Steganography in .bmp File Format

Vincentius Martin 13510017
Informatics Engineering
School of Electrical Engineering and Informatics
Bandung Institute of Technology, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]*vincentiusmartin@students.itb.ac.id*

***Abstract*—Sometimes, some messages need to be hidden in order to make unauthorized parties can't know the content of the messages. Efficient way is needed to do this, one way that can be used is using steganography. Steganography can hide the content of the message without afraid of the message looks suspicious. The medium for steganography is diverse, it can be paper message, advertise, digital media, etc. The digital media is used nowadays in so many aspects of life, that's why the information transfer using digital media is really popular and efficient. There are many digital medias that can be used like image, video, music, etc. In this paper, it will be discussed the use of .bmp image file format to hide message. Some contents are tested to be hidden in .bmp file format that is used. There are some advantages of using .bmp file format and in this paper, they will be stated. Some characteristics and structures from bmp image are needed to be understood in order to use this format.**

***Index Terms*—bmp, hidden file, image, steganography.**

I. INTRODUCTION

Some messages are secret, of course we don't want unauthorized parties know the message. There are many ways of doing this like using cryptography as a means to make the unauthorized parties can't know the content of the message, below is an example of encryted message :

> ÞÑnÙ»?ç?AÞíTÊßÖ?+IâJ?yÇ`G • (RMP?
> ¼wP¸ªíT$?Dêú+?<Ïï#E1eºõj?Â$??Ô)j£9
> ??àõA¿rÈR\?/SÃÝèm*»>)¢VnTPåKÇJ?6
> áñP--¼ín?®'¼6îÈG5L?¸v`·SðºÅ • c?ß? •
> ÁF?PÂ?äÕïÒ½?5J • Î¯#$k1ÃU;{»dkÂãD
> Ñ • nç=ºðí?àt9?yÚ<ß@åkÔxèº',¦^¶;ø

As it can be seen above, cryptography will make the information looks suspicious. People who see the message will be suspicious with the message, because cryptography make the message loses its meaning. We need another way to make other people believe that the message that we send is just like normal message that has no hidden information in it. We need to make the message looks really natural. How can we do this? Steganography is a way that can be used, because it hides the content of the message without afraid the content is about to be known by unauthorized parties that don't have authorization to see the message.

Steganography is the art and science of hiding message in another message by the way of embedding it. The goal of steganography is different with the goal of cryptography where the goal of cryptography it is to make data unreadable from a third party, the goal of steganography is to hide the data from a third party. There are so many steganography methods that are used. With using digital media like computers and networks, these methods are easier to be implemented. Noawadays, steganography can hide large amounts of information in any forms such as image, music, video, text, etc.

In this paper, image data is used as a medium to hide message. By using image, the information can be hidden without significantly changing the image. It just change some bits from the image and the human's eyes cannot tell the different between steganographic image and real image. One of the most used steganography application is for digital watermarking. If we see the content from the internet, any images can be claimed as property of anyone. Even if the image has signature on it, the signature can easily be erased. With digital watermarking, the creator of the image can embed digital signature or any other file that can represent the creator inside an image. The person who sees the image will not notice that there exist a mark that represents the creator in it because the mark itself cannot be seen.

Steganography in different image format will result in different technique. The one that is used in this paper is bitmap image format. Bitmap is one of many digital image representations that is used as an image format. It carries the extension of .bmp. This file format is really suitable for steganography because it is uncompressed. Compressed images are really hard to be modified and may cause information loss inside that image. By using bitmap image file format, we can get pixels from the bytes that we are going to use. One bit is used in a bitmap image to indicate where the pixel should represent the specified color position. By changing one bit in LSB position of a pixel, the change will not be known (LSB modification technique). Bitmap data has header and the frames. The LSB bits of the frames will be changed to hide the information that we want to hide inside the image.

Picture 1.1 Image with signature (top) and image with erased signature (bottom)

Picture 1.2 Digital watermarking technique

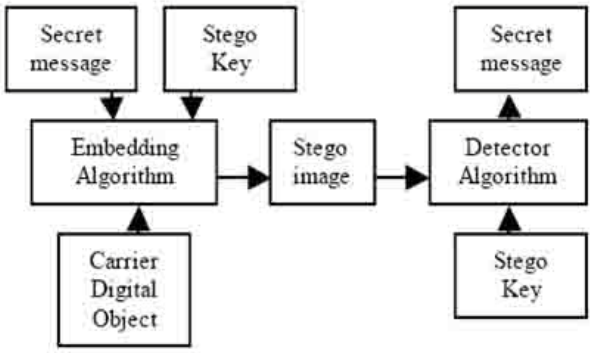
II. BASIC THEORIES

2.1.Steganography

There are some terminologies in steganography :

- Embedded message → hidden message (can be any format such) in specified file format.
- Cover object → the object that is used to hide embedded message, must be in specified format.
- Stego object → the cover object that has filled with hidden messge.
- Stego key → a key that is used to hide and reveal the message.

The outline of steganography itself is "combine" carrier object (cover object) by using certain steganography algorithm. With stego key, it can result in new image which is really similar with original image. This new image, when extracted using stego key again will produce the secret message and show it to the authorized user. Extraction algorithm will get specified bits in cover image and combine them to make bytes that can produce the hidden message again.

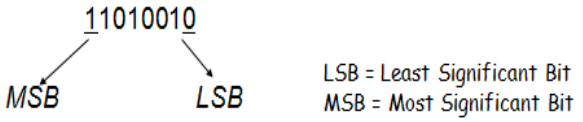


Picture 2.1 Steganography outline

2.2. Spatial Domain Method

Digital image contains of pixels. The pixels can be represented as a matrix (2D array) that is consist of width and height. An image with 200x150 size has 200 * 150 = 30000 pixels in it. Every pixel contains colors information that image has. By modifying the byte that contains pixel in an image, we can insert a bit from the message that is going to be hidden using steganography.

The method that is used in this paper to hide the message is spatial domain (LSB modification) technique. It utilizes the weaknesses of the human senses of sight. It changes the LSB pixel bit with the bit data that is going to be hidden from the original message. The pixel data will just change one bit will not affect human's visual perception when seeing an image.



Picture 2.2 Example of binary data representation in one byte

Let's take an example, suppose we have embedded message three bits : 101. We also have three bytes from image that is used as a medium to hide message like this :

10001100 00110101 01001101

The next step, let's put the bit 101 inside the bytes above so it will become like this :

1000110**1** 0011010**0** 0100110**1**

Only one bit change in the LSB position will not affect the image when it is seen by humans. If we just change one LSB bit, it is needed one byte for every one bit. Of course more than one bit can be used as long as it doesn't change the image colors significantly.

2.3. BMP Image Format

A bitmap image file can be divided into two parts : header and bitmap data. The header which containts 54 bytes data can be divided again into two sub blocks : bitmap header and bitmap information. Bitmap header is used to decide whether the image is in the correct bitmap format or not. Next, the bitmap information that contains some informations about the specified bitmap file. Below is the exact list of the informations that are contained in bitmap information block data :

Bitmap Header :

Byte	Content
0-1	Bitmap identifier 0x42 0x4D (in ASCII it reads BM)
2-5	File size
6-9	Reserved
10-13	Data offset

Table 2.1. Bitmap header content

Bitmap Information :

Byte	Content
14-17	Header size
18-21	Image width
22-25	Image height
26-27	Number of color planes
28-29	Color depth
30-33	Compression method
34-37	Data size
38-41	Image horizontal rule
42-45	Image vertical resolution
46-49	Number of colors used
50-54	Number of important colors used

Table 2.2. Bitmap information content

After bitmap information block, to the last pixel, there are bitmap data. The bitmap data block contains the image pixels. First line, represent the bottom line of an image. Also, pixels are stored in reversed, blue first, after that green, and the last is red. For the steganography use, we can use byte 55 and more.

III. IMPLEMENTATION

Now that the basic theory about technique and some basic knowledge about bmp file, the steganography application can be implemented. The implementation consists of hiding message in cover-image and reveal the message again from the stego-image. In the implementation, it uses header that consist of some elements, this elements are really important to reveal image from stego object, they are filename (including the file extension) and size of the file (the number of byte data that it contains).

For hiding the file, some steps must be done. These steps contain of the steps required to hiding a message in an image with bmp format. The steps are :

- Get the filepath for message that is about to be hidden.
- Make a string that contains the filename (from filepath) of the message.
- Get all bytes in byte[] format from the message.
- Create an array of byte that will contain all data about the message that will be hidden includes its header. The array has size of (length of string filename + 6 + message byte size). Why there exist 6 in the computation? It will be explained later. The array of byte that have been created here will be called as hiddenbyte.
- For n first bytes in hiddenbyte, fill filename string to it.
- After that add a mark to separate it with the next header content, like '|'.
- Insert hiddenbyte size to the next four bytes (because integer). The parsing is needed to convert integer into array of byte with size 4. It can be done just by simply divide it with 256.
- Once again, add mark to separate it with the next content. This is why before +6 is needed because it contains two markers and four bytes integer.
- Fill the rest of hiddenbyte with the message data.
- Get filepath for the cover-image.
- Get all bytes data from cover image and store them in an array of byte, let's call it stegobyte.
- Copy all bits from every byte of hiddenbyte to stegobyte, start from index 55 sequentially. It uses MSB from the checked byte and then left shift it once to get the next bit.
- Make a bmp file result, lets call it "result.bmp" with stegobyte that has been modified in some LSB bytes.

The other methods that are used are fill the stegobytes in random order. It chooses random pixels using a seed that is generated from a key. With the same key it will result the same indexes for every pixel choosen. So with this, the message will be more secure because of the random position that it contains inside the stego image. Below is the algorithm that is used to make random numbers :

```
function pseudoRandom(input integer : seed) →integer
{function that is used to make a pseudo random number from the given seed}
DECLARATION
  m_w, m_z, result : integer
ALGORITHM
  long m_w ←seed;
  long m_z ←211;{choose any number for second seed}
  m_z ← 36969 * (m_z & 65535) + (m_z >> 16);
  m_w ← 18000 * (m_w & 65535) + (m_w >> 16);
  result ← (m_z << 16) + m_w;
  → |result % 25000000| {return absolute result}
```

The result from a sequence will be used in the next sequence as next seed. With a good random algoritm, a new random number will almost likely to be generated. A

good random algorithm is determined by how it can produce random numbers with the previous seed it has iteratively.

One other thing, before the hiding process started the file size that is about to be hidden need to be checked whether the stego-image is sufficient enough to keep the file or not. It can be computed by using image data size that is contained in byte 34 to byte 37 of header information. If the hidden file data size is smaller than image data size then the hiding process can be started, else the hiding process cannot be started and reject the user command.

To reveal the image, the LSB bits from the specified bytes must be gotten and they are concatenated each other. The concatenate is done for eight times to make eight bit representation and then make a one byte character from it. This step is done for several times until all bytes from the hidden file are gotten. In order to reveal the hidden messages, also, some steps are needed to be done, they are :

- Read all bytes from filepath that contains stego image and store it in an array of byte, let's called it stegobyte.
- Save bit from the specified byte's LSB and concatenate it with next byte until eight times (one byte is formed). The concatenate process for bit x with another bit y is done like this, using left shift and or operation :

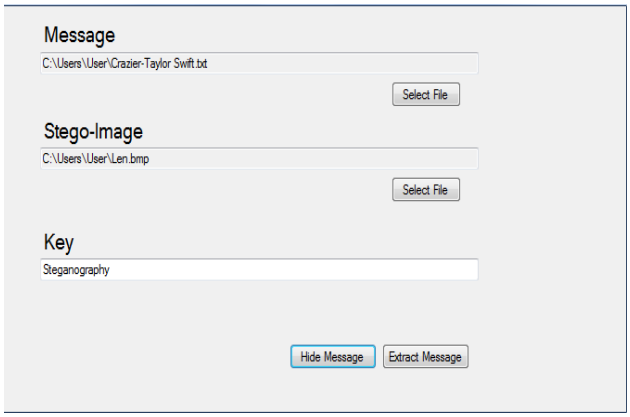
  $$(x << 1) \mid LSB(y)$$
- Save it and insert into a string, let's call it filename.
- The filename will be added with the new character for every eight bits until the marker is found.
- After the marker is found, the iteration to find image data size is started. The steps like before are done to find data bytes until the marker is found.
- The length of the image data size must be four bytes (an integer), after that, convert it into an integer value.
- Last step is to get all LSB bits that are hidden from the original message by iterating the stegobytes in number of image data size times and store it in an array of byte, let's call it hiddenbyte.
- After the hiddenbyte is formed, write all hiddenbyte's characters into a file and name it with the filename that has been gotten at the first process of revealing image.

The process of revealing image can also be done by using pseudo random algorithm. By using the same key with hiding image process, it will result in same pixel's indexes and give the same byte sequences with the hiding image process. With this random process, it can be ensured that the message's bytes is distributed in stego-image's bytes.

IV. Implementation Result

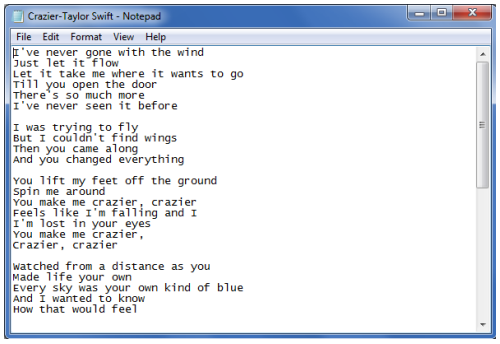
Steganography algorithm in implementation section is implemented in C# language for some .bmp images and some files. A program that accept input filepath of stego image and the message that is about to be hidden. The user can also input a key to use pseudo random algorithm and hide file in random pixels in stego-image.

4.1. Hiding Text File In An .bmp Image



Picture 4.1 Program interface

Picture 4.2 Original image



Picture 4.3 Message

Picture 4.4 Stego image

This implementation succesfully hide the message in the image by using the given key. After that, the stego-image is tested for extraction with the same key. The result is the same text file with the text file before hidden process. This means that the hiding and extracting message are done succesfully.

4.2. Digital Watermarking

Nowadays, we can see that digital arts are widely distributed. Anyone who has the image can claim it as his/her creation because the true owner sometimes cannot prove that the image is his/her own creation. Of course the owner can add their signature on the image such as on the bottom right corner of the image, in the middle, or anywhere on the image. But with current technology, the signature can be erased easily. So it is needed the ownership that cannot be known by other people inside the image and we can do that by using digital watermarking.

Digital watermarking is the insertion of information (called as watermark) that contains the ownership of multimedia data. Digital watermarking is one of the most used application in image steganography. It hides the ownership information inside an image. Unlike signature that can be erased easily, digital watermarking cannot be seen and it is hard to be erased because of it position in image bytes. The information that is hidden can be in various form like digital file signature, card, logo, etc. The purpose of using watermarking is to give copyright protection to an image. Because copyright is really easy to be manipulated, so the use of watermarking technique can help to solve this problem.

Below is the digital watermarking technique by adding signature image from the art creator itself. It is an illustration picture that pictures bull in some appearances.

Picture 4.5. Original art image

Picture 4.6. Creator's signature

Picture 4.7. Stego-image (bulls picture + signature)

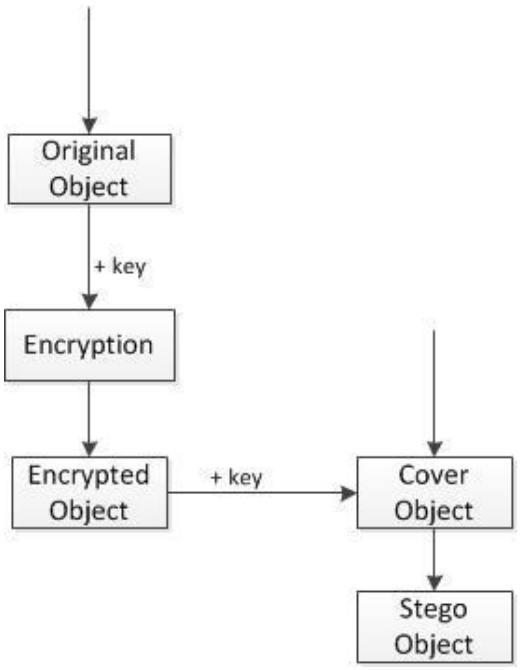
The creator signature's can be extracted succesfully from stego-image that has been created. It means that the stego-image succesfully inserted with creator's signature. As it can be seen above, the image is really similar to the original image so anyone who sees it won't be able to recognize that there is exist an signature in some bytes of the image.

V. Cryptography as An Compelement for Steganography

Cryptography and steganography as it has been explained, are two different things. Although they have the same function that is to remove the existence of information, but the purpose of using that two techniques are really different. Cryptography makes the message cannot be understood while steganography hides the existence of the message. Both methods can provide security but the study of both can make a powerful combination.

So, what is the advantages of using both techniques? Although they both have different usage purpose, can they be combned? Of course they can and there are also some advantages of combining them. The advantages of combine crpytography and steganography are :

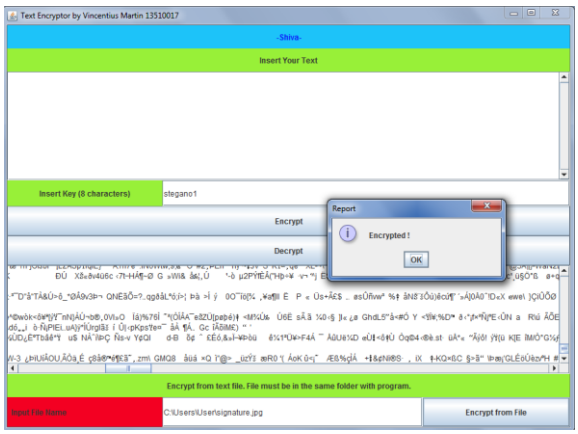
- Add multiple layers of security, by using both techniques we can make something like double protection from both.
- Make the analysis complexity higher by using hidden message and encrypted message.
- Satisfy the requirements such as capacity, security and robustness.



Picture 5.1. Cryptography and steganography combination

In the combination of cryptography and steganography, cryptography is used to make message loses its meaning and after that steganography is used to hide it. First, the message will be encrypted with specified key. After that, the result will be hidden in a cover object using steganography by using the specified key too. The steganography key can be different or same with encryption key.

Let's take example in digital watermarking, although the steganography can be solved, there will be only broken signature that is found. The analyzer must still think about how to solve the enchiper object to reveal the hidden object. The signature itself cannot easily be found because of this security level.



Picture 5.2. Encrypt signature file, after that the encrypted file can be used to be hidden in cover-image.

VI. CONCLUSION

By using steganography, the information inside a message can be hidden without afraid of getting suspicious. Steganography in image is good to be applied in .bmp file because .bmp file is uncompressed image format. In uncompressed image format, the frame is in high quality so it can hide the message easier and better than compressed format. The ability to use existing file format and manipulating it, is really important to use steganography technique. After all, it can be said that steganography is really powerful to communicate because of its nature to hide information.

The combination of steganography and cryptography can result in strong information security. This combination can provide double layer protection to any messages. With the further development, this technique can have a great impact to world communication.

REFERENCES

[1] http://www.kewlwallpapers.com/images/wallpapers/ws_Digital_Art_1024x768-675313.jpeg

[2] http://www.wisegeek.com/what-is-a-bitmap-image.htm

[3] FileFormat. The BMP File Format. March 26, 2013 (8.00 AM) < http://www.fileformat.info/format/bmp/corion.htm>

[4] Grantham, Beau .2007. *Bitmap Steganography : An Introduction.* Dr.Dutton.

[5] Johnson, Neil F. Informatio n Hiding : Steganography and Digital Watermarking. March 26, 2013 (7.00 AM) < http://www.jjtc.com/Steganography/>

[6] Kessler, Gary C. Steganography: Hiding Data Within Data. March 26, 2013 (7.07 AM) < http://www.garykessler.net/library/steganography.html

[7] Munir, Rinaldi. 20012. *College Slides IF3054*. Program Studi Teknik Informatika STEI ITB.

[8] Raphael,A Joseph; Sundaram. *Cryptography and Steganography – A Survey.* Coimbatore : India.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 26 Maret 2013

Vincentius Martin 13510017