Security Enhancement in Audio Steganography Using LSB Method and Encryption Algorithm

Navdeep[1],
PG Student Department of Computer Science, Shri Ram college of Engineering & Management,
NH2, Delhi-Mathura Road, Palwal, Haryana, India

Ms Neha Goyal[2]
Assistant Professor,
Department of Computer Science,
Shri Ram college of Engineering & Management,
NH-2, Delhi-Mathura Road, Palwal, Haryana, India

ABSTRUCT-Steganography or information hiding in a medium is concerned with embedding information in an audio, image, or video file in an imperceptible manner. Human auditory and visual imperfections, which lead to psychoacoustic masking effects in hearing and vision, are exploited for modifying a host, or cover, audio or image in accordance with a given piece of covert information. Since the modification is carried out in the masked regions of perceptibility, the information-embedded medium, or the 'stego' signal, appears to be the same as the original host signal.

Steganography is 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 or Steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second message present.

In a digital world, Steganography and Cryptography are both intended to protect information from unwanted parties. Both Steganography and Cryptography are excellent means by which to accomplish this but neither technology alone is perfect and both can be broken. It is for this reason that most experts would suggest using both to add multiple layers of security.

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.

LSB is a well-known method for Steganography. Using only steganography for security of message might be harmful in certain cases. Some of the other technologies also be embedded with steganography for better security of messages. A lots of researcher suggest encryption with steganography.

Keywords:- Steganography, LSB, Cover Data, Stego File, Text Embedding on Audio, DES.

1. INTRODUCTION

Security measures must be incorporated into computer systems whenever they are potential targets for malicious or mischievous attacks. This is especially so for systems those handle financial transactions or confidential, classified or other information whose secrecy and integrity are critical.

'Steganography' is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message. This is in contrast to cryptography, where the existence of the message itself is not disguised, but the content is obscured. The advantage of steganography over cryptography alone is that messages do not attract attention to themselves, to messengers, or to recipients.

Steganography is an application that makes use of steganographic technique. The Modules include Embedding message, Embedding file, Retrieving message, and Retrieving file. Steganography hides any kind of files including text, picture files, audio files, video files. Cover text can be picture, audio or video files in any format. Steganography provide the option for compressing the contents to be hidden. An encryption password option has been provided, to provide extra security.

The product (software) is also supporting:

i) **Interactive GUI:** with interacts GUI user can easily interact or use this software.

ii) **Encryption/Decryption:** It allows you to embed the messages or files in encrypted form using 32 bit DES algorithm which means that once encrypted, the message or file could be retrieved (or decrypted) from a master file only after specifying the correct password which was used at the time of encryption.

iii) **Compression:** It allows embedding messages and files in compressed form using ZIP compression format. Gives you a choice of compression level to be used: low, normal or high.

1.1 STEGANOGRAPHY AND CRYPTOGRAPHY

Steganography or information hiding in a medium is concerned with embedding information in an audio, image, or video file in an imperceptible manner. Human auditory and visual imperfections, which lead to psychoacoustic masking effects in hearing and vision, are exploited for modifying a host, or cover, audio or image in accordance with a given piece of covert information. Since the modification is carried out in the masked regions of perceptibility, the information-embedded medium, or the 'stego' signal,

appears to be the same as the original host signal.

Steganography is 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 or Steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second message present.

In a digital world, Steganography and Cryptography are both intended to protect information from unwanted parties. Both Steganography and Cryptography are excellent means by which to accomplish this but neither technology alone is perfect and both can be broken. It is for this reason that most experts would suggest using both to add multiple layers of security.

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.

1.2 GENERIC STEGANOGRAPHIC SYSTEM

As with any other science, steganography has its own set of terminology. The term cover is used to describe the original message in which we will hide our secret message. Once we embed our secret message into the cover, the new message is known as the stego data. The stego data is analogous the cipher text of cryptography.

A generic steganographic system, or stego system, works thusly. A secret message is embedded into the cover data using some sort of embedding algorithm. The cover data may be a single file, but that is not necessarily the case. The embedding algorithm then outputs the stego data. Additionally it is advisable to encrypt the secret message prior to embedding it.

There are some requirements that all embedding algorithms should meet. Firstly, the distortion of the cover data as a result of the embedding algorithm should be as imperceptible as possible. Secondly, no part of the secret message should be contained in the header of the stego data file. The message must become part of the cover data and should be immune to manipulation attacks such as re-sampling or filtering.

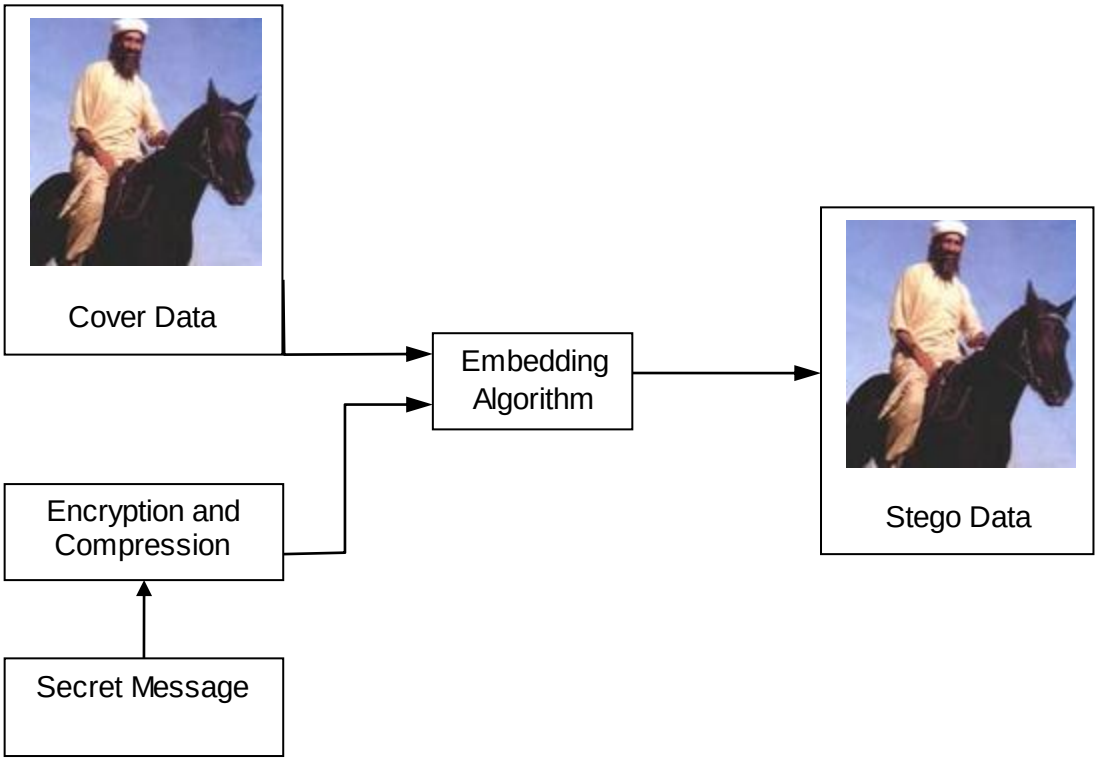


Fig 1.1 An Overview of a Generic Steganographic System.

2. PROPOSED WORK

2.1 Technique used

Steganographic Techniques used is concealing data within encrypted data. The data to be concealed is first encrypted before being used to overwrite part of a much larger block of encrypted data. This technique works most effectively where the decrypted version of data being overwritten has no special meaning or use. Each pixel typically has three numbers associated with it, one each for red, green, and blue intensities, and these values often range from 0-255. Each number is stored as eight bits (zeros and ones), with a one worth 128 in the most significant bit (on the left), then 64, 32, 16, 8, 4, 2, and a one in the least significant bit (on the right) worth just 1.

2.2 How Steganography Works

Steganography strips less important information from digital content and injects

hidden data in its place. This is done over the spectrum of the entire image. Here's one way it could be implemented: The following sequence of 24 bits represents a single pixel in an image. Its 3 bytes of color information provide a total of 256 different values for each color (red, green and blue) and thus can represent a total of 16.7 million colors. This particular value displays as a dark green:

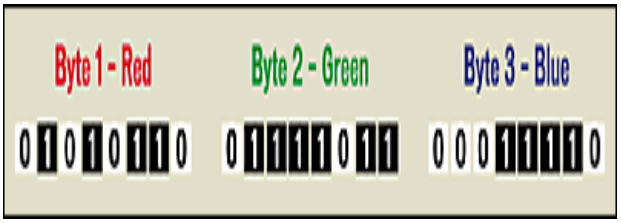


Fig 2.1 Bytewise Insertion of Data

Now, let's take 11 of these pixels that represent, say, part of a solid-color background. In the following sequence, the least significant (rightmost) bit of each 8-bit byte has been co-opted to hide text message—the four characters Aha!—in ASCII binary:

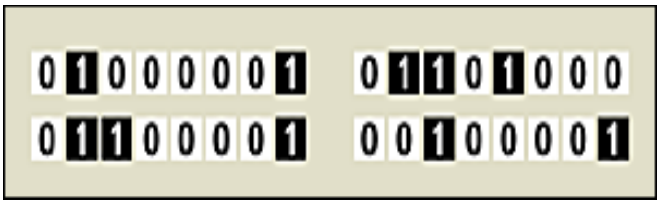
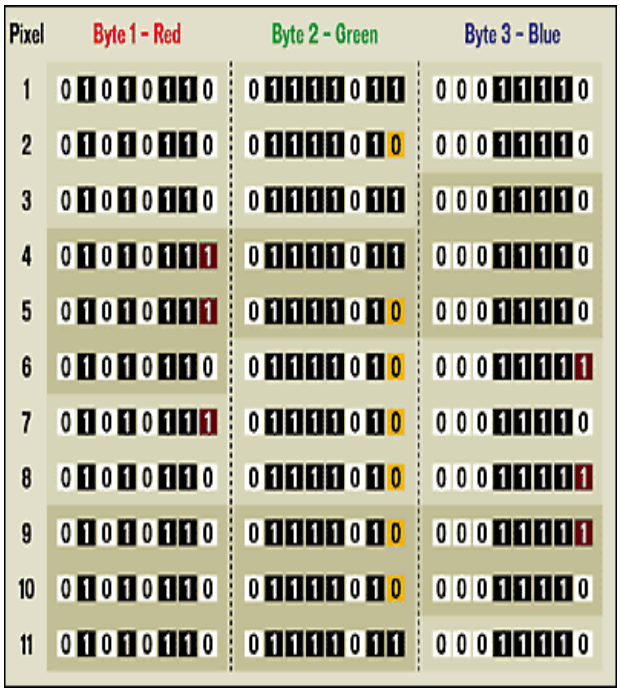


Fig 1.3 Bitwise Representation of Text Message



Here are the bits behind those 11 pixels:

Fig 2.2 Message Hidden in Pixel Positions

The hidden message occupies 32 of those 264 bits (about 12%) and contains four 8-bit bytes. In the diagram, each maroon or gold box represents a bit that had to be changed to include the hidden message. Notice that only 15 of 264 bits (less than 6%) had to be changed and only eight of the 11 pixels were altered.

2.3 LSB Modification for Images

LSB modification is perhaps the most popular method to embed a message into cover data. As its name suggests, this method works by modifying the least significant bit of one of the RGB values of the pixel data. The secret message data is then scattered pseudo-

randomly across the image.By using the LSB of each byte (8 bits) in an image for a secret message, you can store 3 bits of data in each pixel for 24-bit images and 1 bit in each pixel for 8-bit images. As you can see, much more information can be stored in a 24-bit image file. Depending on the color palette used for the cover image (i.e., all gray), it is possible to take 2 LSB's from one byte without the human visual system (HVS) being able to tell the difference. The only problem with this technique is that it is very vulnerable to attacks such as image changes and formatting (i.e. changing from .GIF to .JPEG).

This method is quite effective against human detection because it is difficult for the human eye to discern an LSB modified pixel.24-bit true-color RGB data formats are best suited for LSB modification, it is possible to use this method with 8-bit color-index data formats.

2.4 LSB Modification for Audio Files

Embedding secret messages in digital sound is usually a more difficult process and embedding messages in other media, such as digital images. In order to conceal secret messages successfully, a variety of methods for embedding information in digital audio have been introduced. But the most common method is LSB modification.

Modifications done using LSB will usually not create audible changes to the sounds. Another method involves taking advantage of human limitations. It is possible to encode messages using frequencies that are inaudible to the human ear. Using any frequencies above 20.000Hz, messages can be hidden inside sound files and will not be detected by human checks.

Least significant bit insertion in audio file

Let’s assume an audio file had the following 8 bytes of data in it somewhere:

180, 229, 139, 172, 209, 151, 21, 104

In binary, this would be:

10110100-11100101-10001011-10101100-
11010001-10010111-00010101-01101000

If we wanted to hide the byte value ‘214’ (11010110), we usethe least significant bit from each byte to hide our byte:

10110101-11100101-10001010-10101101-
11010000-10010111-00010101-01101000

The changes result in the following bytes, which are so closeto the originals that the difference will be inaudible:

Modified: 181, 229, 138, 173, 208, 151, 21, 104

Original: 180, 229, 139, 172, 209, 151, 21, 104

2.5 LSB Modification for Video Files

Video files are generally a collection of images and sounds, so most of the presented techniques on images and audio can be applied to video files too. The great advantages of video are the large amount of data that can be hidden inside and the fact that it is a moving stream of images and sounds. Therefore, any small but otherwise notice able distortions might go by unobserved by humans because of the continuous flow of information.

2.6 Algorithm Used

The DES (Data Encryption Standard) algorithm is the most widely used encryption algorithm in the world. For many years, and among many people, "secret code making " and DES have been synonymous. DES works on bits, or binary numbers--the 0s and 1s common to digital computers. Each group of four bits makes up a hexadecimal, or base 16, number. DES works by encrypting groups of 64 message bits, which is the same as 16 hexadecimal numbers. To do the encryption, DES uses "keys" where are also apparently 16 hexadecimal numbers long, or apparently 64 bits long. However, every 8th key bit is ignored in the DES algorithm, so that the effective key size is 56 bits. But, in any case, 64 bits (16 hexadecimal digits) is the round number upon which DES is organized.

3. RESULTS OBTAINED

3.1 EMBEDDING OF FILE

Fig. 4.1 Main Screen

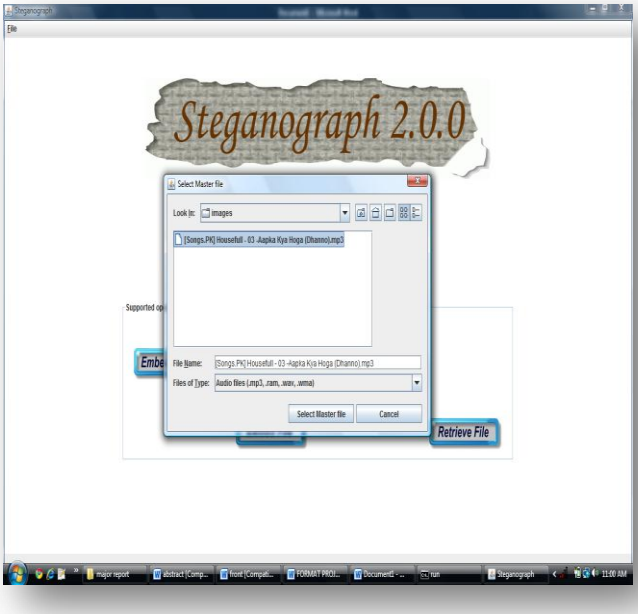


Fig. 4.2 Selecting Master File

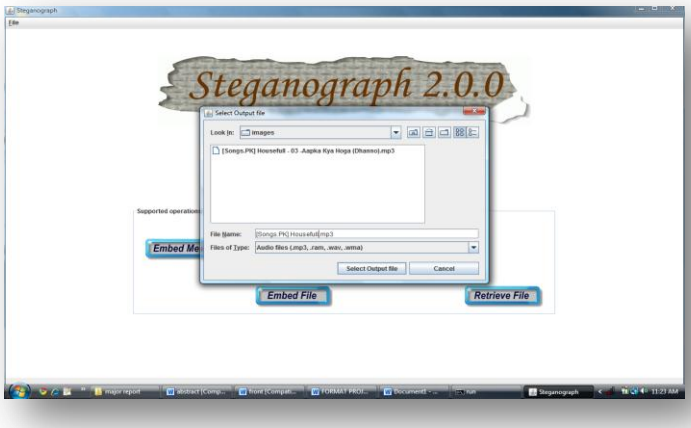


Fig. 4.3 Selecting Output File

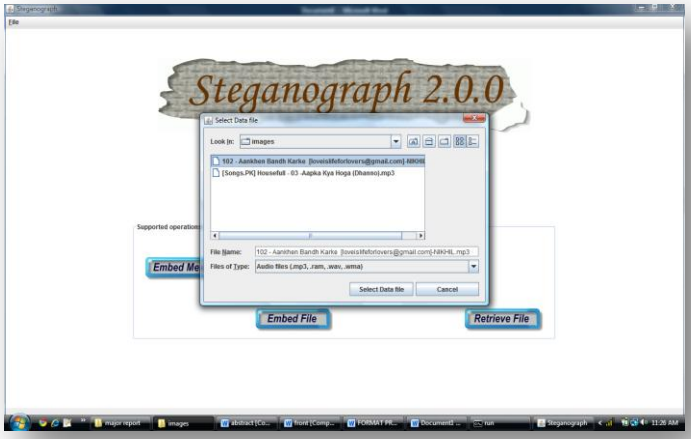


Fig. 4.4 Selecting Data File

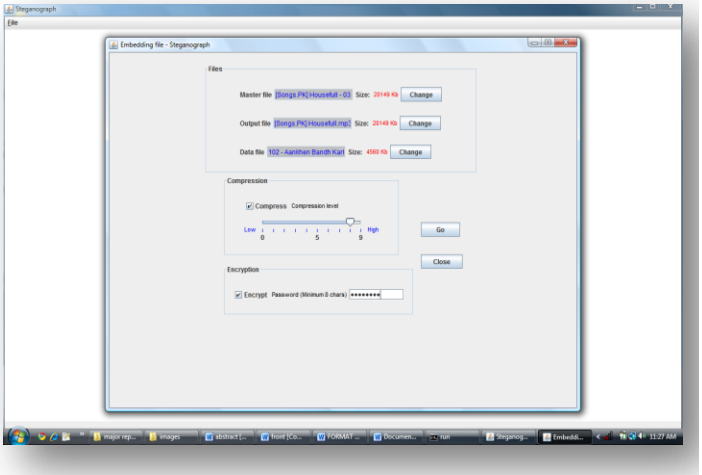


Fig. 4.5 Embedding File

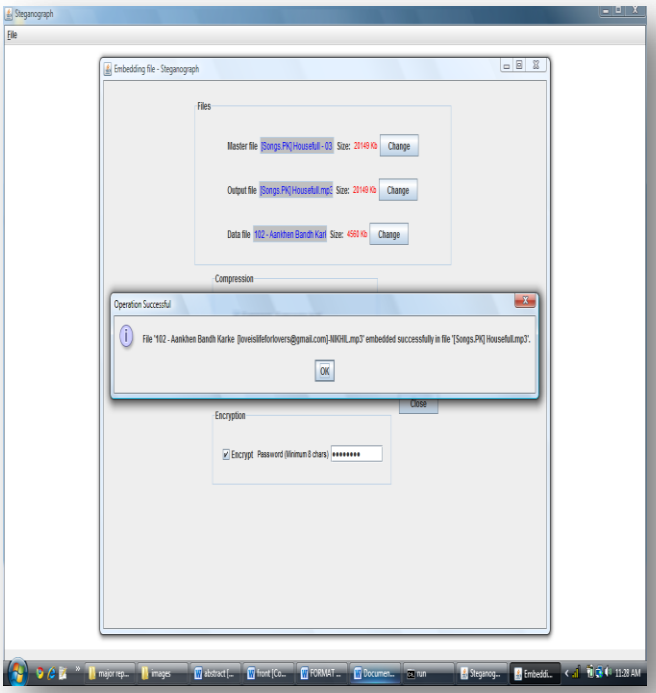


Fig. 4.6 File Embedded

4.2 RETRIEVING OF FILE

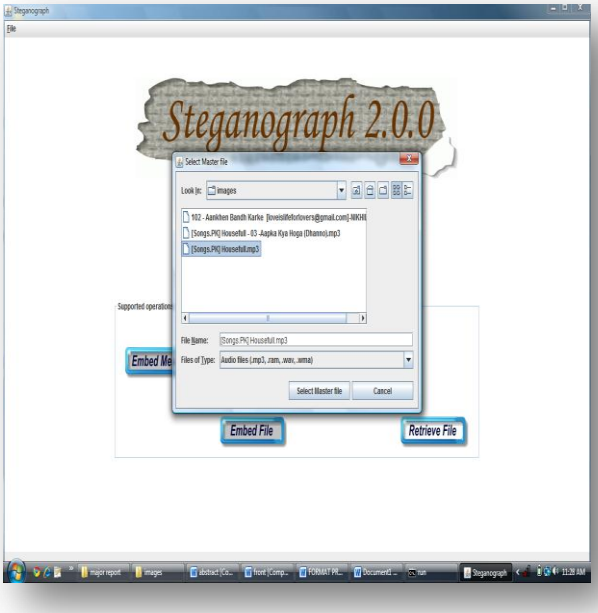


Fig. 4.7 Retrieve File

Fig. 4.8 Information About Hidden File

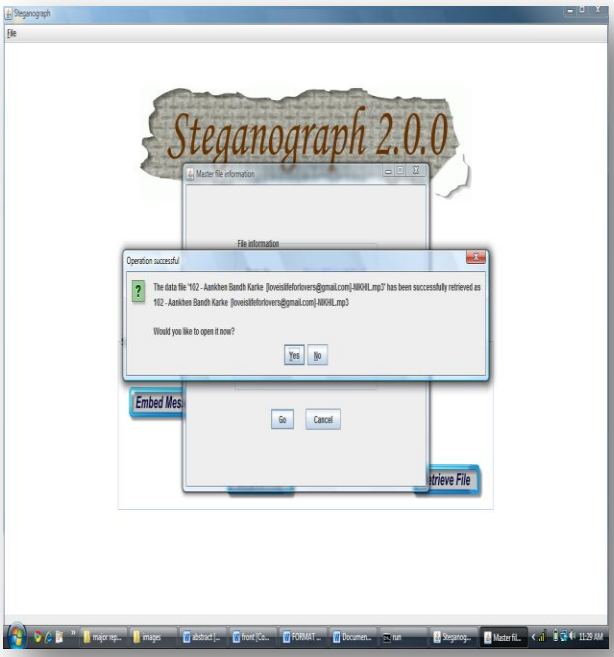


Fig. 4.10 Opening of File

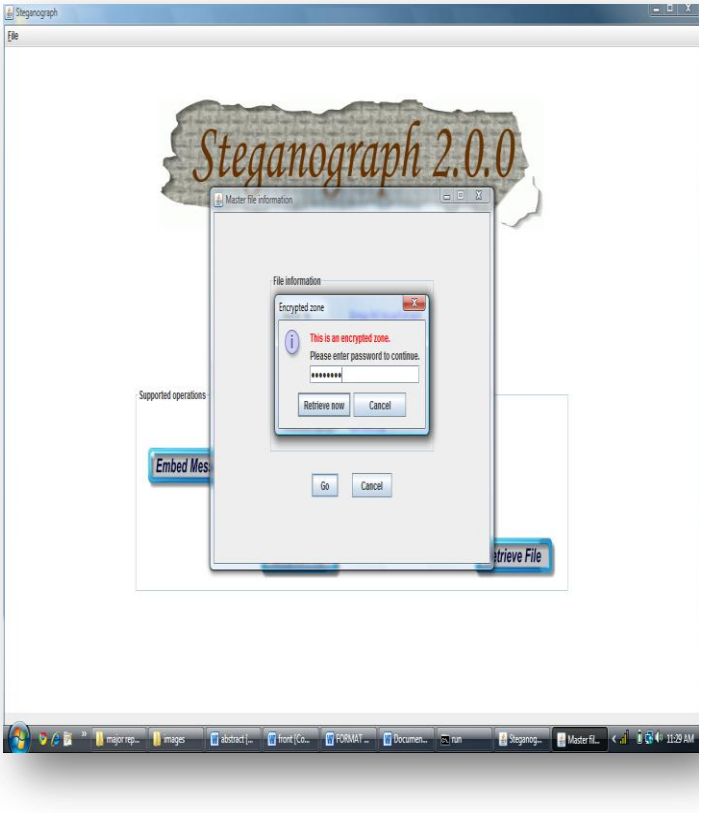


Fig. 4.9 Retrieving File by Entering Password

Fig. 4.11 File Retrieved Successfully

5. CONCLUSION AND FUTURE WORK

This Paper has been providing how we can developed a secure method for embedding any information into audio, video or images files. Enhancements should be made to hide information in TCP/IP or another protocol.

In the near future, the most important use of steganographic techniques will probably be lying in the field of digital watermarking. Content providers are eager to protect their copyrighted works against illegal distribution and digital watermarks provide a way of tracking the owners of these materials. Although it will not prevent the distribution itself, it will enable the content provider to start legal actions against the violators of the copyrights, as they can now be tracked down.

References:-

1. David Flangan (2005) "*JAVA in Nutshell*",O'Reilly publication, New Delhi.

2.Herbert Schildt (2002) "*java 2 complete reference*", McGraw-Hill Publication Pvt. Ltd, New Delhi.

3. Kathy Sierra & Bert Bates"*Head First Java*", O'Reilly publication, New Delhi.

WEB REFRENCES

The websites referred for completing this paper are as follows:-

1. http://en.wikipedia.org/wiki/Steganography
2. http://en.wikipedia.org/wiki/StegoText
3. http://www.webopedia.com/TERM/S/steganography.html
4. http://www.watermarker.com/books/Steganography-.aspx
5. http://www.youdzone.com/cryptobooks_Steganography.html
6. http://www.StegoArchive.com
7. http://www.cl.cam.ac.uk/~fapp2/steganography
8. http://www.jjtc.com/stegdoc/index2.html
9. http://www.wepin.com/pgp/stego.html
10. http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html
11. http://rr.sans.org/covertchannels/mp3stego.php