International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 01,January-2015

Implementation of GA-based Approach with Reduced RSA Encryption Security for Audio Steganography

Ms. Swati R. Shishupal, Mrs. Ujwala V. Gaikwad
MECOMP,
Terna Engineering College,
Nerul, Navi Mumbai

***Abstract* - This approach of an Audio Steganography transmits the hidden information by modifying an audio signal to ensure secured data transfer between parties.**

Reduced RSA encryption algorithm is used for encryption of the message. The encrypted message is encoded into audio data by Genetic Algorithm (GA) based Least Significant Bit approach. However, even knowledge of an existing hidden message should not be sufficient for the removal of the message without knowledge of additional parameters such as secret keys. In order to increase the robustness against intentional attacks and unintentional attacks the encrypted message bits are embedded into random LSB layers.

Here in order to reduce distortion, Genetic Algorithm is used. The basic idea behind this is to maintained uncertainty in message bit insertion into audio data for hiding the data from hackers and to provide a superior, well-organized method for hiding the data and sent to the destination in a safer manner.

Keywords – Audio Steganography; Genetic Algorithm (GA); Least Significant Bit (LSB); robustness, Reduced RSA

I. INTRODUCTION

Secure data transfer is one of the most important aspects in Internet to prevent data from various attacks, we use technologies like data preventing, data hiding, etc. So more robust methods are chosen so that they ensure secured data transfer.

Steganography is the art and skill to hide data in a cover media such as text, audio, image, video, etc. The term steganography in Greek words means, "Covered Writing" [1]. Steganography is the main part of the fast developing area of information hiding [2]. Steganography provides techniques for hiding the existence of a secondary message in the presence of a primary message. The main or primary message is referred to as the carrier signal or carrier message, the carrier signal can be text, audio, image, video, etc. The secondary message is referred to as the payload signal or payload message [3].

Various features influence the quality of audio stenographic methods. The significance and the impact of each feature depend on the application and the transmission environment. The mainly important properties include robustness to noise and to signal manipulation, security and hiding-capacity of embedded data. Robustness requirement is tightly related to the application and is the most challenging to satisfy in a stenographic system. In addition, there is a tradeoff between robustness and hiding-capacity. Generally, they barely coexist in the same steganographic system.

In most of the applications, the steganography algorithm has to insert additional data without affecting the perceptual quality of the audio host signal. The reliability of the steganography algorithm is usually defined as a perceptual similarity between the original and stego audio sequence. However, the value of the stego audio is usually corrupted, either intentionally by an adversary or unintentionally in the transmission process, before receiver perceives it. In that case, it is more adequate to define the reliability of a steganography algorithm as a perceptual similarity between the stego audio and the original host audio at the point at which they are presented to a consumer [4].

II. PROBLEM STATEMENT

This approach transmits the hidden information by modifying an audio signal to ensure secured data transfer between parties. Reduced RSA encryption algorithm is used for encryption of the message. The encrypted message is encoded into audio data by Genetic Algorithm (GA) based Least Significant Bit approach.

This approach increases the robustness against intentional attacks in which the hackers always strive to reveal the hidden message as well as some unintentional attacks such as noise addition etc; the encrypted message bits are embedded into random LSB layers [5]. GA operators are used to reduce distortion. To maintain randomness in message bit insertion into audio data is done for hiding the data from hackers and sent the message to the destination in an efficient way.

Reduced RSA algorithm is implemented using Mersenne Primes which guarantees the primality. This is an improved algorithm which increases the power of RSA by generating large prime numbers and also reduces the size of encrypted file.

III. PROPOSED SYSTEM

Different methods are already used to hide message into audio file, i.e., in Audio Steganography. At first, simple LSB, then modified LSB method were used [6]. Some of the authors tried to increase the LSB layer to increase the robustness against attack. It always increases the distortion in audio host file.

A. Outline of Proposed System:
Input: Audio File, Message File

Process Details- Encryption module:
The message is encrypted using Reduced RSA algorithm.

Encoding Module: The encrypted message is encoded into audio data by Genetic Algorithm (GA) based Least Significant Bit approach.

Output:
1. New audio file
2. Decrypted Message file
3. Reduced RSA file

B. Reduced RSA algorithm:

This algorithm which uses large prime numbers as a input parameter as follows:

Declarations:
a) Ptext.txt: Plain text (source) files to be encrypted.
b) Ctext.txt: Cipher text file to be decrypted.
c) Array L: Used to store the contents of Ptext.txt used for encryption.
d) Array S: Used to store the contents of Ctext.txt used for decryption.

Algorithm:
1. Choose two large prime numbers p, q.
2. Generate two very large mersenne prime numbers as: m = 2p -1 and n = 2q -1.
3. Calculate c= m* n.
4. Calculate the value of Φ using the formula: Φ(c) = (m-1) * (n-1).
5. Generate the public key 'e' such that it is co-prime with Φ(c).
6. Find the value of private key'd' such that (d *e) ≡ 1 mod Φ(c)
7. Read plaintext in the form of binary data from the file Ptext.txt, store it in an array (L).
8. Perform Le mod c (on each element of an array L) to get cipher text and store it in the file Ctext.txt.
9. For decryption, read the file Ctext.txt, store it in an array (S) .
10. Perform Sd mod c (on each element of an array S) to get a plain text.

Enhancement:
We perform enhancement in the above algorithm to reduce the size of encrypted file. So the reduction logic is given as follows:
For Encryption

1. Each element of array L is reduced modulo 'e' and the quotients are stored is an array (QUE) and store the remainders in array (REM).
2. Now array (REM) is a cipher text, supply it in the file ctext.txt.

For decryption
1. Read the file ctext.txt, stores it in an array (S).
2. Retrieve an array (QUE) and multiply each element of array (QUE) by e.
3. Now add each element of array (QUE) into array (S) respectively.
4. Perform Sd mod c to get a plain text.
In this enhancement, the public key e is used again for reduction. Dividing each array element of the cipher text again to get quotient and take mod e of each array element of the cipher text to get the remainder. The array of remainder is now reduced with great extent. Hence the size of each character (in digits) to be encoded get reduced which results the reduced file size. But at the phase of decryption, the values of quotients in array (QUE) are needed to retrieve to get the cipher text again. [7][8]

C. Genetic Algorithm Approach:
In genetic algorithms, a chromosome (also sometimes called a gene) is a set of parameters which defines a proposed solution to difficulty that the genetic algorithm is trying to resolve. The chromosome is often represented as a simple string; although a wide variety of other data structures are also used. Our possible solutions are the integers from 0 to 255, which can be represented as 8-digit binary strings. Thus, we may use an 8-digit binary string as chromosome. If a specified chromosome in the population represents the value 155, its chromosome is 10011011.

The simplest forms of genetic algorithm involve three types of operators: selection, crossover (single point) and mutation.
*Selection***:** This operator selects chromosomes from the population for reproduction of bit positions. The fitter the chromosome, the more times it is likely to be selected to reproduce.

*Crossover***:** This operator randomly chooses a locus and exchanges the subsequences before and after that locus between two chromosomes to create two offspring. For example, the bit strings 10000100 and 11111111 could be crossed over after the third locus in each to produce the two offspring 10011111 and 11100100.

*Mutation***:** This operator randomly flips some of the bits in a chromosome. For example, suppose the string 00000100 might be mutated in its second position to yield 0100010**.** Since transparency is simply the difference between original sample and modified sample. So, if we can decrease the difference of them, transparency will be improved. There is an example of adjusting for expected intelligent algorithm below.
Sample bits are: 001**0**1111 = 47
Target layer is 5, and message bit is 1
Without adjusting bit string: 001**1**1111 = 63
(Difference is 16)
After adjusting bit string: 001**1**0000 = 48
(Difference will be 1 for1 bit embedding)

Following are the steps involved in Genetic Algorithm:
Alteration

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

Modification
In fact this step is the most important and essential part of algorithm. All results and achievements that we look forward to are depending on this step. Well-organized and intelligent algorithms are useful here. In this stage algorithm try to decrease the amount of error and improve the transparency. For doing this stage, two different algorithms will be used. One of them is simpler like ordinary techniques, but in aspect of perspicacity will be more efficient to modify the bits of samples better. Since clearness is simply the difference between original sample and customized sample, with a more intelligent algorithm.

Verification
In fact this stage is quality controller. That is the algorithm could do has been done, and now the outcome must be confirmed. 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. Fig. 1 Genetic Algorithm Approach Diagram
Reconstruction

The last step is new audio file (stego file) creation. This is done sample by sample. There are two states for the input of this step. Either modified sample is input or the original sample that is the same with host audio file. That 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.

D. Advantages:

- This approach increases the robustness against intentional attacks in which the hackers always try to reveal the hidden message as well as some unintentional attacks such as noise addition, the encrypted message bits are fixed into random LSB layers.
- GA operators are used to reduce distortion.
- To maintain randomness in message bit insertion into audio data is done for hiding the data from hackers and sent the message to the destination in an well-organized way.

IV. PROPOSED METHODOLOGY

First, we encrypt text message using Reduced RSA encryption algorithm. And then applying proposed LSB algorithm, insert message bits to the audio bit stream (16 bit sample) in random and higher LSB layer positions (increase the robustness) to get a collection of chromosomes. Here now Genetic Algorithm operators are used to get the next generation chromosomes. Next select the best chromosome as per to the best fitness value. Fitness value is nothing but the value of LSB position for which we get a chromosome with the minimum deviation comparing to the original host audio sample. Here upper LSB layer is given higher preference in case of cover selection. We have original audio sample and inserting message bit in different LSB layer positions we get some new samples. Sometimes it can happen that for more than one LSB layer we get the same difference between original audio sample and new audio samples. In this case, we will select the higher LSB layer. In this, an intelligent algorithm is used to embed the message bits in the deeper layers of samples and alter other bits to decrease the error and if alteration is not possible for any sample it will pay no attention to them, which helps in achieving higher capacity which refers to the amount of information that a data hiding scheme can successfully embed without introducing perceptual distortion in the marked media and robustness which measures the ability of embedded data or watermark to withstand against intentional and unintentional attacks.

A. GA-based Approach with Reduced RSA Encryption Security for Audio Steganography Algorithm:

Embedding of message streams in an audio file:
Steps:

1. Convert Message data to byte streams.
2. Encryption of message is done by using Reduced RSA algorithm. The sender sign a message with public key exchange that is two sides cooperates to exchange message.
3. Read an audio file byte-wise. Convert to integer array (16 bits sample).
4. Generate n (2-16) number of chromosomes of 16 genes by inserting a Message bit into 16 bits audio sample in n (2-16) random positions.
5. Algorithm: To generate better next generation population, following GA operator based algorithm is used:

Let ‘position’ is the insertion position of the audio data,
If position = 1, then no action is taken
If position = 2 to 16
If 0 bit is to be embedded
If data bit is for position i
If data bit on i-1 position to 1 position are 0’s, perform crossover operation with 1……11 (i-1 no’s) bit string
If data bit on i+1 position is 0, perform mutation operation oni+1 position and crossover operation for i-1 to 1 with0…..00(i-1 no’s) bit string
If data bit on i+1 and i-1 to 0 are 1 no action is taken
If 0 bit is to be embedded
If data bit is 0 = no action is taken
If 1 bit is to be embedded
If data bit is 1 = no action is taken
If 1 bit is to be embedded
If data bit is 0 then
If layer is 1, then no action is taken
If layer = 2 to 16
If i-1 to 0 position are holding 1 crossover operation for i-1 to1 with 0…..00(i-1 no’s) bit string

If i+1 position holding 1 and i-1 to 1 position holding 0,perform mutation operation on i+1 position and crossover operation for i-1 to 1 with 1…..11(i-1 no's) bit string

6. Now select the best chromosome, where most excellent one is the chromosome which has the minimum difference with the original 16 bit audio sample
7. Here fitness value is the position number for which we get the best chromosome. Again, the position number, best chromosome and distortion are closely related, because whenever we will choose the best chromosome, which will reduce the distortion.
8. Fitness value are representing two things here
   i) Position number is very important at the receiving end to extract the message.
   ii) Distortion which again very important regarding security(distortion can convinced hacker to hack the message data).So, multi-objective GA is used here
9. Embedding the fitness value to the next audio data sample.
10. Writing stego-audio samples: Convert 16 bit stego audio sample to 8 bit bytes. And write stego audio byte stream to an audio file

Extracting hidden message:

At the time of inserting message data bits into audio sample, we optimize the difference between cover data and stego-data, so stego-audio is more or less equal to the original audio. So, to extract the hidden message data we need to know only the position number of hidden message bits.

Steps:

1. Read the stego audio file byte-wise. And convert to integer array (16 bits sample).
2. By getting the fitness value or location number of the hidden message data bit's into the stego audio sample, extract message data bit from the stego audio file.
3. To get 8 bits of the message data and random location number from audio data, choose 16 (16 bits) stego audio data.
4. Get the message byte streams for all the random positions.
5. Convert message byte stream to a data file
6. Write the extracted message data to a message file
7. Decryption of message is done by using Reduced RSA algorithm. The receiver signs a message with its private key and decrypts the message file.
8. Write the audio byte stream to an audio file by converting16 bit sample to 8 bits byte.

V. RESULTS AND DISCUSSIONS

A. *Results*

In this section we describe some experimental results, for RSA and Reduced RSA.

Table 5.1 RSA and Reduced RSA comparison table

Audio file size	Message file size	RSA file size	Reduced RSA file size
duck3	4 bytes	64 bytes	43 bytes
ALARME4	7bytes	112 ytes	92bytes
Mouth shut	9bytes	114bytes	96bytes
ALARME1	18bytes	288bytes	235bytes
Nuclear	21 bytes	336bytes	200bytes
Jump	25 bytes	400bytes	249bytes
Test1mb	32bytes	512bytes	400bytes

Here, In Table 5.1 we have compared the result of RSA encryption file size and Reduced RSA encryption file size. We have taken different wave audio file sizes and different message file sizes for encryption. Consider, mouth shut audio file which is size 487 kb and message file size is 9 bytes, so by applying simple RSA algorithm we got the encrypted file size as 114 bytes, and as per our proposed Reduced RSA algorithm we got the file size as 96 bytes.

B. *Performance Parameters*

To analyze the performance of steganography techniques three parameters are used [9]

After the extraction of hidden secret data, audio is reconstructed. Various performance measures such as MSE, PSNR and Embedding capacity has been evaluated. These parameters are used for audio quality analysis.

Peak Signal to Noise Ratio (PSNR) is used to measure quality of reconstruction of an object. It is the ratio between signal (Object) and noise (error). Generally higher PSNR indicates higher quality.

The Peak to Noise ratio provides the resemblance between the cover object, and the stego object, the file where the secret or private message has been hidden. It is defined through the Mean Square Error (MSE).

Mean Square Error (MSE) is used to measure the level of distortion (error) between original and stego audio file. It is calculated as in

$$MSE = \left(\frac{1}{mn}\right)\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$$

Where $I(i, j)$ is original audio file

$K(i, j)$ is stego audio file

$$PSNR = 10.log_{10}\left(\frac{MAX^2}{MSE}\right)$$

Where MAX is the maximum pixel value of an audio 255 for 8 bit message or image .The PSNR is nothing but the SNR when all pixel values are equal to the maximum possible value.

Typical values for the PSNR in images and video compression are between 30 and 50 dB, provided the bit depth is 8 bits, where higher is better. For 16-bit data typical values for the PSNR are between 60 and 80 dB. [10][11]

Table 5.2 GA's PSNR comparison of RSA and reduced RSA (db)

Audio file size	Message size	PSNR (GA)	PSNR (Fuzzy)
duck3	4 bytes	64.9498	38.791
ALARME4	7bytes	43.9256	22.3119
Mouth shut	9bytes	48.2885	32.2842
Test_mono2	11bytes	56.9656	28.5061
ALARME1	18bytes	48.8073	22.5057
Jump	25 bytes	67.4514	47.1444

Here, in Table 5.2, We have taken different wave audio files and different message files, here we have compared PSNR values of RSA encryption algorithm with Reduced RSA, so we found that PSNR values of Reduced RSA algorithm 's are high as compared to RSA.

Table 5.3 Reduced RSA's PSNR values of GA with Fuzzy System

Audio file size	Message size	GA	Fuzzy
ALARME1	1byte	3.4252e-05	0.00303425
	6bytes	0.000205512	0.00320551
	9 bytes	0.000308268	0.00330827
	13bytes	0.000445276	0.00344528
	16bytes	0.000548032	0.00354803

And in Table 5.3 we have compared the PSNR result of GA based LSB method with the Fuzzy Gaussian system, and we found that GA based LSB method has improved PSNR values as compared to Fuzzy system.

Table 5.4 Reduced RSA's MSE values of GA with Fuzzy system

Audio file size	Message size	MSE (GA)	MSE (Fuzzy)
duck3	4 bytes	3.05822	4.93639
ALARME4	7bytes	27.3325	32.9129
Mouth shut	9bytes	16.5399	10.4411
Test_mono2	11bytes	6.0909	16.1308
ALARME1	18bytes	15.581	32.1867
Jump	25 bytes	1.82134	1.88686

Here, in Table 5.4, We have taken different wave audio files and different message files, here we have compared MSE values of GA based LSB method with Fuzzy Gaussian system, so we found that GA based LSB system has lower values of MSE as compared to Fuzzy system. So the level of distortion (error) between original and stego audio file is less in proposed GA based LSB method.

An obvious technique for the disruption of steganography in any media is the addition of noise, where individual samples are changed by a random value hence modeling electrical noise. Noise addition or interference happens to transmitted signals on various communication channels, e.g. due to thermal noise or crosstalk. As a result audio data is affected as well when transmitted over such channels. Due to an increased noise level and distorted sample values the steganography receiver may not be able to extract all the embedded information properly, depending on the actual embedding algorithm used. This operation directly influences the quality of the audio signal, which degrades with increasing noise amplitude. In general a signal-to-noise ratio (SNR) above 20dB guarantees for is a reasonable audio quality. As a result our approach uses very little noise amplitudes, therefore keeping the SNR above 20dB and minimizing the influence on the audio quality while still interfering with the stenographic receiver [12].

The embedding capacity (EC) [13] indicates the maximum data size that it is probable to hide in the cover object. It is defined as follows:

$$EC = \frac{secrete\ message\ size}{cover\ object\ size}$$

Table5. 5 Compare Reduced RSA's capacity of GA with Fuzzy system

Audio file size	Message size	RSA (PSNR)	Reduced RSA(PSNR)
duck3	4 bytes	64.4205	64.9498
ALARME4	7bytes	42.8147	43.9256
Mouth shut	9bytes	47.8811	48.2885
ALARME1	18bytes	42.141	48.8073
Nuclear	21 bytes	34.2401	34.4459
Jump	25 bytes	62.997	67.4514

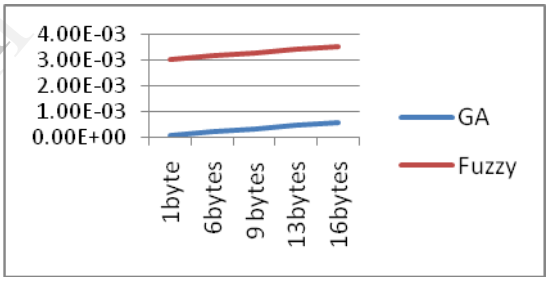

Fig 5.1 Compare Reduced RSA's capacity of GA with Fuzzy system

Here in Table 5.5 we are comparing our proposed Reduced RSA GA based system with the fuzzy system. We have taken one audio file size and different message file size. Regarding the size of message, and the ratio between message size and host size, the host size must be large enough that the message can be embedded into. In other word, the size of host and the size of message are related to each other.

So, here we found that proposed GA based LSB system can not handle more payload as compared to Fuzzy system. i.e. From the above result, message embedding capacity in audio file, of fuzzy system is more as compared to our proposed GA based LSB system.

Table 5.6 Compare Reduced RSA's MSE, PSNR of GA with Fuzzy system

Audio file size	Message size	MSE (GA)	MSE (FUZZY)	PSNR (GA)	PSNR (FUZZY)
ALAR	1byte	0.6497	0.666664	76.4049	56.1811
	6bytes	3.91681	6.56871	60.8005	36.3096
	9 bytes	7.7219	8.89115	54.9047	33.68
	13bytes	15.8555	14.1499	48.6556	29.6441
	16bytes	29.8017	23.9054	43.1744	25.0893

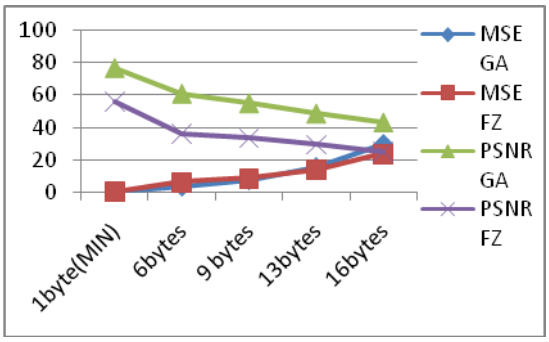

Fig 5.2 Compare Reduced RSA's MSE, PSNR of GA with Fuzzy system

VI. CONCLUSIONS

By using this method of data hiding the observer will not be able to suspect that the data is there at all. for a second time, if someone knows that data is in the audio, it is very difficult to extract the data from the host audio. The key idea of the algorithm is random and higher LSB layer bit embedding keeping minimal embedding distortion of the host audio. Using the proposed genetic algorithm, message bits can be embedded into multiple, indistinct and deeper layers to achieve higher capacity and robustness. The basic idea behind this is to maintained uncertainty in message bit insertion into audio data for hiding the data from hackers and to provide a good, well-organized method for hiding the data from hackers and sent to the destination in a safer manner.

By using Reduced RSA algorithm, we successfully decreased the size of the RSA encrypted file by using the merssanes prime numbers and enhanced RSA encryption algorithm.

By comparing GA based LSB method with Fuzzy system, we have found that GA is more robust as compare to fuzzy, because fuzzy can not handle more distortion. But by comparing payload of both the methods, fuzzy has more capacity to handle messages.

REFERENCES

[1] Fridrich, J. et al. (2000) 'Steganalysis of LSB encoding in color images', Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE Press, New York, pp.1279–1282

[2] Petitcolas, F. A. P., Anderson, R. J., & Kuhn, M. G.(1999). Information hiding—a survey. Proceedings of IEEE, 87(7), 1062–1078.

[3] Zamani, M., Manaf, A.A., Ahmad, R.B., Zeki, A.M., & Abdullah, S. A genetic-algorithm-based approach for audio steganography. World Academy of Science, Engineering and Technology, 54 (2009).

[4] Mazdak Zamani, Hamed Taherdoost, Azizah A. Manaf, Rabiah B. Ahmad, and Akram M. Zeki, "Robust Audio Steganography via Genetic Algorithm", IEEE, 2009.

[5] Juhi Saurabh, Asha Ambhaikar , Audio Steganography using RPrime RSA and GA Based LSB Algorithm to Enhance Security, International Journal of Science and Research (IJSR), India Online ISSN: 2319-7064

[6] Samir K Bandyopadhyay, Debnath Bhattacharyya, Debashis Ganguly, Swarnendu Mukherjee and Poulami Das, "A Tutorial Review on Steganography"

[7] Shilpa M Pund, Chitra G Desai," Enhancing RSA algorithm using Mersenne Primes with reduced size of encrypted file", International Journal Of Computers & Technology Vol 7, No 1, ISSN 22773061

[8] Shilpa M Pund, Chitra G Desai, " Implementation of RSA algorithm Using Mersenne Prime" , International Journal of Networking & Parallel Computing www.cirworld.com (ISSN: 2319-4529) Volume 1, Issue 3, Dec 2012-Jan 2013

[9] Raffaele Pinardi1, Fabio Garzia12, Roberto Cusani1,"Peak-Shaped-Based Steganographic Technique for MP3 Audio", Journal of Information Security, 2013, 4, 12-18

[10] Toran Lal Sahu1, Mrs. Deepty Dubey , "A Survey On Edge Detections And Denoise Techniques", International Journal of Science, Engineering and Technology Research (IJSETR) Volume 2, Issue 1, January 2013, ISSN: 2278 – 7798

[11] Vijaypal Dhaka, Ramesh C. Poonia Yash Veer Singh, " A Novel Algorithm for Image Steganography Based on Effective Channel Selection Technique" , International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8, August 2013 ISSN: 2277 128X

[12] Prof. Samir Kumar, Bandyopadhyay Barnali, Gupta Banik, "Lsb modification and phase encoding technique of audio steganography revisited", International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 4, June 2012

[13] Raffaele Pinardi, Fabio Garzia, Roberto Cusani, "Peak-Shaped-Based Steganographic Technique for MP3 Audio", Journal of Information Security, 2013, 4, 12-18