AN APPROACH TO REDUCE THE SIZE OF SECRET TEXT AND A SECURE TEXT STEGANOGRAPHY

[1]VISHAL RANJAN, [2]SHAILJA BAGDI

Department of Computer Science and Software Engineering, SRM University, Chennai

Abstract - Steganography is an art on which the data can be hide in other data as cover, the text files is the commonly used for hiding data. The main aspects of the steganography is the capacity and security, where the capacity refer to how much data can be hidden in the cover carrier, while the security ,where concern with the ability of disclose or altering the data by unauthorized party. The aim of this project is to implement an algorithm to reduce the size of objects created using steganography. In addition, the security level of each approach is made more secured. This project presents an overview of text steganography and various existing textbased steganography techniques. Highlighted are some of the problems inherent in text steganography as well as issues with existing solutions. A new approach is proposed in information hiding using inter- word spacing which reduces the amount of information to hide. This method offers generated stego-text of maximum capacity according to the length of the secret message. This project also analyzed the significant drawbacks of each existing method and how this new approach could be recommended as a solution.

Keywords - Capacity, Security, Text Steganography Stego Object, Data Hiding.

I. INTRODUCTION

In the field of Data Communication, security-issues have got the top priority. So, of late the degree of security provided by a security tool has become the main evolutionary criteria of it. Classical cryptography is one of the ways to secure plain text messages. Along with that at the time of data transmission, security is also implemented by introducing the concept of steganography, watermarking, etc. In this types of combined approach, there exits some drawbacks. In remote networking, at the time of transmission of hidden encrypted text message, if the eavesdroppers get the track of the hidden text, then they could easily get the encrypted text. Now breaking of encrypted text message can be achieved by applying some brute force technique. So, there remains some probability of snooping of information. So, this type of techniques incurs another level of security which can route the Cryptanalyzer or Steganalyzer in a different direction.

II. STEGANOGRAPHY

Steganography or Stego as it is often referred to in the IT community, literally means, "Covered writing" which is derived from the Greek language. Steganography is defined by Markus Kahn [2] as follows, "Steganography is the art and science of communicating in a way which hides the existence of the communication. In contrast to Cryptography, where the unauthorized party is allowed to detect, intercept and modify messages without being able to violate certain security premises guaranteed by a cryptosystem, the goal of Steganography is to hide messages inside other harmless messages in a way that does not allow any unauthorized party to even detect that there is a second message present".

Steganography can be used in a large amount of data formats in the digital world of today. The most popular data formats used are .bmp, .doc, .gif, .jpeg, .mp3, .txt and .wav.

III. EXISTING SYSTEM

PETER WAYNER'S MIMICRY ALGORITHM

P eter Wayner [5] proposed a mimicry algorithm that aimed at text in his book Mimic Functions, Cryptologia XVI-3. His approach is to produce mimicked text that looks similar to the real structure of the original text. Peter Wayner used a set of grammatical rules to generate stegotext and the choice of each word determines how secret message bits are encoded.

Merits:

- Use of grammar makes easy to debug, reduce the errors.
- Secret message can be encoded into something innocent looking, in a form of spam where nobody will notice there is a secret message being concealed.

Demerits:

- Complex logic
- The stego-object file is larger.

BRASSIL'S DOCUMENT CODING METHOD

B rassil et al. [7] gave the initial idea of document coding methods in his paper by proposing life-shift coding, word-shift coding and feature coding (character coding) to discourage illicit dissemination of document distributed by computer network.Line-shift coding is a method of altering a document by vertically shifting the locations of text lines to uniquely encode the document. Word-shift coding is a method to alter a document by horizontally shifting

the locations of words within text lines to uniquely encode the document. Character coding or feature specific coding is a coding method that is applied only to the bitmap image of the document and can be examined for chosen character features, and those features are altered, or not altered, depending on the codeword.A document is marked in an indiscernible way by a codeword identifying the registered owner to whom the document is sent. If a document copy is found that is suspected to have been illicitly disseminated, that copy can be decoded and the registered owner identified.]

Merits:

- Easy to implement
- Large number of characters can be embedded.

Demerits:

- This can be done in hard copy documents only.
- It can be observed easily.

IV. PROPOSED SYSTEM

In the proposed system i have started an overview of text steganography and various existing text-based steganography techniques. Highlighted are some of the problems inherent in text steganography as well as issues with existing solutions.

A new approach is proposed in information hiding using inter- word spacing which reduces the amount of information to hide. This method offers generated stegotext of maximum capacity according to the length of the secret message. This proposed system also analyzed the significant drawbacks of each existing method and how this new approach could be recommended as a solution.

A very secure text steganography is been implemented that is explained in two phases.

Phase I : Sender side process like encryption in cryptography.

Phase II: Receiver side process like decryption in cryptography.

SENDER SIDE MANIPULATION

In sender side the sender has to give the information going to hide and also the stego-key which is normally password. The following figure illustrates the overall concept of this proposal. The flow diagram shows the general blocks present in the flow of hiding process. The each block is explained in the following manner.

ALGORITHM

- Get the secret message file and stego-key from sender in order to hide the information.
- Find the size of the file. According to the size, generate the cover text file dynamically.
- Give the cover text file along with the password and secret information to Position Identifier which returns the position vector which contains the occurrence byte number in the cover text file of each character present in secret file.
- Give the position vector to Binary Convertor which returns the binary value of each element of the position vector.
- Then Manchester Encoding is performed for the sequence of bits.
- Finally those bits are hided in the cover text file using different hiding methods by Bits Hider.

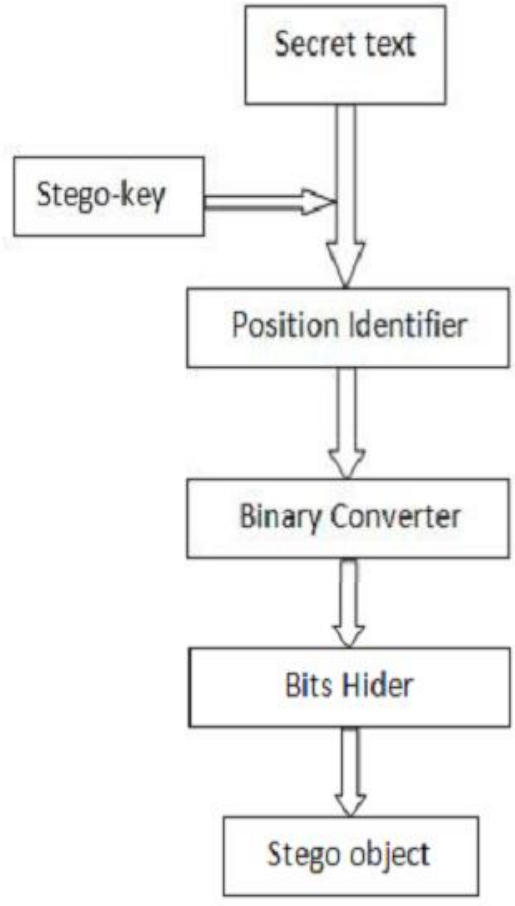

Fig.1.1 Sender Side Manipulation

RECEIVER SIDE MANIPULATION

T he receiver side process is opposite to sender side process. That is first Bits Extractor extracts the bits from stego object file. Then Decimal Converter converts the binary number to decimal number which is position vector values. It assembles the values and returns the position vector. Next Character Identifier identifies the character at the position value from

position vector. It returns sequence of character. Then the character assembler assembles the character as secret message.

ALGORITHM

- Read the Stego Object file character.
- Give these characters to Bits Extractor which identifies the bits and return the Bit Sequence. The process of Bits Extractor can be understood from the opposite process of Bits Hider in the sender side manipulation.
- Give this Bit Sequence to Decimal Converter which returns the Position vector consisting of all decimal numbers. The process of Decimal Converter can be understood from the opposite process of Binary Converter in the sender side manipulation.
- Give this position vector to Character Identifier which identifies the character at the positions in the position vector. The process of Character Identifier can be understood from the opposite process of Position Identifier.
- Finally Character Assembler gets the character from Character Identifier, and assembles them. It returns the Secret Text.

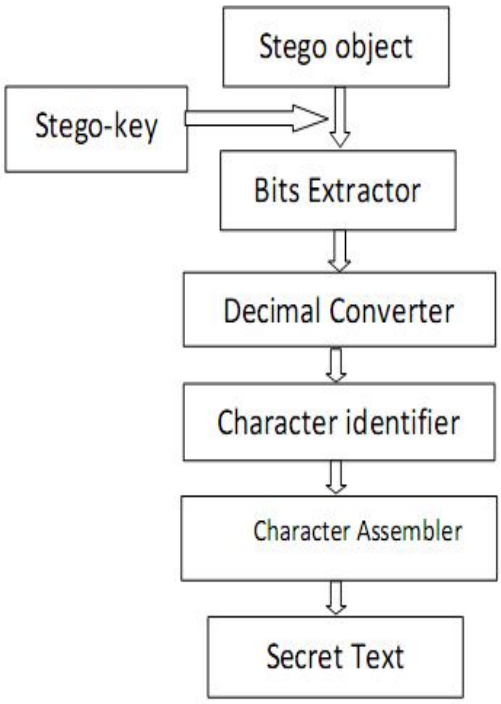

Fig.1.2 Receiver Side Manipulation

The experiment on various size of input file our method shows reduced usage of space to hide. This is explained as follows.

For example consider the following line of text we are going to hide: "I will come on Monday." This sentence has 22 characters. That is the size of the file is 22 bytes. In order to hide these characters by this proposed method the cover medium is taken as pangram sentences which have all 26 letters in English. So the position vector contain 22 entries where each one entry for one character. The value ranges from 0 to 32 because the first sentence itself contains all 26 letters and we are looking for first occurrence. So to hide the numbers 0 to 32 requires 5 bits. So for 22 numbers totally 110 bits needed to hide. So it consumes around 14 bytes in memory. By traditional schemes they are directly hiding the character in binary form so it requires (22 X 8= 176) 176 bits to hide. It consumes 22 bytes. So the amount of space saved is 8 bytes.

CAPACITY

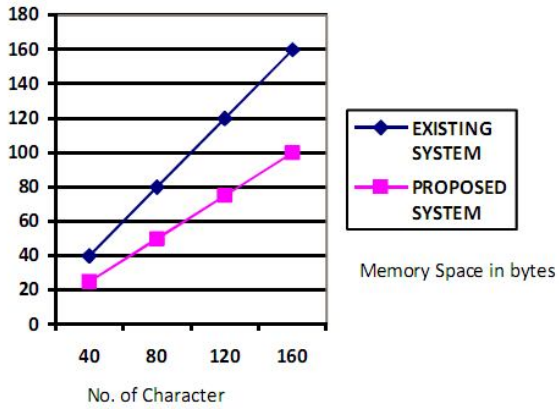

Fig.1.3 Existing System Vs Proposed System

SECURITY

Security issues mentioned in proposed system are better than that of existing system because five layers are used whereas in existing system only three layers are present as show in the Fig 1.4. Let us understand this with the help of an example. If cover file contains 66 characters then the minimum cover size will become 66! .This 66! can be converted into any of the six forms such as decimal, octal, binary, hexadecimal etc. Using permutation no. of ways to convert in any form stated above is (66!)6. Again in the next layer, position vector can be converted into binary, hexadecimal, octal and any other forms now no. of ways to convert in any form is $((66!)^6)^6$. We are converting 0's to 10 and 1's to 01 but attacker is unaware how 0's and 1's are converted thus there will be 'n' no of ways from attacker's point of view hence permutation will be $(((66!)^6)^6)^n$ after that 0 is converted into one space and 1 is converted to two space again attacker is unaware of this so total no of permutation is $((((66!)^6)^6)^n)^n$ and thus obfuscation can occur.

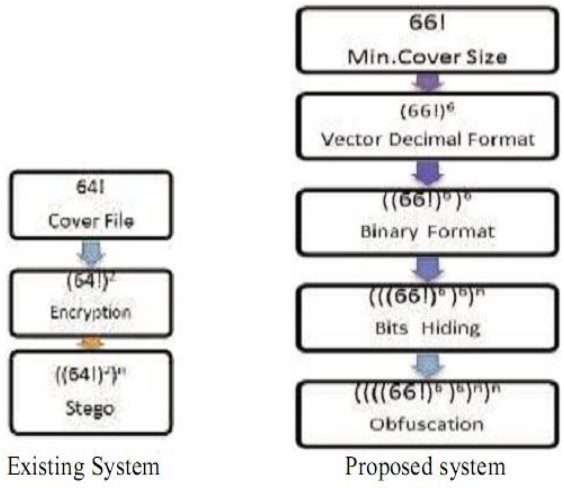

Fig.1.4 Existing System Vs Proposed System

CONCLUSION

This is a new approach of An Approach to reduce the size of stego object method using Text file as cover medium for hiding information. The unique feature about the method is to generate a cover-text according to the length of the secret message and information hiding using inter- word spacing which reduces the amount of information to hide. Experiments on the project work shows the better performance comparing to other method.

FUTURE ENHANCEMENTS

The future work should be focused towards optimizing the robustness of the decoding algorithm. This is because the hidden data will be destroyed once the spaces are deleted by some word processing software. Besides that, it is important to improve the capacity of the embedded scheme by taking other compression method into consideration.
The following changes can be made for future enhancement:

- If the secret information is medium the capable size image is taken as the cover medium.
- If the secret information is large the audio or video is taken as the cover medium depending upon the relative size of information.

REFERENCES

[1] L.Y.POR.B.Delina," Information Hiding: A new Approach in Text Steganography",7th WSEAS Int. Conf. on APPLIED COMPUTER &APPLIED COMPUTATIONAL SCIENCE '08, Hangzhou, china, April 6-8,2008

[2] Johnson, Neil F., "Steganography", URL:http://www.jjtc.com/stegdoc/index2.html

[3] M. Chapman, G. Davida, and M. Rennhard, "A Practical and Effective Approach to Large-Scale Automated Linguistic Steganography", Proceedings of the Information Security Conference,October 2001, pp. 156-165.

[4] K. Bennett, "Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text", Purdue University, CERIAS Tech. Report, 2004.

[5] P. Wayner, Disappearing Cryptography: Being and Nothingness on the Net, Academic Press, Inc., 1996.

[6] "Spammimic", 2000. [Online] Available: http://www.spammimic.com [Accessed July 1 2009]

[7] J. Brassil, S. Low, N. F. Maxemchuk, and L. O'Garman. "Electronic Marking and Identification Techniques to Discourage Document Copying". IEEE Journal on Selected Areas In Communications, Vol. 13, Oct. 1995, pp. 1495-1504.

[8] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", IBM syst. J., vol. 35, nos. 3 – 4, 1996, , pp. 313 – 336.

[9] N. Provos, P. Honeyman, "Hide and Seek: An Introduction to Steganography", The IEEE Computer Security, 2003.

[10] Dr. Mohammed Al-Mualla and Prof. Hussain Al-Ahmad, "Information Hiding:Steganography and Watermarking". [Online].Available: http://www.emirates.org/ieee/information_hiding.pdf [Accessed: July 01, 2011].

★★★