Int. J. Advanced Networking and Applications 1690
Volume: 04 Issue: 04 Pages:1690-1694 (2013) ISSN : 0975-0290

Steganography: A Security Model for Open Communication

K. Chandra Sekhar,
Department of Computer Applications,Madanapalle Institute of Technology and Science – 517325, AP,
Email: chandramcame@gmail.com
M.Chandra Sekhar,
Department of Computer Science & Technology, Sri Krishnadevaraya University, Anantapur, AP, India.
Email: chandragvp@gmail.com.
Mr. K.Chokkanathan,
Madanapalle Institute of Technology & Science, Madanapalle – 517325, AP, India
Sri Krishnadevaraya University, Anantapur, AP, India.

----------ABSTRACT----------

Steganography is the art and science of hiding messages. Steganography is basically about embedding a secret message in an innocuous wrapper; to communicate privately in an open channel. Steganography is often combined with cryptography so that even if the message is discovered it cannot be read. The key difference between Cryptography and Steganography lies with the fact that when a message is encrypted, third party knows that the message exists and it is encrypted, whereas with Steganography third party might not even know (or observe) that there does exists a message too, simply because its hidden behind an image, video or an audio etc which without keen observation might go without any suspicion. In the proposed system Steganography involves the method of acquiring insignificant bits in the audio file. After acquiring the insignificant bits in the audio header, replace them with significant bits (covert message) of information hiding[6]. Video file consists of both audio and images so the above two methods can be combined implemented with this.

Keyword: Redundant Bits, Stego-Medium, Embed, Carrier File Data hiding, Stegonagraphy , LSB

Date of Submission: December 13, 2012 Date of Acceptance: January 22, 2013

1.0 Introduction

Steganography Techniques

1. Physical steganography
2. Digital steganography
3. Network steganography
4. Printed steganography
5. Steganography using Sudoku Puzzle

1.1 Physical Steganography

Data hiding [1] within wax tablets: in ancient Greece, people wrote messages on the wood, and then covered it with wax upon which an innocent covering message was written. Hidden messages on messenger's body: also used in ancient Greece. The story of a message tattooed on a slave's shaved head, hidden by the growth of his hair, and exposed by shaving his head again. This method has obvious drawbacks, such as delayed transmission while waiting for the slave's hair to grow, and the restrictions on the number and size of messages that can be encoded on one person's scalp. Messages written on the back of postage stamps

In WWII, the French Resistance sent some messages written on the backs of couriers using invisible ink.

Hidden messages on paper written in secret inks, under other messages or on the blank parts of other messages.

1.2 .Digital steganography

Concealing messages within the lowest bits of noisy images or sound files. Concealing data within encrypted data or within random data. The data to be concealed is first encrypted before being used to overwrite part of a much larger block of encrypted data or a block of random data Mimic functions convert one file to have the statistical profile of another.

Concealed messages in tampered executable files, exploiting redundancy in the i386 instruction set. Digital water marking [3] is also one of the techniques.

Pictures embedded in video material which is called video steganography [4] (optionally played at slower or faster speed).Injecting imperceptible delays to packets sent over the network from the keyboard. Delays in key presses in some applications (telnet or remote desktop software) can mean a delay in packets, and the delays in

Volume: 04 Issue: 04 Pages:5 (2013) ISSN : 0975-0290

the packets can be used to encode data. Content-Aware Steganography hides information in the semantics a human user assigns to a datagram. These systems offer security against a non-human adversary/warden.

1.3 Network steganography

Contrary to the typical steganographic methods which utilize digital media (images, audio and video files) as a cover for hidden data, network steganography utilizes communication protocols' control elements and their basic intrinsic functionality. As a result, such methods are harder to detect and eliminate.

Modification of the properties of a single network protocol. Such modification can be applied to the PDU (Protocol Data Unit), to the time relations between the exchanged PDUs,[or both (hybrid methods).

The concealment of messages in Voice-over-IP conversations, e.g. the employment of delayed or corrupted packets that would normally be ignored by the receiver (this method is called LACK - Lost Audio Packets Steganography), or, alternatively, hiding information in unused header fields.WLAN Steganography – the utilization of methods that may be exercised to transmit steganograms in Wireless Local Area Networks)

1.4 Printed steganography

Digital steganography output may be in the form of printed documents. A message, the plaintext, may be first encrypted by traditional means, producing a cipher text. Then, an innocuous cover text is modified in some way so as to contain the cipher text, resulting in the stegotext. For example, the letter size, spacing, typeface, or other characteristics of a cover text can be manipulated to carry the hidden message. Only a recipient who knows the technique used can recover the message and then decrypt it. Francis Bacon developed Bacon's cipher as such a technique.

The cipher text produced by most digital steganography methods, however, is not printable. Printing introduces much noise in the cipher text, generally rendering the message unrecoverable. There are techniques that address this limitation, one notable example is ASCII Art Steganography.

2.0 Steganography Methods

According to modification in covers, the methods can be categorized as

1. Substitution
2. Transform domain
3. Spread spectrum
4. Statistical
5. Distortion

3.0 Images Using Lsb Based Chromatic Steganography Image Steg

Redundancy is one of the major aspects of creation. A close inspection reveals that redundancy does exist. For e.g. an image on a computer is represented by tons and tons of pixels, which in turn have many redundant information's. The simplest technique here is to fabricate the redundant bits so as to do the covert communication. For e.g. each pixel of an image consists of a variation of all three primary colors, red, green and blue, in a standard 24-bit bitmap, requiring 8 bits each for these three colors. i.e. there are 256 different variations, ranging from 00000000 to 11111111, for each colour in a pixel. So, to represent the colour white, the code would look like 11111111 11111111 11111111. Keeping in mind that, the human eye cannot distinguish the difference between too many colours, the colour 11111110 11111110 11111110 would look exactly the same as white, which means that the last digit in every bit in every pixel could be changed without much visual degradation of quality. This is the basis of the Least Significant Bit Insertion technique. It requires 8 bits to represent an ASCII text and there are three potential slots extra in every pixel of a picture. Therefore, in a conducive environment, with every three pixels, one ASCII text could be concealed. In order to make this practical to the user, a computer program would be needed. After typing in the secret message and determining a suitable cover message, the program would go through every pixel to find the potential candidate pixels and will change the least significant bit to represent each bit of the message. The image could then be sent to the recipient who in turn runs his program to take off the least significant bits to form the secret message. The existing system works with windows bit map image file format with loss less compression in to consideration. The algorithm would require secret message (M), a wrapper (W) and a pseudorandom seed (S) as input. In Windows bit map format, every image will have three separate colour channels; a channel dedicated for the red component (rCom), another one for the green component (gCom), and a third one for the blue component (bCom). After separating the colour channels, the program would go through each pixel to find all those pixels where the value of the rCom and gCom is equal to that of the supplied R and G values. Spatial details of every such pixel will be stored in an array named Candidate Pixel (CP) and the total numbers of such potential candidate pixels are calculated. If the length of the message (in bits) is more than the length of CP then a message will be displayed prompting the unsuitability of the wrapper under consideration.

4.0 Steganography in Audio Files

In proposed system Steganography audio [8] involves the method of acquiring insignificant bits in the audio file. Then acquire the insignificant bits in the audio header and replace them with significant bits of data.

When performing data hiding on audio that is in digital media[7], first the data is encrypted by password based

Volume: 04 Issue: 04 Pages:5 (2013) ISSN : 0975-0290

encryption using DES algorithm to generate the cipher text. Now the cipher text is kept hidden in the audio file using low bit encoding method. When extracting the data from audio first cipher text is separated from audio then the plain text is generated by decrypting the cipher text.

Audio steganography[8] has to satisfy three basic requirements. Perceptual transparency, capacity of hidden data and robustness.

There are two types of attacks to steganography and therefore there are two type of robustness. One type of attacks tries to reveal the hidden message and another type tries to destroy the hidden message. Substitution techniques are vulnerable against both types of attacks. The adversary who tries to reveal the hidden message must understand which bits are modified. Since substitution techniques usually modify the bits of lower layers in the samples -LSBs, it is easy to reveal the hidden message if the low transparency causes suspicious. Also, these attacks can be categorized in another way: Intentional attacks and unintentional attacks. Unintentional attacks like transition distortions could destroy the hidden message if is embedded in the bits of lower layers in the samples -LSBs. substitution techniques of audio steganography:

• Having low robustness against attacks which try to reveal the hidden message

• Having low robustness against distortions with high average power

Genetic Algorithm For Audio Steganography:

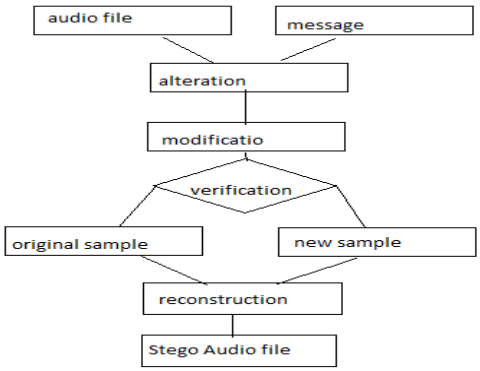

Fig 1. Approach Diagram

4.1 Alteration

At the first step, message bits substitute with the target bits of samples. Target bits are those bits which place at the layer that we want to alter. This is done by a simple substitution that does not need adjustability of result be measured.

4.2 Modification

In fact this step is the most important and essential part of algorithm. All results and achievements that we expect are depending on this step. Efficient and intelligent Algorithms are useful here. In this stage algorithm tries to decrease the amount of error and improve the transparency. For doing this stage, two different algorithms will be used.

One of them that are more simple likes to ordinary techniques, but in aspect of perspicacity will be more efficient to modify the bits of samples better. Since transparency is simply the difference between original sample and modified sample, with a more intelligent algorithm, I will try to modify and adjust more bits and samples than some previous algorithms. If we can decrease the difference of them, transparency will be improved. There are two example of adjusting for expected intelligent algorithm below. Sample bits are:

00101111 = 47
Target layer is 5, and message bit is 1
Without adjusting: 00111111 = 63 (difference is 16)
After adjusting: 00110000 = 48 (difference will be 1 for 1 bit embedding)
Sample bits are: 00100111 = 39
Target layers are 4&5, and message bits are 11
Without adjusting: 00111111 = 63 (difference is 24)
After adjusting: 00011111 = 31 (difference will be 8 for 2 bits embedding)

4.3 Verification

In fact this stage is quality controller. What the algorithm could do has been done, and now the outcome must be verified. If the difference between original sample and new sample is acceptable and reasonable, the new sample will be accepted; otherwise it will be rejected and original sample will be used in reconstructing the new audio file instead of that.

4.4 Reconstruction

The last step is new audio file (stego file) creation. This is done sample by sample. There are two states at the input of this step. Either modified sample is input or the original sample that is the same with host audio file. It is why we can claim the algorithm does not alter all samples or predictable samples. That means whether which sample will be used and modified is depending on the status of samples (Environment) and the decision of intelligent algorithm.

4.5 Distortion Analysis

Distortion analysis of stego images is carried out by studying distortion / similarity messages statistically. There are many methods for measuring distortion that can be used for distortion analysis. Distortion between two different images is measured by considering Mean Square Error (MSE), Mean Absolute Error (MAE) or Histogram Similarity (HS).

Volume: 04 Issue: 04 Pages:5 (2013) ISSN : 0975-0290

4.6 Depth Vs Distortion Analysis

Distortion occurred in different steganos is required by varying the depth of hiding for embedding information in stegno image[2]. The relation between depth of hiding used and distortion occurred in the stego images is shown in Fig 2. that depth of hiding within some LSB region is most suitable for message embedding as the distortion is very small in this region. As the depth of hiding increases beyond preferable region, the distortion becomes noticeable and unsuitable for message hiding.

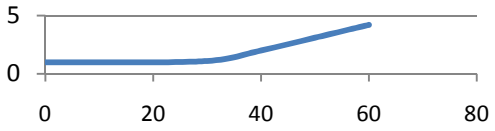

Fig 2. Depth Vs Distortion Analysis

To develop the algorithm multiple bits of each sample of the file have been changed or modified to insert text data in it. It has also been observed the degradation of the host audio file after modification of the bits. The bit modification was done by various ways, like 1, 2, 3, 4 bits were changed in turn. But after going through all the modification it has been observed that 1 bit change in LSB gave the best result.

The relation between depth of hiding used and distortion occurred in the sego images can be varied when by considering the No.of Bits form LSB position. If you consider only one bit in LSB position then the Distortion is very low, but if you choose two bits then the distortion will be not. So the distortion will be dependent on the no.of bits you are choosing from LSB position. The below Fig 3.represents the expected relation between Depth of Hiding and Distortion .

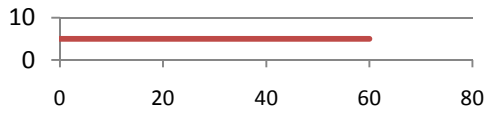

Fig 3. Depth Vs Distortion Analysis

5.0 Algorithm (For Embedding of Data):

1. Leave the header section of the audio file untouched...
2. Start from a suitable position of the data bytes. (For the experiment purpose the present start byte was the 51st byte). Edit the least significant bit with the data that have to be embedded.
3. Take every alternate sample and change the least significant bit to embed the whole message.
4. The data retrieving algorithm at the receiver's end follows
5. The same logic as the embedding algorithm.

Algorithm (For Extracting of Data):

1. Leave first 50 bytes.
2. Start from the 51st byte and store the least significant bit in a queue.
3. Check every alternate sample and store the least significant bit in the previous queue with a left shift of the previous bit.
4. Convert the binary values to decimal to get the ASCII values of the secret message.
5. From the ASCII find the secret message.

6.0 Experimentation and Results

First 44bytes samples are left because it is an Header..The data embedding with LSB modification is started after header section. If the data embedding process is started from 46th sample then the LSB value of the 46st sample should be modified. If the binary value of the corresponding sample is "01110100" then "1" should be modified. From Table I show the work.

Sample no	Binary value of Sample value	Binary value to be embedded	Binary value of Modification
46	0010 1110	1	0010 1111
48	0011 0000	0	0011 0000
50	0011 0010	1	0011 0011
52	0011 0100	0	0011 0100
54	0011 0110	1	0011 0111
56	0011 1000	1	0011 1001

Table I : Samples of Audio File with Binary Values Before And After Embedding

6.1 Retrieving algorithm has to be followed:

First, change the audio message into binary format that has come from the source as stego-object. Leave first 44 bytes with no change in them because it is an header data..Start from 46th bit, check the least significant bit, and store it in a queue. Check every alternate sample to collect the whole messages. Like 48th , 50th and 52nd and so on. Store the least significant bits of the alternate samples in the queue with left shift of previous bit. Convert the binary values to decimal to get back the ASCII from which the text can be retrieved. The whole retrieval process can be depicted with the following table more thoroughly:

Volume: 04 Issue: 04 Pages:5 (2013) ISSN : 0975-0290

Sample No.	Binary value with embedded secret data	Bits that are stored in data
46	0010 1111	1
48	0011 0000	10
50	0011 0011	101
52	0011 0100	1010
54	0011 0111	10101
56	0011 1001	101011

Table II

Extraction of data from audio file

7.0 Conclusion

A method of embedding text-based data into an audio file using the method of bit modification has been presented in this paper. A procedure is data field is edited to embed intended data into the audio file. To proceed with this, the header section of the audio has been checked perfectly because a minimal change in the header section may leads to a corruption of whole audio file.

In this algorithm, as an experiment first 44 bytes have been left untouched and starting from the 46th bytes every alternate sample has been modified to embed textual information. How the performance is affected by changing different bit fields has not been reported in this work. However a rough study was made to see how the changing of a specific bit field creates degradation in the host audio file and in which point it leads to perceptible change in the audible sound quality to any other third party other than the sender or receiver. It was noticed that changing the least significant bit of the bytes gave the best results. An audio file with size 952 KB has been used. The maximum text file size that can be embedded in this audio file without degrading the file structure can be traced through a survey.

REFERENCES

[1]. W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", IBM Systems Journal, vol. 35, Issues 3&4, 1996, pp. 313-336.

[2]. Kharrazi, M., Sencar, Husrev T., and Memon, N., "Image Steganography: Concepts and Practice", WSPC, April 22, 2004.

[3]. Stefan Katzenbeisser, Fabien A. P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking". Boston, Artech House, pp. 43 – 82. 2000.

[4]. K. Matsui and K. Tanaka. Video-steganography. In: IMA Intellectual Property Project Proceedings, volume 1, pp 187-206, 1994.

[5]. N.F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen," Computer, vol. 31, no. 2, pp. 26-34, IEEE, Feb. 1998.

[6]. Matsuoka, H., "Spread Spectrum Audio Steganography using Sub – band Phase Shifting", Proceedings of the 2006 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP'06), IEEE, 2006.

[7]. S.S. Agaian, D. Akopian, O. Caglayan, S. A. D'Souza, "Lossless Adaptive Digital Audio Steganography," In Proc. IEEE Int. Conf. Signals, Systems and Computers, pp. 903-906, November 2005.

[8]. K. Gopalan, "Audio steganography using bit modification", Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Vol. 2, pp. 421-424, April 2003.

[9]. Mohammad Pooyan, Ahmed Delforouzi, "LSB-based Audio Steganography Method Based on Lifting Wavelet Transform", International Symposium on Signal Processing and Information Technology, IEEE, 2007.