International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 6 | ISSN : 2456-3307

Overview on Various Techniques Developed In Steganography By Different Agencies

Arpana Chaturvedi, Poonam Verma
Jagannath International Management School, Vasant Kunj, Delhi, India

ABSTRACT

In the era of ICT, every stream needs to use the online transactions. Although the citizens of the developed nations are at ease to work with it, however the citizens of the developing nations are still struggling to handle the same. The basic reason behind the lack of ease to work with the online transactions is the security issues of it. In this paper, we have discussed the concept of steganography, which has been popularly used. We have also given an overview of the various techniques developed by different agencies to make it more secure and not only use it for the basic images but also for secure storage and linkage , along with the watermarking techniques.

Keywords: Steganography, Watermarking, Online Signature, LSB

I. INTRODUCTION

The need of the hour is to maintain secret in the transactions of the e-commerce. The number of people buying and selling business transactions on Web is increasing at a phenomenal pace, which requires the special form of security to be added. The security of common public key was based on the Non polynomial complete problems, which was not assured to be solved even by Turing machine. The steganography is helpful to hide the secrets in secure communications. It is the art of concealing the existence of information and it combines both the networking protocols use and security protocols.

The word *steganography,* derived from the Greek language, literally means *covered writing*. This concept includes a high volume of methods of secret communication that hides the very existence of the message. Steganography is the technique of taking one piece of information and hiding it within another. Computer files, whether they are images, sound recordings, text and word processing files, or even the medium of the disk itself, all contain unused areas where data can be stored. The files can then be exchanged with no indication of the additional information that is stored within. The general idea of hiding some information in digital content has a wider class of applications that go beyond steganography, Fig. 1

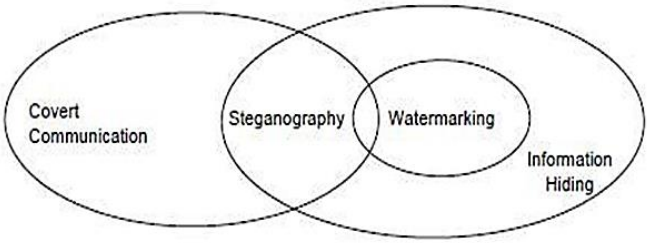


Figure 1. Relationship between Steganography and related fields

Storage space on disks is divided into ‘clusters,’ that is file systems are of a fixed-size. When data is stored to the disk, even if the actual data being stored requires less storage than the cluster size, an entire cluster is reserved for the file. The space which has not been used by the file in the cluster is termed as the ‘slack space.’ Thus, this slack space was used to hide information for steganography. Nowadays, the combinations of steganography and cryptography methods are used to ensure data confidentiality [5] and to improve information security. Steganography is used in other grounds also like copy right, preventing e-document forging. Table 1 shows the comparison of various secret communication techniques used nowadays.

Table 1. Comparison between different Secret Communication Techniques

Secret Communication Techniques	Confidentiality	Integrity	Un-removability
Encryption	Yes	No	Yes
Digital Signatures	No	Yes/No	No
Steganography	Yes/No	Yes/No	Yes

Digital Steganography

The applications of digital steganography in various e-commerce applications is rampant and include digital watermarking for multimedia data, digital signature authentication and validation of electronic documents, digital data storage and linkage for digitized photographs with personal attribute information, as well as secure communication of multimedia data. Targeting these applications, DataMark Technologies (DMT) agency has developed four digital

Steganography products which are as follows:

1. Secure Communication
2. Digital Signature Authentication
3. Digital Watermarking
4. Digital Storage and Linkage

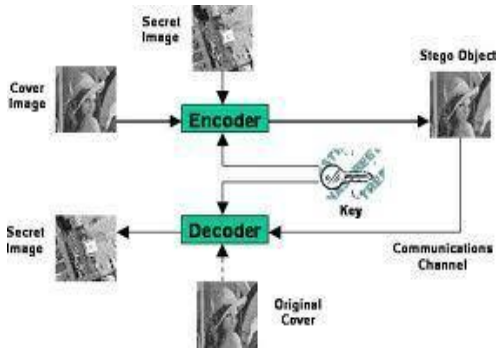


Figure 1. Process of steganography

Figure 1 shows a simple representation of the generic embedding and decoding process in steganography. In this example, a secret image is being embedded inside a cover image. The first step in embedding and hiding information is to pass both the secret message and the cover message into the encoder. Inside the encoder, one or several protocols will be implemented to embed the secret information into the cover message.

II. SECURE COMMUNICATION

Above listed is a digital steganography that permits the user to select a multimedia data file for embedding the required text in the file. A keyfile named as "Stegfile" utilizing the basics of the digital images is used to encrypt the given file. The basic operation is to hide the required code in the least significant bit of the code as it does not create much impact on the original look of the file.

The basic advantage is that it permits to utilize any digital format to be used for the encryption.The cryptographic algorithm does not corrupt or overwrite the container file, multimedia data posted on any webpage, such as images (JPEG, GIF), video clips (AVI, MPEG) or audio files (WAV, MIDI), and thus the media can be selected as the container file. Furthermore, customized container files, such as the voices and images of the sender captured via video conferencing, can be generated very easily.

The operations involved in using are listed as below:

- ✓ A multimedia container file is first chosen from the PC hard disk or from a webpage on the Internet.
- ✓ The knowledge of this container file must be pre-determined and communicated securely between the sender and receiver.
- ✓ The algorithm generates a hash file from the inputs of the container file and the hidden text. This file contains random data based on a number of mathematical operations between the two input files which does not bear any data resemblance to either the container or the hidden file.

III. DIGITAL SIGNATURE AUTHENTICATION

Digital Signature Authentication helps to prevent malicious tampering of private and confidential documents such as company memos, Emails and letters. This algorithm is highly useful and can provide a various applications such as business transactions between banks and customers, legal document exchanges between lawyers and clients, and scenarios involving non-repudiation issues. This authentication warns the user of the any unauthorized activity that takes place. A digital signature and a multimedia container password are embedded where; the container password can either be a normal text string, an image,

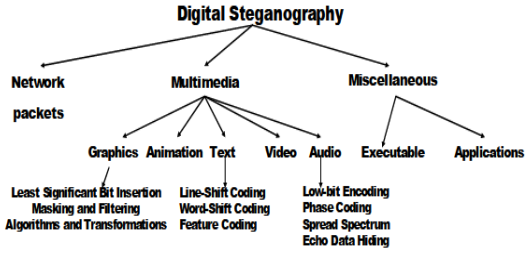
or a binary file. Mathematical random lock combinations for data embedding are also included as an added security layer for the version dealing with the highly sensitive data.

IV. DIGITAL WATERMARKING

Piracy of data has led agencies to develop digital watermarking software for copyright protection of digital images, music CDs, DVDs, and other formats of popular multimedia data. In the case of digital images, the files can come from a variety of sources, such as the Internet, digital still cameras, and video cameras. This software allows embedding either a text or image watermark invisibly into an "unlabelled" image. The recognition of a watermark is an important issue as it can be used in the court of law to defend the true ownership of the intellectual property.

V. DIGITAL LINKAGE AND STORAGE

A Software algorithm can be used to provide a secure data linkage between digital images and attribute text information. The attribute information can be any personal records, which may be manually or electronically filed. In case the hacker is able to gain the security access to the database, the hackers can easily tamper the records.The main task of the algorithm is to link the personal record and the digital photograph together and then create a hash file that can be safely stored in a database. This hash file is unique to and can only be decoded with the original photograph and associated personal record. Tampering with any one of these files will render the decoding process ineffective. The database administrator will be able to determine whether these files have been modified, by checking the original hash file with the digitized photograph. An optional password is also available to protect the hash file prior to data storage.



VI. ALGORITHMS USED IN STEGANOGRAPHY

There are four algorithms currently implemented, each use least significant bit steganography and some filter the image first.

BLINDHIDE:
This is the simplest way to hide information in an image. It *blind*ly *hide*s because it just starts at the top left corner of the image and works it's way across the image (then down - in scan lines) pixel by pixel. As it goes along it changes the least significant bits of the pixel colors to match the message. To decode the process the least significant bits starting at the top left are read off. This is not very secure - it's really easy to read off the least significant bits. It also isn't very smart - if the message doesn't completely fill up the possible space then just the top part of the image is degraded but the bottom is left unchanged - making it easy to tell what's been changed.

HIDE SEEK:
This algorithm randomly distributes the message across the image. It is named after "*Hide* and *Seek*" - a Windows 95 steganography tool that uses a similar technique. It uses a password to generate a random seed, then uses this seed to pick the first position to hide in. It continues to randomly generate positions until it has finished hiding the message. It's a little bit smarter about how it hides because you have to try every combination of pixels in every order to try and "crack" the algorithm - unless you have the password. It's still not the best method because it is not looking at the pixels it is hiding in - it might be more useful to figure out areas of the image where it is better to hide in.

FILTER FIRST:
This algorithm filters the image using one of the inbuilt filters and then hides in the highest *filter* values *first*. It is essentially a fancier version of BlindHide as it doesn't require a password to retrieve the message. Because we are changing the pixels we need to be careful about filtering the picture because we don't want to use information for filtering that might change. If we do, then it may be difficult (if not impossible) to retrieve the message again. So this algorithm filters the most significant bits, and leaves the least significant bits to be changed. It is less noticeable on an image because using the filter ensures we are hiding in the parts of the image that are the least noticeable.

BATTLE STEG:

The best of the above described algorithms. This algorithm performs "*Battle*ship *Steg*anography". It first filters the image then uses the highest filter values as "ships". The algorithm then randomly "shoots" at the image (like in HideSeek) and when it finds a "ship" it clusters it's shots around that hit in the hope of "sinking" the "ship". After a while it moves away to look for other ships. The effect this has is that the message is randomly hidden, but often hidden in the "best" parts to hide in thanks to the ships. It moves away to look for other ships so that we don't degrade an area of an image too greatly. It is secure because you need a password to retrieve the message. It is fairly effective because it is hiding (if you set the values right) the majority of the information in the best areas.

VII. DYNAMIC BATTLESTEG AND FILTERFIRST

These two algorithms do the same as BattleSteg and FilterFirst, except they use dynamic programming to make the hiding process faster and less memory intensive. They are NOT compatible with the original algorithms because the order of pixels kept in the dynamic array is not exactly the same.

EVALUATION

The most important requirement is that a Steganographic algorithm has to be imperceptible. Below criteria has been proposed for imperceptibility of an algorithm:

1) **Invisibility:** The invisibility of a Steganographic algorithm is most important requirement. strength of Steganography lies in its ability to be unnoticed by human eye. The moment that one can see that an image has been tampered with, the algorithm is compromised [8].

2) **Payload capacity:** Watermarking, needs to embed only a small amount of copyright information, In other hand Steganography requires sufficient embedding capacity [9].

3) **Robustness against statistical attacks:** Statistical Steganalysis is the practice of detecting hidden information by applying statistical tests on image data. Many Steganographic algorithms leave a "signature" when embedding information that can be easily detected through statistical analysis.

4) **Robustness against image manipulation:** While being transmitted the image may undergo changes by an active attacker in an attempt to remove hidden Information. Image manipulation, such as cropping or rotating, can be performed on the image. This may destroy the hidden message. It is required for Steganographic algorithms to be robust against malicious changes to the image.

5) **Independent of file format:** The most powerful Steganographic algorithms thus possess the ability to embed information in any type of file.

6) **Unsuspicious files:** This requirement includes all characteristics of a Steganographic algorithm that may result in images that are not used normally and may cause suspicion. Abnormal file size, for example, is one property of an image that can result in further investigation of the image by a warden.

VIII. CONCLUSION

Although only some of the main image steganography techniques were discussed in this paper, there exists a large selection of approaches to hiding information in images. Different image file formats have different methods of hiding messages, that having different strong and weak points respectively. Whereas one technique lacks in payload capacity, while other lacks in robustness. For example, the patchwork approach has a very high level of robustness against most type of attacks, but it can hide only a very small amount of information. The Least significant bit (LSB) technique in both BMP and GIF can be used for large files, but these both approaches can result in suspicious files that increase the probability of detection when in the presence of a warden. An agent can decide on which steganography algorithm they wish to use, for which they require to decide upon the type of application on which the algorithm has to be implemented and dependent on the priority of an application, algorithm can be implemented.

Table 2. Comparison of image Steganography techniques

	Invisibility	Payload capacity	Robustness against statistical attack	Robustness against image manipulation	Independent of file format	Unsuspicious files
LSB in BMP	High	High	Low	Low	Low	Low
LSB in GIF	Medium	Medium	Low	Low	Low	Low
JPEG	High	Medium	Medium	Medium	Low	High
Spread Spectrum	High	Medium	High	Medium	High	High

IX. REFERENCES

[1]. B. Dunbar, 'A detailed look at Steganographic Techniques and their use in an Open-Systems Environment', Sans Institute: Information Security Reading Room, 2002.

[2]. E. Kawaguchi, 'Applications of Steganography', Datahide.org, 2015. [Online]. Available: http://datahide.org/BPCSe/applications-e.html. [Accessed: 05- Apr- 2015].

[3]. G. Kessler, 'An Overview of Steganography for the Computer Forensics Examiner', Garykessler.net, 2014. [Online]. Available: http://www.garykessler.net/library/fsc_stego.html. [Accessed: 05- Apr- 2015].

[4]. C. Oliboni, OpenPuff. EmbeddedSW.net, 2012. http://embeddedsw.net/OpenPuff_Steganography_Home.html

[5]. E. Zukerman, 'Review: OpenPuff steganography tool hides confidential data in plain sight', PCWorld, 2015. [Online]. Available: http://www.pcworld.com/article/2026357/review-openpuff-steganography-tool-hides-confidential-data-in-plain-sight.html. [Accessed: 05- Apr- 2015].

[6]. A. Abdel-Raouf, 'Picture, Java Files'.

[7]. J. Judge, 'Steganography: Past, Present,

[8]. Future', Sans.org, 2001. [Online]. Available: http://www.sans.org/reading-room/whitepapers/stenganography/steganography-past-present-future-552. [Accessed: 04- May- 2015].A. Davidson, 'Java Art Chapter 6. Steganography', Java Prog. Techniques for Games., 2009. http://fivedots.coe.psu.ac.th/~ad/jg/javaArt6/stego.pdf