International Journal on Future Revolution in Computer Science & Communication Engineering
Volume: 3 Issue: 10
ISSN: 2454-4248
234 – 237

Securing Text Transmission in E-learning through Natural Language Steganography: An Object Oriented Approach

Kh Amirul Islam[1]
Research Scholar, Dept. of Computer Science,
The University of Burdwan,
Burdwan, West Bengal, India
ramiz.amirul@gmail.com

Soumendu Banerjee[2]
Research Scholar, Dept. of Computer Science,
The University of Burdwan,
Burdwan, West Bengal, India
bansoumendu@gmail.com

Sunil Karforma[3]
Professor, Dept. of Computer Science,
The University of Burdwan,
Burdwan, West Bengal, India
sunilkarforma@yahoo.com

Akash Nag[4]
Research Scholar, Dept. of Computer Science,
The University of Burdwan,
Burdwan, West Bengal, India
nag.akash.cs@gmail.com

Abstract: With the increasing availability of Internet, the e-learning is also getting popularity at a high speed. But for a secure and efficient e-learning system security is a great matter of concern. Some of the important documents transmit between the participants of e-learning are text files like the user id of the learner or the teacher, password, private keys etc. If the hacker can reach these documents, they can get full access of the system which resulting fake marks sheet or admit card, which is very harmful for any e-learning institute. So text steganography is a good technique through which confidential and valuable data may be transacted securely by integrating text steganography along with AES block cipher. In this paper, we proposed a model for securing the transmission of text documents from the sender to receiver in an e-learning system wrapped with AES encryption algorithm, which provide better security.

Keywords: *E-learning, Text Steganography, AES algorithm, SNOW Algorithm*

*****

I. INTRODUCTION

With the increasing of availability of Internet, information and communication technology (ICT) is growing at a high speed and e-learning is an application of ICT in the field of learning. Now-a-days e-learning is gaining popularity at a high speed. The main issue in e-learning is security. Since, this field is totally based on Internet and it being publicly accessible, so security plays an active role. Steganography is a well known tool to provide security and text steganography is a sub part of steganography. Steganography can be classified into four types: Text steganography, Image steganography, Audio steganography and video steganography. Text steganography is such a technique through which the developer of an e-learning institution can hide the secret texts behind a cover text file. In case of e-learning, some texts like user id, password, secret keys are very essential and if it goes to wrong hand, then it would be very harmful for the student as well as for the institute[1]. Steganographic Nature Of Whitespace (SNOW)[2] is used as a tool for text steganography to analyze the field of natural language steganography[3,4]. This program is used for concealing messages and extracting messages in ASCII text file. It is a stream cipher which works on 32 bit words and supports both 128 bit and 256 bit keys[5]. In our proposed model, we have applied Advanced Encryption Standard (AES) algorithm instead of Information Concealment Engine (ICE) for doing the encryption and decryption. ICE is a symmetric key block cipher which uses 64 bit key whereas AES is also a symmetric key algorithm[6,7] and accept up to 256 bit key but for our model we have used 128 bit key. We design our model based on the secure transmission of user id and password between the developer and the learner in an e-learning institute with the help of modified SNOW algorithm. To achieve better security while sending the AES key to the learner, we apply RSA encryption technique and the public key is stored in the database of the e-learning institution. After Huffman encoding data length may not be a multiple of 8 bits, for which the data is padded with zeros and the number of zeros thus added is also encoded as white space from a 3-bit binary number, as the pad count is equal to 0 to 7.

 ISSN: 2454-4248
234 – 237

Section II describes the organization of the proposed model and section III contains the pseudo code of the program. Section IV includes the class diagram of our proposed model and in section V, we have shown some results of our proposed system. Finally, we conclude at section VI by showing some limitations and some future scopes of our proposed model.

II. ORGANIZATION OF PROPOSED MODEL

Our proposed model is based on the secure transmission of secret texts like user id or password or any other secret texts, hiding within a cover text, from the developer to the learner, related to the document which helps in authentication of the recipient in future related to the e-learning system. Now, security being a major issue for any successful e-learning system, here we apply a modified SNOW algorithm, replacing ICE algorithm by AES standard. The process flow of our proposed model is shown in fig1, in annexure. The SNOW program is free for non commercial use and it uses an open space method to embed data in a text file that means the secret texts are hidden in the trailing spaces of each line of text. The whole process can be divided by three parts: compression, encryption and encoding scheme.

The developer choose the secret text which is stored in the 'msg-in' file and the cover file named as 'cover' where the secret texts will make hidden. Then we apply Huffman coding, which is a lossless data compression algorithm and the main two applications are to build Huffman Tree from input characters and traverse the Huffman Tree and assign codes to characters[8].

Then we apply AES algorithm for the encryption of 128 bit key which is sufficient to protect information at a secret level[9] and also easy to implement with a high speed and low RAM.

At the beginning of the message, immediately append after the text after the first line to separate the block of spaces, which is not possible unless the last 3 bits coded to zero spaces.

III. PSEUDO CODE OF THE PROPOSED ALGORITHM

The pseudo code is a detailed description also readable of what a computer program or algorithm has to do[10]. It helps the programmer in designing the model in any programming language. The pseudo code of our proposed model is shown in fig2, in annexure.

IV. CLASS DIAGRAM OF THE PROPOSED MODEL

Class diagram is a part of structural Unified Modeling Language (UML) which shows the relationship among the system's classes, objects and methods[11]. In our proposed model, we have used five classes: HuffmanEncoder, HuffmanTree, Node, AESencryption and StegoMain, which is shown in fig3, in annexure. The first three classes are used for the Huffman scheme, the class AESencryption is used for making the encryption applying AES and StegoMain for synchronizing the whole program and also for the making the whitespaces and tabs.

V. RESULTS

In this section, we will show some outputs of our proposed model in tabular form and discuss on those results.

Cover Text (size)	Secret Text (size)	Stego Text (size)	Increased by ((stego/(cover +secret))
105 bytes	18 bytes	3.85 kb	31.30
2.58 kb	18 bytes	77.9 kb	29.98
3.52 kb	18 bytes	100 kb	28.26
4.48 kb	43 bytes	126 kb	27.85
4.91 kb	32 bytes	142 kb	28.73
5.35 kb	42 bytes	156 kb	28.93
4.72 kb	51 bytes	136 kb	28.50

Table1: Results of proposed model

In this table (Table1), for the first three values, we have taken cover text files with various increasing sizes and keeping the secret text files fixed. From the result, we can observe that rate of increasing is lesser while the size of cover file is increasing, in this case if the hacker be able to reach the document, then also can't realize about the hidden text. In the next 4 cases, we have taken the sizes of cover text files and the secret text files randomly, where the result is also showing same scenario.

From the above two tables, we can see that the results of our proposed model are according to the expected values.

VI. CONCLUSION

Though the result of our proposed model is quite okay but the increasing of size always makes doubt in the viewer's mind. In case of image steganography the change of size is negligible. As we see the result is better with the greater size of cover image, if the administrator uses the study materials as the cover image, then it will be better. This model is also applicable for other online communication systems like e-commerce, e-governance to transmit text documents like ATM pin number or other text documents. If the sender sends the document along with digital signature, then the authentication purpose can also be fulfilled, which is beyond the scope of this paper.

REFERENCES:

[1] Weippl, R.E (2005), Security in E-Learning, Springer
[2] M.Kwan, "SNOW", Darkside Technologies Pty Ltd CAN 082 444 246 Australia, 1998
[3] S.Mansor, R.Din and A.Samsudin, "Analysis of natural language steganography", International journal of

computer science and security (IJCSS), vol:3(2), pp:113-125
[4] https://en.wikipedia.org/wiki/SNOW
[5] https://en.wikipedia.org/wiki/ICE_(cipher)
[6] https://en.wikipedia.org/wiki/Advanced_Encryption_Standard
[7] Andrew, S. Tanenbaum (2005), Computer Networks, Pearson Prentice Hall
[8] http://www.geeksforgeeks.org/greedy-algorithms-set-3-huffman-coding
[9] https://en.wikipedia.org/wiki/Advanced_Encryption_Standard
[10] http://whatis.techtarget.com/definition/pseudocode
[11] https://en.wikipedia.org/wiki/Class_diagram

ANNEXURE

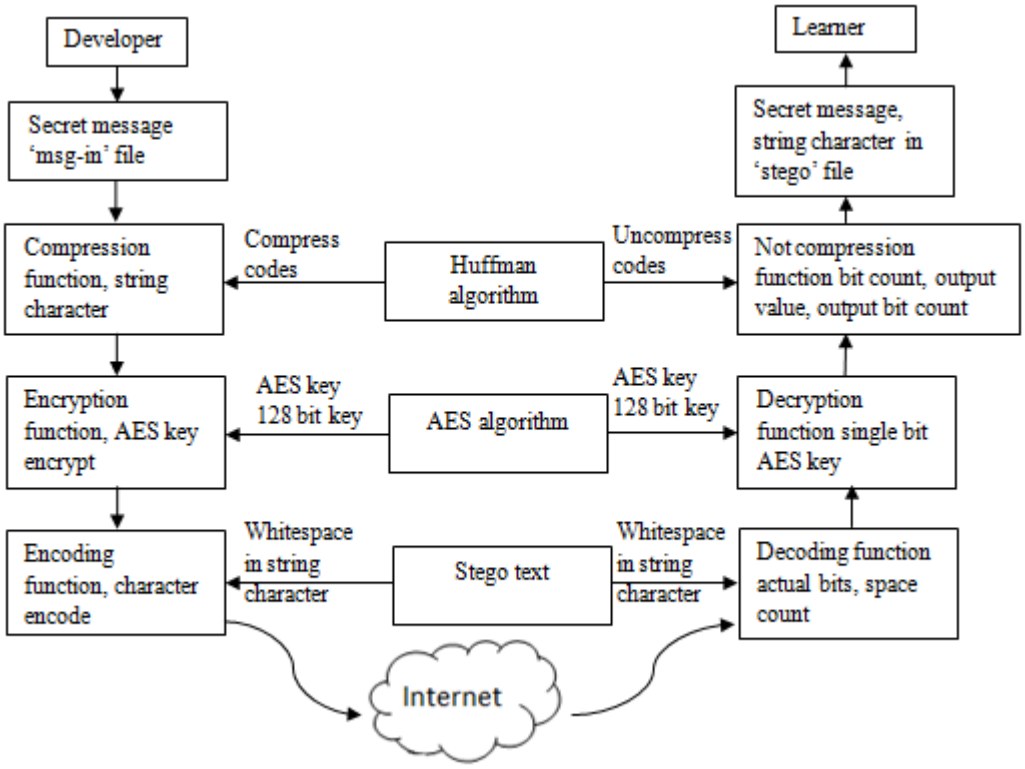

Fig1.The process flow of SNOW steganography tool in respect of hiding secret texts

Algorithm 1. The text-steganography encoding algorithm

```
function ENCODE(MessageFile, CoverFile, OutputStegoFile)
    CoverText ← ReadFile(CoverFile)
    Message ← ReadFile(MessageFile)
    Frequencies[ ] ← ComputeCharacterFrequencies(CoverText)
    HuffmanTree ← GenerateHuffmanTree(Frequencies)
    CharacterCodes[ ] ← GetCharacterCodes(HuffmanTree)
    Temp ← EncodeText(Message, CharacterCodes)
    HuffmanEncodedText ← PadTextWithZeroes(Temp)
    PadCount ← Length(HuffmanEncodedText) – Length(Temp)
    AESKey ← GenerateAESKey(128)
    EncryptedText ← EncryptAES(HuffmanEncodedText, AESKey)
    LineCount ← CountLines(CoverText)
    Share ← Length(EncryptedText) / LineCount
    WhitespaceText ← EncodeAsWhitespace(EncryptedText)
    WTL ← Length(WhitespaceText)
    PadCountText ← EncodeAsWhitespace(ToBinary(PadCount,3))
    j ← 0
    for i=1 to LineCount
        Write LineAt(CoverText, i) to OutputStegoFile
```

 ISSN: 2454-4248
234 – 237

```
                for k ← 1 to Share
                        if (k+j ≤ WTL) then Write CharAt(WhitespaceText, k+j) to OutputStegoFile
                end for
                j ← j + Share
        end for
        Write PadCountText to OutputStegoFile
end function
```

Fig2.Pseudo code of the proposed model

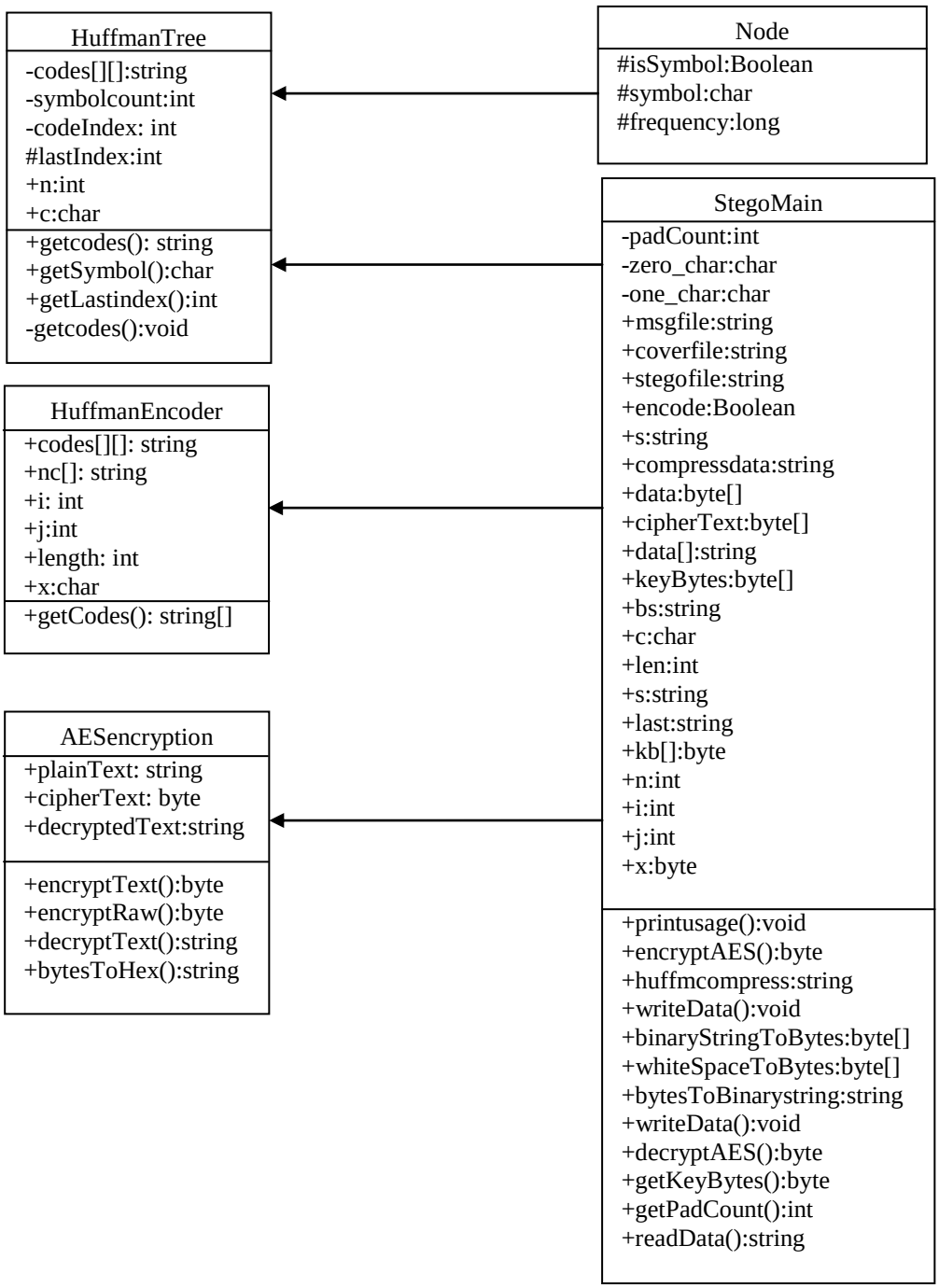

Fig3: Class diagram of the proposed model