 128bits. The blocks are encrypted, using different T-boxes. Initially, aT-box is created by the user's key.SHA-256 makes the key as message digest .This message digest is given as an input to PRNG.The consecutiveT-boxes are created by getting theprevious message digest as key.The encrypted messages are sent to ISACfor compression.

The remaining part of this paper is organized as follows.Section 2 reviews Interval Splitting Arithmetic Coding.Section 3analyses the security issues of Interval SplittingArithmetic Coding. Section 4 presents T-box method. Section5 presents results and discussion.

DOI : 10.24297/jac.v13i10.5837
www.cirworld.com

2.ISAC

Splitting of intervals in arithmetic coding can be performed by having a key, commonfor both the sender and receiver. This approach isin contrast to the classical arithmetic coding, in which continuous intervals are used. To illustrate this,an input sequence,such as E, C, L, M, O, W,# is to be considered.

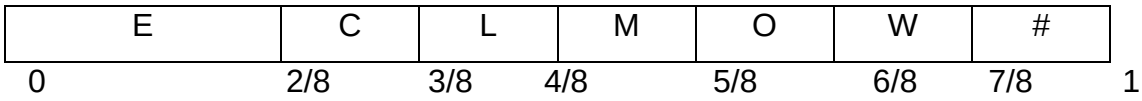

Fig 1.1 shows Traditional Approach of Arithmetic Coding

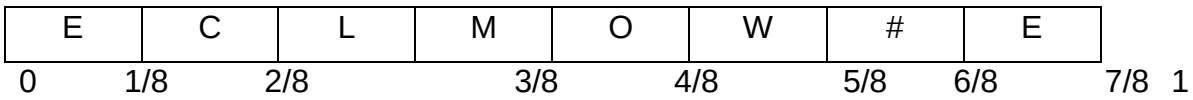

Fig 1.2 shows ISAC

Probability of p (E) =2/8 and the others are 1/8.In the Classical approach, the intervals in which the symbols are plotted, is placed in a continuous manner. Fig1.1clarifies that E is plotted in the range of 0-2/8 and the others are plotted in the range of (2/8 - 3/8 , 3/8 - 4/8, 4/8 - 5/8 ,5/8 - 6/8 ,6/8 - 7/8, 7/8 - 1). In Interval Splitting, if there are N symbols, the key vector becomesbecome k $\{0, 1\}^N$. Interval Splitting differs from the Classical Approach by splitting the first symbol with key k1 as follows:

i. Find the relative position(r) based on the key

$$r = \begin{cases} 2p(A)k1 & for\ 0 < k1 < 0.5 \\ p(A) + 2p(B)(k1 - 0.5) & for\ 0.5 < k1 < 1 \end{cases}$$

ii. Divide the intervals of the symbols based on r value

iii. The symbolsare systematically placed with intervals as shown in fig1.2

Interval Splitting maintains the constraints that the added value of the distinct intervals allocates to a symbol, based on its probability and it should be equal to its original probability.The restrictions ofarithmetic coding, in which the intervals are split, within the overall length of 0-1, is preserved.In Interval Splitting, the length of the code word(*lc*) should be equivalent to the following formula.

$$[-N \log_2 p(A)] \leq lc \leq [-N \log_2 p(B)] + 2$$

Refer [19-21]for further details .

3. SECURITY ISSUES

To incorporate security inSAC, input and output permutation are added to interval splitting. Input permutations are done on the plain text sequence. Permutation starts on the plain text symbols by Raster Order Mapping. Key based shifts are performed on rows and column outputs. Output permutations are done on the code word of ISAC. Code word permutation eliminates the last four bits from the code sequence of length *lc*[8] .Column and row shifts are applied to the remaining sequence of length *lc-4*.The resultant sequences are appended with the removed bits. Permutation of row and column is performed for a second time. Finally, a source code word is obtained.

The secure code word is subjected to various attacks. Ultimately, the chosen cipher text attack breaks the key vector ofthe row and column shift with complexity *O*(*l*)[[19-21]. ISAC, without permutation, is vulnerable to the chosen plain text[19-21].Split location can be easily revealed with the four two symbol input sequence. This makes the code word of SAC insecure.

4. PROPOSED METHOD -T-BOX

The proposed method aims at increasing security in SAC.Due to lack of security in SAC, permutation is removed and it is substituted by T-box. To achieve security the input sequence is divided into smaller blocks of 128 bits [12].Everyblock is exposed to different T-box [10].In the beginning of encoding , to generate the first T-box,the agreed key by the sender and receiver is processed by SHA-256.[2]. The message digest is obtained to seed the PRNG.The randomized output of PRNG fills the 16*16 T-box [18]. Successive T-box are created using the preceding message digest as keys.[4]The encrypted data acquired as output through table substitution, in turn, is fedto Interval Splitting Arithmetic Coder, whose output is the secure code word. While decoding to get the decompressed data the secure code is given to the reverse ISAC. Over again the output of reverse ISAC is provided for Inverse T-box .At this instant the plain text is acquired fromInverse T-box .The stages to develop T-box are explained as follows: 1.Dynamic T-box Creation 2.Table Substitution 3.Inverse T-boxcreation 4.Inverse Table Substitution. Fig 2 gives a schematic representation of the overall structure.

4.1.Stage 1: Dynamic T-box Creation

Initially, a 16*16 T-box is created with a key (K0) known to the encoder and decoder. SHA-256 processes the Key *K0*. With its message digest,PRNGfills the T-box cells in the range of 0-255.PRNG maintains the constraints in which its output should not be repeated. The previous message digest becomes the key for the successive T-boxes. [18]

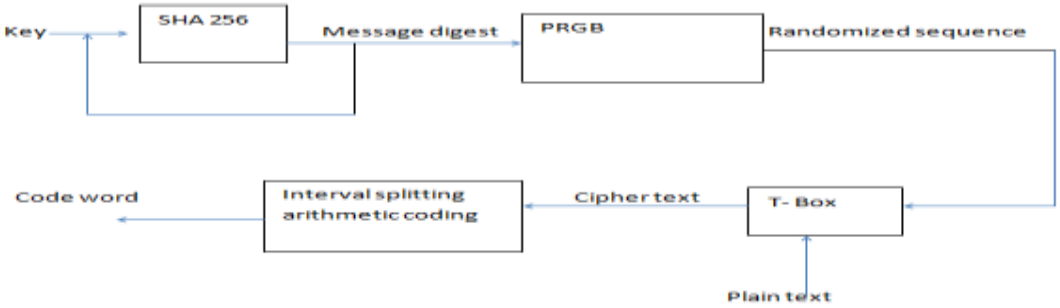

Fig2 shows proposed scheme of T box

The illustration of T-box creation is given below:

1. KEY GENERATION

1.1if block==1

1.1.1 Key=k0,chosen by sender and receiver.

1.1.2 Previous=key

1.2Else

1.2.1 Key=f(sha256(previous))

1.2.2 Previous=key

2. f(sha256(key))= MessageDigest

3. Repeat Till T-box==FULL

3.1IF(!(f(PRNG(MessageDigest)in T-box)

31.1T-Box Empty

=PRNG(MessageDigest);

3.2Else

3.2.1f(PRNG(MessageDigest))

4.2 Stage 2: Table Substitution

Theinput sequences are divided into blocks of 128 bits. 8-bit ASCII value of the first input symbol of thefirst block is split into two equal halves. Each 4 bit is interpreted separately as row and column respectively. Hence, the input ASCII value is replaced by the corresponding row and column value of T-box. For instance, the ASCII value of a 01100001 is split into 0110and0001.Here 0110 is considered as row and 0001 is considered as column. So, the value of the 6th row, the 1st column of T-box, is substituted by the input symbol. The process is repeated for the first blocks,using the same table .Substitution is done for the other blocks with the newly generated T-boxes, corresponding to those blocks [10].

4.3Stage 3: Inverse T-Box

For the first block, Key *K0* is processed with SHA-256 algorithm so as to produce the message digest.This is fed as an input to the PRNG. The output of PRNG is split into two equal halves, which represent the row and column respectively. In the cited row column of T-box, the order of PRNG output is filled. The message digest of the previous key acts as a key for the remaining inverse T-boxes [18]. The steps are as follows:

1. f(SHA-256 (key))= MessageDigest

2. Initialize count=1;

3. Repeat till T-box=FULL

3. 1If(!(PRNG(MessageDigest)in T-box)

3.1.1(PRNG(MessageDigest (1st halves))=ROW

3.1.2(PRNG(MessageDigest (2nd halves))=COLUMN

3.1.3T-box[ROW,COLUMN]=count

3.1.4Count++;

3.2Else

3.2.1F(PRNG(MessageDigest))

4.4Stage 4: Reverse Table Substitution

The output of Interval Splitting Decoder is subjected to Reverse Table Substitution. In the same way in the table Substitution, the output ofInterval Splitting Decoder is split into two halves torepresent the row and columnrespectively. Then, the corresponding row column value of thereverse substitution box is mapped to get the original value.[10]

5.RESULTS & DISCUSSION

After replacing permutation by T-Box Method, the following results will be carried out.For any crypto system, evaluating security will be a complicating task. Since revealing the strength of known attacks does not prevent the weakness againstunknown attacks [8].The encrypted data attained from T-box is laid to various types of cryptographicattacks [4].Moreover the compression ratio is also compared with arithmeticcoder. The results are explained as follows.

5.1 Cipher Text Attacks

In Cipher Text Attacks, the attacker gets the secret information from a cipher text, without knowing any key or encryption algorithm. The word text is encrypted with different keys, having a minutechange in bits. However thebits of code achievea major change.In Fig 3.1the bit difference is attained by comparing the cipher text of a word text with key K1 for the different keys. From the plotted values, it is interpreted that a small change inthe key has produced a significant change in the cipher text. Thus, the cipher text attack fails.

5.2 Known Plain Text Attacks

In Known Plain Text Attacks, the attacker tries to deduce the key from a set of plain textsand cipher texts. Since the same T-box is not used for the entire sequence, the output for the repetitive word text of a file getsaltered. The result for the Fig 3.2compares thebit difference of the cipher text of a word text with its repeated occurrence. As a result, the illustration shows that therepeated patterns have different cipher texts.The bitdifference is high, and thus it is complicated to get the key throughthe known plain text attacks.

5.3 Chosen Plain Text Attacks

In Chosen Plain Text Attack, the attacker attempts to decrypt the message to find the key from the chosen sequence of symbols. The plain text containing slight bit difference such as text1, text2, text3, text4, text5 are encrypted. The resultrelates the bit difference of theplaintext withthe cipher text for different texts. Consequently, the difference in thebits of compared plaintext is not reflected in its cipher text. Fig 3.3 reveals a drastic variation of patterns in the bit difference of the plain text and cipher text, plotted between the bit differnce of the texts.Therefore, the proposed approach provides a good level of security.

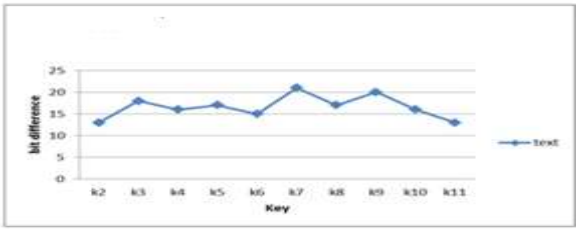

Fig 3.1 shows the bit difference between cipher text for different keys obtained from the bit difference of [k_1-$k_{(x+1)}$]

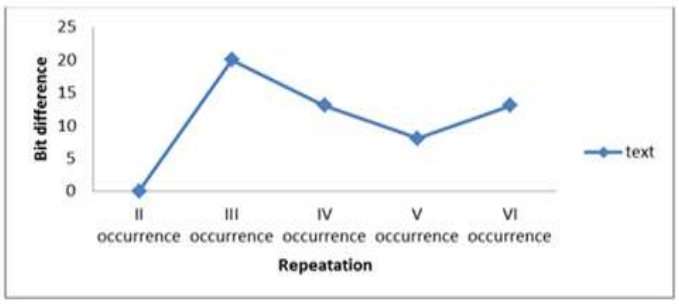

Fig 3.2 shows the plot between the difference of bits of repetitive text, obtained

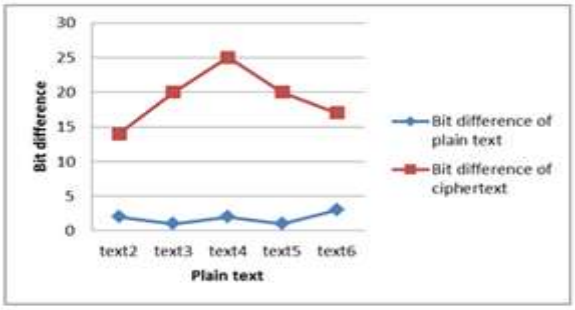

Fig 3.3 shows the drastic variation of patterns in bit difference for both plain text and cipher text

5.4Coding Competency

Table I Coding Competency

Original File Size(KB)	Reduced File Size After Compression(KB)	
	Arithmetic Coding Method	Interval Splitting Arithmetic Coding Method
5	3.15	3.55
10	7.5	8
100	65	66
1000	680	680.5
2000	1340	1340.1
5000	4400	4400.01

To analyze the coding competency the output of the T-box is subjected to ISACas well as arithmetic coder. From tableI the interpreted data showsthat the difference in thefile size by ISAC is little bit exceeding compared to thetraditional arithmetic coding.This difference can be minimized further to a large extent by increasing the length of the input sequence. Consequently, security is obtained, which is notaccessible in traditional arithmetic coding.

6.CONCLUSION

In this paper, the security issues of Interval Splitting are addressed by T-box method. The proposed system provides adequate level of security against cipher text attacks, known plain text attacks and chosen plain text attacks. Coding competency of smaller length input sequence is slightly lower in ISAC than traditional arithmetic coding. In future, there is an intention to do research byfusing security and compression in a single system with a high level of coding competency.

REFERENCES

1. H. A. Bergen and J. M. Hogan. , 1993. “A chosen plaintext attack on an adaptive arithmetic coding compression algorithm,” Computer Security, vol.12, pp. 157–167
2. Chaitya b. Shah , Drashti R. Panchal .Oct 2014."Secured Hash Algorithm-1: Review Paper”, International Journal For Advance Research In Engineering And Technology Volume 2, Issue X, (ISSN 2320-6802)
3. J. Cleary, S. Irvine, and I. Rinsma-Melchert. 1995. “On the insecurity of arithmetic coding,” Computer. Security., vol. 14, no. 2, pp. 167–180.
4. Gary C. Kessler . 2016. ”An Overview of Cryptography ”,
5. M. Grangetto, E. Magli, and G. Olmo. Oct. 2006. “Multimedia selective encryption by means of randomized arithmetic coding,” IEEE Trans. Multimedia,vol. 8, no. 5, pp. 905–917.
6. Hung-Min Sun, King-Hang Wang, Wei-Chih Ting . December 2009.”On the Security of the Secure Arithmetic Code” IEEE transactions on information forensics and security, vol. 4, pp 781`-789.
7. Jyotika Doshi et al. January 2012.” Practical Implementation of Faster Arithmetic Coding Using Total Frequency in Power of Two”, International Journal of Computing Science and Communication Technologies, vol.4 NO. 2.(ISSN 09743375)

8. H. Kim, J. T. Wen, and J. D. Villasenor.May 2007."Secure arithmetic coding," IEEE Trans. Signal Proc., vol. 55, pp.2263–2272.
9. Y. Mao et al. 2006."A joint signal processing and cryptographic approach to multimedia encryption," IEEETrans. IP, vol. 15, pp. 2061–2075.
10. M.Pitchaiah, Philemon Daniel, Praveen. , March -2012."Implementation of Advanced Encryption Standard Algorithm",International Journal of Scientific & Engineering Research, Vol. 3, Issue 3 (ISSN 2229-5518)
11. Poornima.P.V et al.May 2015." Security Enhanced Communication Scheme withError Correction Capability and Efficient Channel Utilization",International Journal of Computer Applications Volume 117 – No.14, (ISSN 0975 – 8887)
12. Rajesh R Mane.September 2015" A Review on Cryptography Algorithms, Attacks and Encryption Tools",International Journal of Innovative Research in Computer and Communication Engineering Vol. 3, Issue 9, (ISSN 2320-9801)
13. Ruchita Sharma, Swarnalata Bollavarapu,. May 2015 " Data Security using Compression and Cryptography Techniques", International Journal of Computer Applications (0975 – 8887) Volume 117 – No.14
14. E. Sankari, R. Shanmuga Priya, R. Suriya, G. Prabhakaran,T. Sowkarthik,Jan – Mar,2016."Enhancing Security through Data Hiding",International Journal of Advanced scientific research and developmentVolume 03, Issue 01, Version II, (ISSN2395-6089)
15. J. T. Wen, H. Kim, and J. D. Villasenor .Feb. 2006"Binary arithmetic coding with key-based interval splitting," IEEE Signal Proc. Letters, vol. 13, pp. 69–.
16. E.Wiselin Kiruba ,K.Ramar.2016."Enhancing Security for Gnome Data Using Referential Compression with Symmetric Cryptography Scheme", Asian Journal of Information technology",Volume 03, Issue 01,pp. 3449-3454
17. C.Wu et al. 2005."Design of integrated multimedia compression and encryption systems," IEEE Trans. Multimedia,vol. 7, pp. 828–839,.
18. Yuh-Ming Huang, Yin-Chen. 2011 ."A Secure Arithmetic Coding Algorithm Based on Integer Implementation",ISCIT/ 978-1-4577-1295-1/11/,pp 518-521
19. J. Zhou, O. C. Au, and P. H. Wong.Feb. 2008."Adaptive Chosen cipher text attack on secure arithmetic coding," Submitted to IEEE Trans. Signal Proc;
20. J. Zhou, O. C. Au, X. Fan, and P. H. Wong. , 2008."Joint security and performance enhancement for secure arithmetic coding," In Proceedings of the IEEE International Conference on Image Processing (ICIP), pp.3120-3123
21. J. Zhou, O. C. Au, P. H. Wong, and X. Fan. 2008. "Cryptanalysis of secure arithmetic coding," In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1769-177.

First Author:

E.iselin Kiruba completed ME in computer Science and Engineering. Now she is an assistant professor in University V.O.C College of Engineering, Thoothukudi.Currently doing research in multimedia security. She has published papers in reputed journals.Her area of interest are multimedia security,network security,image processingetc.

Second Author:

Dr. K.Ramar is a Ph.D. holder in Computer Science and Engineering. Now he is the Professor and Principal of Einstein college of Engineering. He has organized many workshops and faculty development programs for the technocrats. He has published many papers and has guided many research scholars. His areas of interest are image processing, pattern recognition, and Computer vision and machine intelligence.

DOI : 10.24297/jac.v13i10.5837
www.cirworld.com