Image Steganography using Least Significant Bit algorithm

Sujana Krishnan[1], Reuma Akhtar[2], Saurav Singh[3], Arun.V[4]

[1] student,department of computer science and engineering, SRM IST Ramapuram, tamilnadu, India
[2] student, department of computer science and engineering, SRM IST Ramapuram, tamilnadu, India
[3] student, department of computer science and engineering, SRM IST Ramapuram, tamilnadu, India
[4] asst professor ,department of computer science and engineering, SRM IST Ramapuram, tamilnadu, India

ABSTRACT

Steganography is the technique of hiding private or sensitive information within something that appears to be nothing but a usual image. Steganography involves hiding text messages, so it appears that to be a normal image. This system lets a user upload an image and enter the text to send secretly, and gives a key or a pass word to lock the text.User then sends the image and key to the receiver and receiver first opens the image, and then he enters the key or password for decryption of text, he then presses decrypt to get secret text from the sender. If sender sends this image in public, others will not know what is it, and it will be only be received by receiver with the key to decrypt it.The objective of this project is to develop a secure path for sending or receiving secret text messages. The one that is implemented here is a variation of plain LSB (Least Significant Bit) algorithm. The text message is encrypted and sent to receiver very securely. The system uses AES encryption to encrypt the user's secret text message and key information while sending it to receiver. Also Diffie-Hellman key generation is used for sharing secret key between sender and receiver.

Keyword: *-image, steganography, cryptography, security, encryption, hidden message*

1. INTRODUCTION

Steganography is the art of hiding the fact that communication is taking place, by hiding information in other information. Many different carrier file formats can be used, but digital images are the mostpopular because of their frequency on the Internet. For hiding secret information in images, there exists a large variety of steganographic techniques some are more complex than others and all of them have respective strong and weak points. Different applications have different requirements of the steganography technique used. For example, some applications may require absolute invisibility of the secret information, while others require a larger secret message to be hidden. This paper intends to give an overview of image steganography, its uses and techniques. It also attempts to identify the requirements of a good steganographic algorithm and briefly reflects on which steganographic techniques are more suitable for which applications.

1.1 Different kinds of steganography

Almost all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. Redundancy can be defined as the bits of an object that provide accuracy fargreater than necessary for the object's use and display. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. Image and audio files especially comply with this requirement, while research has also uncovered other file formats that can be used for information hiding.

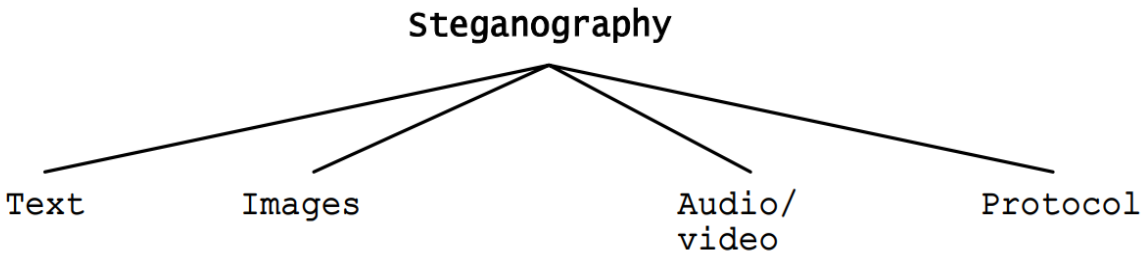

Figure 1: Categories of steganography

Hiding information in text is historically the most important method of steganography. An obvious method was to hide a secret message in every nth letter of every word of a text message. Text steganography using digital files is not used very often since text files have a very small amount of redundant data.

Given the proliferation of digital images, especially on the Internet, and given the large amount of redundant bits present in the digital representation of an image, images are the most popular cover objects for steganography. To hide information in audio files similar techniques are used as for image files. One different technique unique to audio steganography is masking, which exploits the properties of the human ear to hide information unnoticeably. A faint, but audible, sound becomes inaudible in the presence of another louder audible sound. This property creates a channel in which to hide information. Although nearly equal to images in steganographic potential, the larger size of meaningful audio files makes them less popular to use than images. The term protocol steganography refers to the technique of embedding information within messages and network control protocols used in network transmission. In the layers of the OSI network model there exist covert channels where steganography can be used. An example of where information can be hidden is in the header of a TCP/IP packet in some fields that are either optional or are never used.

2. IMAGE STEGANOGRAPHY

As stated earlier, images are the most popular cover objects used for steganography. In the domain of digital images many different image file formats exist, most of them for specific applications. For these different image file formats, different steganographic algorithms exist. Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image. The least significant bit (in other words, the 8th bit) of some or all of the bytes inside an image is changed to a bit of the secret message. When using a 24-bit image, a bit of each of the red, green and blue colour components can be used, since they are each represented by a byte. In other words, one can store 3 bits in each pixel. An 800 × 600 pixel image, can thus store a total amount of 1,440,000 bits or 180,000 bytes of embedded data.
For example a grid for 3 pixels of a 24-bit image can be as follows:

(00101101 00011100 11011100)
(10100110 11000100 00001100)
(11010010 10101101 01100011)

When the number 200, which binary representation is 11001000, is embedded into the least significant bits of this part of the image, the resulting grid is as follows:

(00101101 00011101 11011100)
(10100110 11000101 00001100)
(11010010 10101100 01100011)

Although the number was embedded into the first 8 bytes of the grid, only the 3 underlined bits needed to be changed according to the embedded message. On average, only half of the bits in an image will need to be modified to hide a secret message using the maximum cover size. Since there are 256 possible intensities of each primary colour, changing the LSB of a pixel results in small changes in the intensity of the colours. Thesechanges cannot be perceived by the human eye - thus the message is successfully hidden. With a well-chosen image, one can even hide the message in the least as well as second to least significant bit and still not see the difference. In the above example, consecutive bytes of the image data – from the first byte to the end of

the message – are used to embed the information. This approach is very easy to detect. A slightly more secure system is for the sender and receiver to share a secret key that specifies only certain pixels to be changed. Should an adversary suspect that LSB steganography has been used, he has no way of knowing which pixels to target without the secret key.

In its simplest form, LSB makes use of BMP images, since they use lossless compression. Unfortunately to be able to hide a secret message inside a BMP file, one would require a very large cover image. Nowadays, BMP images of 800 × 600 pixels are not often used on the Internet and might arouse suspicion. For this reason, LSB steganography has also been developed for use with other image file formats.

3. IMPLEMENTATION

```
Color pixel = bmp.GetPixel(j, i);

// now, clear the least significant bit (LSB) from each pixel element
R = pixel.R - pixel.R % 2;
G = pixel.G - pixel.G % 2;
B = pixel.B - pixel.B % 2;

// for each pixel, pass through its elements (RGB)
for (int n = 0; n < 3; n++)
{
    // check if new 8 bits has been processed
    if (pixelElementIndex % 8 == 0)
    {
        // check if the whole process has finished
        // we can say that it's finished when 8 zeros are added
        if (state == State.Filling_With_Zeros && zeros == 8)
        {
            // apply the last pixel on the image
            // even if only a part of its elements have been affected
            if ((pixelElementIndex - 1) % 3 < 2)
            {
                bmp.SetPixel(j, i, Color.FromArgb(R, G, B));
            }

            // return the bitmap with the text hidden in
            return bmp;
        }
```

Figure-2: Code Snippet

The least significant bit algorithm is implemented in C#. A user interface is also built for easy usability of the program. This system comprises of 5 Modules as follows:-

- **Image Selection:** Here, User selects an image for sending a secret message.
- **Encrypting the Data**: Here, User enter/inputs the text that is to be hidden in the image. User sets a key and uses the encryption technique to encrypt the data.
- **Downloading the Image**: After hiding the text with the encryption technique, user can save the encrypted image file and store it into local system.
- **Send Email**: Here, user can share the secret message with one or more people by sending them the process id, required key details via email.
- **Decrypting the Data**: Once, the receiver receives the image, he/she can decrypt the image and the original secret message will be displayed.

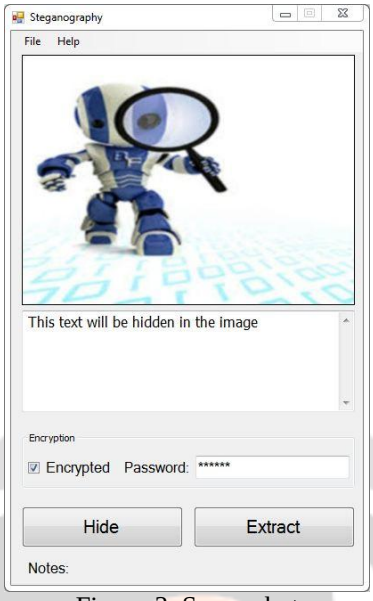

Figure-3: Screenshot

4. CONCLUSIONS

In this work, we have successfully implemented image steganography using least significant bit(LSB) algorithm in C#.

5. REFERENCES

[1]. M. Murali Krishna, Nirmal Roberts, "Enhancement of embedding capacity and security in reversible steganography", Wireless Communications Signal Processing and Networking (WiSPNET) International Conference on, pp. 803-807, 2016.
[2].Abdullah AlWatyan, Wesam Mater, Omar Almutairi, Mohammed Almutairi, Aisha Al-Noori, Sa'ed Abed, "Security approach for LSB steganography based FPGA implementation", Modeling Simulation and Applied Optimization (ICMSAO) 2017 7th International Conference on, pp. 1-5, 2017.
[3].Hassan Elkamchouchi, Wessam M. Salama, Yasmine Abouelseoud, "Data hiding in a digital cover image using chaotic maps and LSB technique", Computer Engineering and Systems (ICCES) 2017 12th International Conference on, pp. 198-203, 2017.
[4]. Anderson, R.J. & Petitcolas, F.A.P., "On the limits of steganography", IEEE Journal of selected Areas in Communications, May 1998
[5].Chandramouli, R., Kharrazi, M. & Memon, N., "Image steganography and steganalysis: Concepts and Practice", Proceedings of the 2nd International Workshop on Digital Watermarking, October 2003