Text to Text Embedding Approach for Information Security System

Hsint Hsint Htay[1], Kaythi Aung San[2], Phyu Phyu Htun[3] , Chaw Kalyar Than[4], Saw Zaw Lin[5], Kyaw Zin Htun[6], Aung Myint Aye[7], Moh Moh Aung[8]

[#]all authors and co-authors are from University of Computer Studies, Loikaw

[1]hsinthsinthhh@gmail.com, [2]kaythi.tgi@gmail.com
[3]phyuphyuhtun24@gmail.com, [4]chawkalyarthan07@gmail.com
[5]sawzawlin.johnsaw@gmail.com, [6]Skb083@gmail.com
[7]Dr.aungmyintaye@gmail.com, [8]moemoea2009@gmail.com

***Abstract*— This paper presents a new text to text data hiding steganography using arbitary two bits including least significant bits. It also presents a novel algorithm to hide a large amount of text in cover text file without affecting the cover format, meaning and font types, by using position file in which all indexes are stored according to the secret message with respect to the cover text file. The size of secret message and cover text file can vary the processing time, while extracting secret message back from cover files using position file. Unlimited amount of secret message can be hidden in a cover text file because the indexes are used as much time as we want with round loop function. But the more the size of secret message, the more the size of position file. In this research work, various secret message and cover files are used to analysis of extracting time. This proposed system is implemented by using MatLab Programming Language.**

***Keywords*— embedding system, steganography, information security, least significant bit, secret message.**

I. INTRODUCTION

With the rapid development of Internet, safe covert communications in the network environment become an essential research direction. Steganography is a significant means that secret information is embedded into cover data imperceptibly for transmission, so that information cannot be easily aware by others.

In today's world, keeping the digital information safe from being misused is one of the most important criteria [1].

Steganography is also the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security without knowledge of its existence. Our major need is hiding [9].

In steganography the original message is not modified but the existence of message is hidden in the selected medium by embedding techniques [10].

In order to embed messages, a cover text must provide information carriers that can be modified to represent the secret [6].

II. RELATED STEGANOGRAPHY THEORY

Steganography comes from the Greek Words: **steganos** – "Covered", **graphie** – "Writing". Generally the sender writes an innocuous message and then conceals a secret message on the same piece of paper. The main goal of steganography is to communicate securely in a completely undetectable manner and to avoid drawing suspicion to the transmission of a hidden data. It is not to keep others from knowing the hidden information, but it is to keep others from thinking that the information even exists. The data can be hidden in basic formats like: Audio, Video, Text and Images etc. The various types of steganography include:

A. Image Steganography: In image steganography, the data is hidden in a cover image. Image contain a lot of redundant information in which the data or the message can be embedded efficiently. The conventional image steganography algorithm is LSB embedding algorithm.

B. Audio Steganography: In audio steganography data is hidden by modifying the audio signal so that the changes cannot be easily intercepted by unauthorized personnel.

C. Video Steganography: Video steganography hides the data in a video. The pixel changes in the repective frames of videos are harder to detect than image steganography.

D. Text Steganography: Text Steganography hides data behind a cover text file [5].

There are several types of text steganography: structural, random, statistical generation and linguistic. Structural text steganography contains replace the physical structure of the text, for example by insert whitespace or increasing the line spacing. Random and statistical generation contains generating the covertext either randomly, or according to some algorithms. Linguistic contains process the syntactic or semantic specification of the existing text [8].

Text steganography can consist of anything from edit the formatting of an existing text, like replacing word within text, to generating random character sequences or using context-free grammars to generate readable texts. Fig. 1 illustrates text steganography idea. Firstly, a secret message will be hiding in a cover-text by applying an embedding algorithm to produce a stego-text and position file. After that the stegotext will transfer by communication channel [8].

III. PROPOSED SYSTEM AND ALGORITHM

The block diagrams of proposed system are shown in Fig. 1.

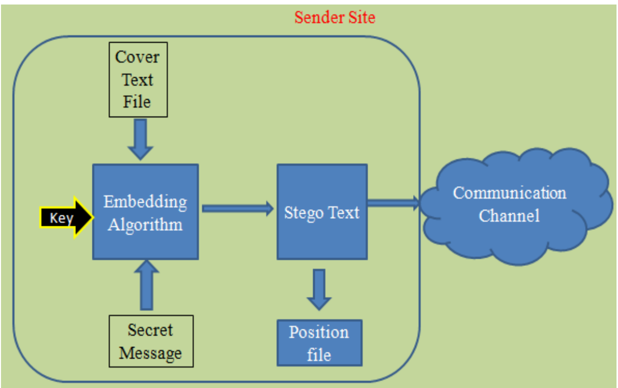

Figure1 (a) Sender site

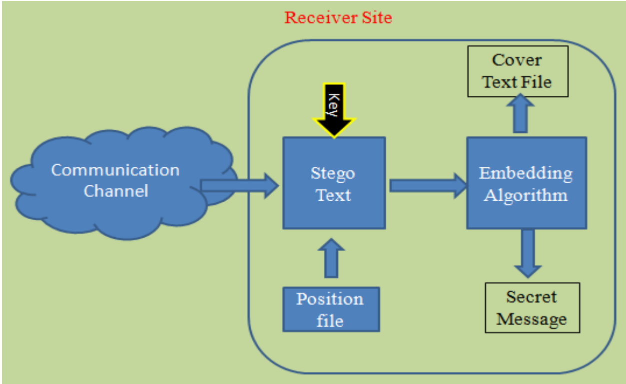

Figure 1 (b) Receiver site

Before any message exchange can take place, the sender and receiver must share embedding algorithm and public keys as the input 2 bits key, which is used in data hiding process at sender site.

A. Text Steganography of proposed system

In order to hide secret message, a cover text must provide surfficient amount of characters that can be modified to represent the secret. The cover text file can contain any text, number and characters. To do so, all contents of cover text file are converted into binary bits (each character must be in 8 bits binary form).

In this case, the 2 desired bits as the keys are used to point out the position of 2 bits sequence (‘00’, ‘01’, ‘10’, ‘11’). The first key bit must be greater than 1 and less than 9, and the second key must be from 2 to 8 values that user can choose, as each character of cover text files have been converted into 8 bits binary form. The following table can show the detail transformation of proposed steganography.

TABLE I

TEXT AND BINARY BITS EXPLANATION

Character from a cover file	ASCI I value	Binary 8 bits value	Remark
A	65	010000**01**	For 7 and 8 bits of input key values.
B	66	010000**10**	
C	67	010000**11**	
D	68	010001**00**	
E	69	010001**01**	
F	70	010001**10**	

The explanations example of changing processes from the content of cover text file and secret message are shown in Table 1. In this table, the public 2 bits key as 7 and 8 values are used and the 'Binary 8 bits value' column shows the index values (also called position values) of cover text file. For the content of secret message, all the 2 bits stream of it are used to find out the position in the desired 2 bits stream of cover text file.

B. Embedding Process (sender site)

The proposed embedding algorithm is very simple to embed and extract. In this research work, nothing is changed in the cover text file. But the positions of secret message are stored in separated position text file. In this research work, each 2 bits stream of secret message was embedded in desired 2 bits (such as 2 key bits position) of cover text file until the end of secret message bit stream.

- Step (1): Choose the cover text file;
- Step (2): Convert the content of cover text file into binary from ASCII codes;
- Step (3): Point out the 2 bits location according to the key positions; (in this case, the key postions values are represented as the bits to be hidden)
- Step (4): Accept the secret message;
- Step (5): Convert the input secret message into binary values from ASCII codes;
- Step (6): Read each 2 bits from the secret message and create position file by looking forward desired consecutive 2 bits of the cover text file according to the input 2 key bits;
- Step (7): go to step (6) until the end of secret message bits.
- Step (8): Finally, the output position file is saved in desired folder.

C. Extracting Process (receiver site)

As the embedding algorithm is very simple and useful to embed, the extracting processes at the receiver site also are simple and effective.

- Step (1): Choose the cover text file;
- Step (2): Choose the position file that have been sent from the sender site
- Step (3): Load 2 bits key as the public keys;
- Step (4): Extract binary bit stream from the cover text files according to values of the position file
- Step (5): The result binary bit streams are transformed into character format;
- Step (6): do step (4) until the end of values of position file;

IV. SYSTEM IMPLEMENTATION

The proposed system is implemented using MatLab Programming Language. The main proposed system is shown in the following Fig. 2. There are two portions, the first one is sender site and second one is receiver site. The user should choose cover text file from desired folder by clicking browse button. After that, the contents of cover text file have to be converted into binary form.

By accepting and converting secret message into binary bit stream which is entered from the keyboard, the desired 2 keys bit are used to find out postions of the digital bit stream (such as '00', '01', '10', '11') from the contents of cover text file. Finally the desired output result position file is saved in a folder.

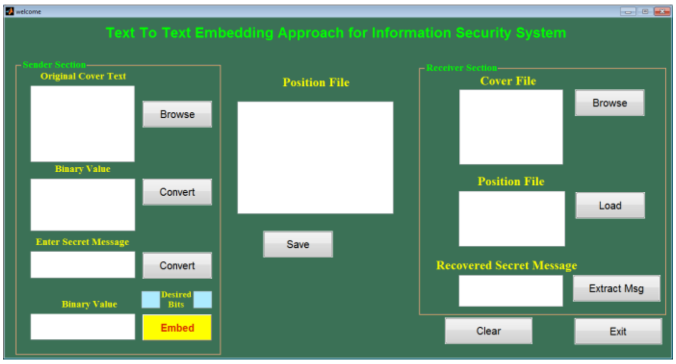

Figure 2 Main interface of proposed system

In the receiver site, the user should load the cover text file and position file. After that, public keys (2 desired keys bits) are used to extract secret message from the cover text file according to the position file.

V. EXPERIMENTAL RESULTS

Although, the arbitrary size of secret message can be hidden in a cover text file, the bit stream of cover text file must contain desired four pairs of bit stream (such as ‘00’, ‘01’, ‘10’, ‘11’). If there are such bit streams in the cover text file, the desired arbitrary size of secret message can be hidden in it.

The size of secret message can affect the size of position file. If the user wants to get secret message correctly, the chosen cover text file, position file and extracting algorithms are chosen definitely.

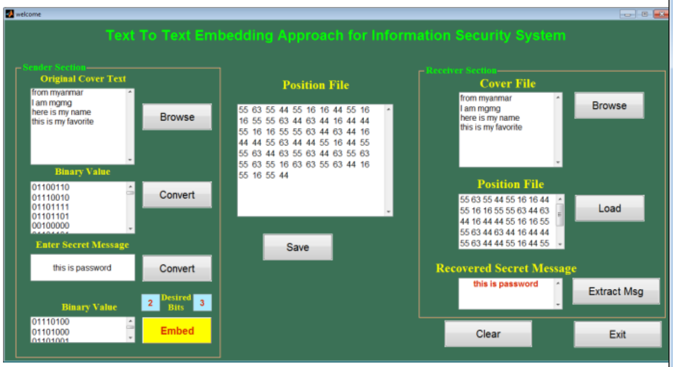

Figure 3 Main interface of proposed system

The Figure 3 shows the entire process of embedding and extracting secret message to/from cover text file.

The following Figure 4 (a) and (b) show how to accept secret message and public 2 bits key from the sender site and extracting of secret message using position from the cover text file. In this process public 2 bits key of 7 and 8 values are used.

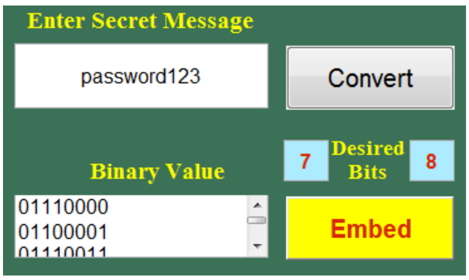

Figure 4(a) Accepting secret message and public 2 bits key at the sender site

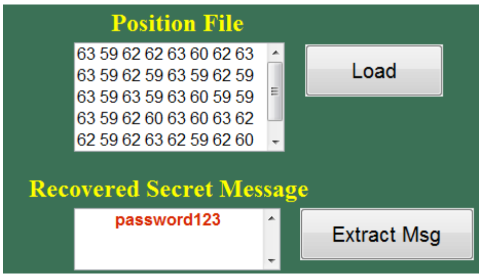

Figure 4(b) Extracting Secret message using position file at the receiver site

The Figure 5 illustartes the analysis of the extracting time with different secret message sizes and cover text files.

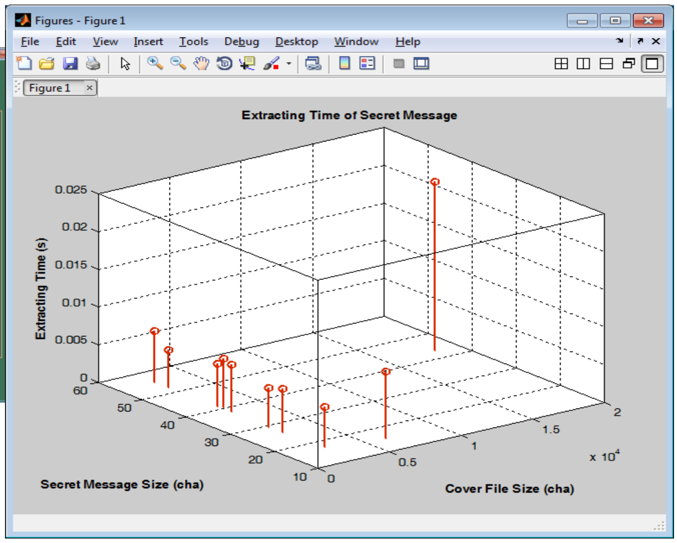

Figure 5 Main interface of proposed system

VI. CONCLUSIONS

The proposed technique is considered to be an efficient method for concealing text in a cover text files such that information can reach the destination in a safe way without being modified [10].

The presented research work expresses the new text to text data hiding approach for information security system. According to the analysis results, the arbitrary size of secret message can be hidden in a cover text file. As the proposed system used each 2 bits of secret message sequence, the cover text file must have desired 2 bits sequences (called ‘00’, ‘01’, ‘10’ and ‘11’). Otherwise, the cover text file that has no any one of such sequence, cannot be used as a cover file.

It is clear that the size of secret message and the size of cover text file do not depend on each other, but the size of secret message which can be hidden, depends on the content of cover text file.

The fluctuation of extracting time can be sometime faced the unusual rates, as the processing time of CPU usage also depends on the size of RAM, type of OS, the numbers of running applications at this time and so on.

VII. FURTHER EXTENSION

As the proposed system used the text cover file, the follower researchers should conduct another cover file types. Also it needs to conduct text to text data hiding system using different file sizes, different file formats, different methods. Any encryption method with text steganography should be applied in this system to improve more secure system. As the steganography object from the sender site is equally the same as the cover text file, the output of sender site (such as stegno object) should be transformed into other file format or file types.

ACKNOWLEDGMENT

Dr. Thinn Thu Naing, the Pro-rector of University of Computer Studies, Taunggyi, is being thanked for giving them a chance to participate in their Research Conference and Product Show 2017. The authors and co-authors would like to express their thanks to their group leader, Dr. Aung Myint Aye, Professor and Principal of University of Computer Studies, Loikaw for his valuable suggestion. They also like to present their thanks to machine authorizer, U Saw Zaw Lin, who helped whenever they want to run and conduct their experiment according to their research and for his participating in this research. Finally the authors would like to express their gratitude to all their colleagues for their valuable suggestions, comments and advice in preparation of the research work.

REFERENCES

[1] S. T. Abaas, *Improve Capacity in Text in Text Steganography,* Education College, University of Kufa, Iraq, European Aademic Research, Vol. II, Issue 12/marh 2015.

[2] A. K. Signh, *Steganography in Images using LSB Technique,* Amity University Gurgaon, India, International Journal of Latest Trends in Engineerning and Technology (IJLTET), Dec, 2015.

[3] S. Bhavana, Department of ECE, Bangalore, *Text Steganography using LSB Inserting Method Along with Chaos Theory,* International Journal of Computer Science, Engineering and Application (IJCSEA), Vol.2, No.2. April 2012.

[4] A. Hamarsheh, *Exploiting Omega Networks to Hide Text-in-Text Messages,* department of Computer Information Technology, the Arab American University, IJCSNS International Journal of Computer Science and Network Securiity, Vol. 15, No.5, May 2015.

[5] P. Tatwadarshi, *A Survey of Hindi Text Steganography,* International Journal of Scientific and Engineering Research, Vol. 7, Issue 3, March-16.

[6] C. Y. Chang, *the Secret's in the Word Order: Text-to-Text Generation for Linguistic Steganograpy,* University of Cambridge, UK, Proceeedings of COLING 2012: technical Papers, pages 511-528, December 2012.

[7] P. M. Nishigandha, Prof. S. S. Dhaopte, *Data Hiding Using Steganography,* International Journal of Science and Research (IJSR), Information Technology Department, Institute of Technology and Research, Amravati, India Vol.3, Issue 11, Nov 2014.

[8] Ass. Prof. Dr. Suhad M Kadhem, *Proposed Arabic Text Steganography method based on New Coding Technique,* Computer Science Department/ University of Technoloogy, Baghdad, Iraq, International Journal of Engineering Research and Application, Vol.6, Issue 9, Sept 2016.

[9] S. Gupta, *Text-Steganography Review Study and Comparative Analysis,* TITS Bhiwani, Interntional Journal of Computer Science and Information Technologies, Vol.2(5), 2011.

[10] K. U Singh, *LSB Audio Steganography Approach,* Department of Computer Science, Faculty of Science Banaras Hindu University, Varanasi, India, International Journal of Emerging Technology and Advanced Engineering, Vol. 4, Issue 4, April 2014.

[11] B. Lavanya, *Data Hiding in Audio by using Image Steganography Technique,* Vignan Bharathi Institute of Technology, Hyderabad, International Journal of Emering Trends and Technology in Computer Science (IJETTCS), Vol.2, Issue 6, Nov 2013.