

On the Development of Steganographic Tools

Batool Al-Sadoon, Hassan Mathkour, and Ghazy Assassa

*Department of Computer Science,
College of Computer and Information. Sciences
King Saud University
Riyadh, Saudi Arabia*

balsadoon@yahoo.com, mathkour@ccis.ksu.edu.sa, ghazy@ccis.ksu.edu.sa

Abstract

Digital steganography refers to the insertion of a secret message into a carrier file to covertly communicate the message. Steganography systems are usually composed of insertion and extraction systems. The insertion system takes a host file, a prepared message file, and an optional key to insert the message into the host for creating a cover host. The cover host is then stored or transmitted. The extraction system operates in reverse. It takes a covert host and an optional key as input and extracts the message. Many steganography systems have some forms of built in encryption and will automatically encrypt and decrypt the message as part of the process. The most common steganography systems use various image and audio formats as host files. These files are usually large and offer the bandwidth necessary for reliable hidden. Video systems are also emerging to exploit the large size of video data files and streams. Some steganography systems do not use a host file at all, but will generate a covert host based on the contents of the message. The degree of covertness of the host depends on the capabilities of the perceived threat. A steganography attacker aims to detect whether a message is present, and if so, to extract the message and exploit it. Detecting a message requires identifying a signature or in some way determining that something about a file is unusual. Until now, there is no universal steganalysis system that is suitable to all of the current steganography techniques. In this paper, we investigate various steganography techniques and tools. We develop a set of criteria to analyze and evaluate the strengths and weaknesses of the presented techniques. We discuss the requirements of more robust steganography techniques that takes advantages of the presented strengths and avoids the limitations.

Keywords: *Steganography techniques, steganalysis, digital images.*

1. Introduction

Similar to cryptography, steganography provides a means of communicating secrets. While cryptography scrambles a message so it cannot be understood, steganography hides the very existence of the message. An eavesdropper can intercept a cryptographic message, but may not even know a steganographic message exists. Encryption and steganography achieve the same goal via different means. Encryption encodes the data so that an unintended recipient cannot determine its intended meaning. Steganography, in contrast attempts to prevent an unintended recipient from suspecting that the data is there [4]. Combining encryption with steganography allows for a better private communication. The goal of steganography is to avoid drawing suspicion to the transmission of the secret message. On other hand, steganalysis is a way of detecting possible secret communication using against steganography. That is, steganalysis attempts to defeat steganography techniques. It relies on the fact that hiding information in digital media alters the carriers and introduces unusual signatures or some form of degradation that could be exploited.

A steganography system is usually composed of insertion and extraction subsystems. The insertion system takes a host file, a prepared message file, and an optional key to insert the message into the host for creating a cover host. The cover host is then stored or transmitted. The extraction system operates in reverse. It takes a covert host and an optional key as input and extracts the message. Some steganography systems have some forms of built in encryption and will automatically encrypt and decrypt the message as part of the process. Using steganography, communicating messages can be hidden in different media including text, audio, and image files. Such file are called carriers.

In the remaining of the paper, we discuss the different types of steganography in Section 2. In Section 3, we discuss image steganography techniques. In Section 4, we discuss different criteria for a successful steganography technique. An appraisal of image steganography techniques is given in Section 5. In Section 6, we evaluate a set of image steganography tools. Section 7 concludes the paper.

2. Types of Steganography

Text-Based Steganography

Text steganography can involve anything from changing the formatting of an existing text, changing words within a text or using context-free grammar to generate readable texts. It is an open question whether secure and robust steganography is possible with text messages. An example, attacker can simply try to reformat the text and so destroy all the information encoded in the text format. Additionally, text messages can be stored in different formats as (HTML, Postscripts, or PDF); the change from one

format to another might also be harmful to the embedded message. Text hiding techniques include: The Line-Shift Coding, the word shift coding, feature coding, syntactic technique, semantic technique, and cover generation techniques.

Audio Steganography

Like the text document, the sound files may be modified in such a way that they contain hidden information [8]. Such techniques embed data in sound files using the properties of the Human Auditory System (HAS). Figure 1 depicts the mechanism of hiding data in an audio file. Examples of audio steganography techniques include least significant bit, phase coding, and echo Hiding.

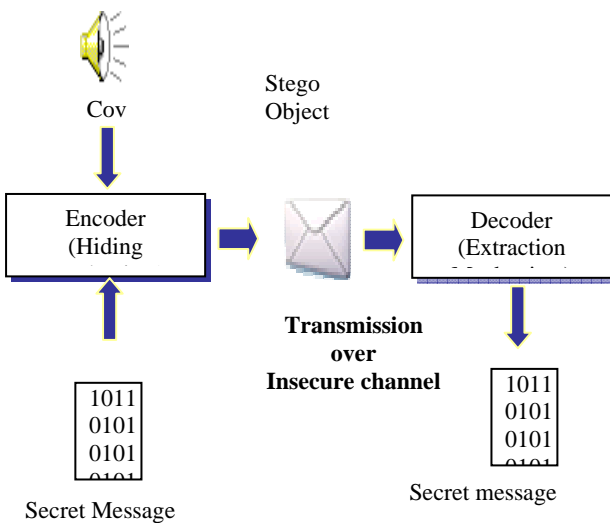


Figure 1: Hiding data in audio

Steganography in OSI Network Model

Data can be hidden in any of the OSI layers. For example, the network layer hide information using IP headers used for routing information. The unused IP

header bits (e.g. the *DF* and *MF* bits) or the two unused bits (the least significant bits) in the “*Type of service field*” can be used to create a covert channel. See Figure 2.

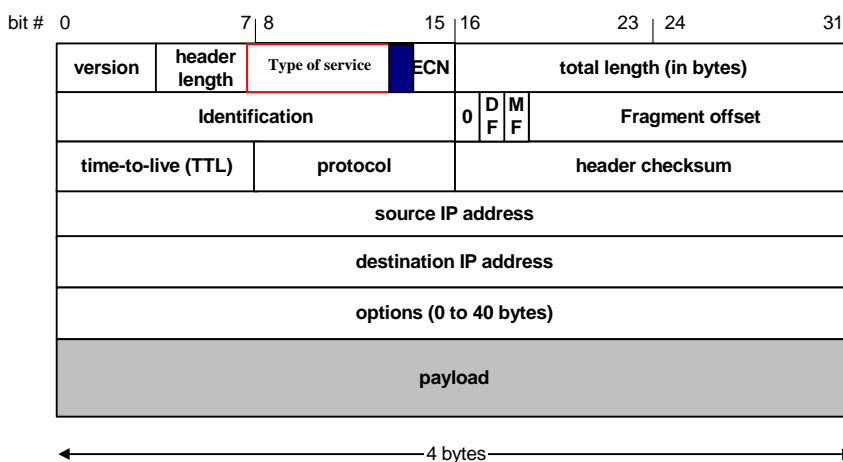


Image Steganography Figure 2: The IP header Compared to other types of steganography, image steganography has attracted extensive research as well as more popular usability in recent year. This is

due to the fact that huge amounts of data can be hidden without perceptible impact to the carriers and possibly because of the flood of electronic images that have recently become available. Since the focus

of this research is on image steganography, we will provide a detailed discussion of it. First we summarize the related work in the area of image steganography. Then, we describe the techniques for hiding information in images digital image properties and formats, and discuss the properties of successful techniques and evaluate the techniques against such criteria.

A early work on the image steganography is Least Significant Bit technique that attempts to minimize the detectability of hidden data by introducing as little distortion as possible during embedding. However, as pointed out by Fridrich and Goljan [14], recent advances in steganalysis have shown that this approach does not guarantee detectability, evidenced by the fact that they can be successfully attacked using statistical [15], or even visual attacks [13]. For Palette -based images, three techniques are proposed. The first one that embeds message directly in palette indices as illustrated in [9], so the palette will look suspicious as pointed out by Johnson [12]. Many efforts by Machado [7] try to reduce distortion by embedding message in the LSB of the palette colors channels and due to the duplicate colors the palette can be detected easily as explained by Fridrich [16]. Due to these facts, Fridrich [5] proposed a new technique using parity of palette color. For JPEG images, Westfeld [6] in his steganographic system F5 decrements the DCT coefficient's absolute values of JPEG image instead of overwriting the LSBs to defend against the proposed statistical and distortion attacks. In addition, message bits are distributed over the whole cover image, to defend against the visual attack. Since the F5 algorithm destroys stego image histogram [16], Provos [17] incorporated error correction technique in his steganographic system, Outguess that uses a two-pass algorithm, where bits are embedded in the first pass and changes are made to DCT coefficients in the second one to match the histogram of DCT coefficients of the stego-image with that of the cover image. Recently a Stegadetect tool can discover F5 and Outguess techniques successfully [16,18,19] and the main disadvantage of using DCT coefficient's techniques that the stego images can only be stored in the JPEG format. Many of the existing steganographic tools were developed

upon these techniques. Such tools include S-Tools, Hide & Seek, and Hide4PGP [7,8,11,12,16,18].

3. Image Steganography Techniques

Information can be hidden in many different ways in images. Messages are either directly inserted or they are scattered randomly throughout the cover image. Most image steganography techniques are image-based format dependent that mean each of these can be applied to various images, with varying degrees of success or suffers varying from operations performed on images, such as cropping or decreasing in the color depth. There are several such techniques. These include least significant bit, pseudorandom permutations, cover-regions and parity bits, patchwork technique, palette-based images, and transform domain techniques for JPEG images. For space constraints, we describe two popular techniques.

Least significant bit

This technique is common in steganography and is relatively easy to apply in image as in audio [1, 3, 8, 10]. It exploits the representation of binary information where in many architectures, the bits in each binary word, or byte, are stored in order from the most significant bit (MSB) to the least significant. For images, researchers believe that very little information is conveyed in the LSBs has little effect on the quality of the image. LSB encoding process performed by breaking the covert message into individual bits, and replacing the selected pixels' LSBs with the message bits.

When applying LSB technique to each byte of a 24-bit image, three bits can be encoded into each pixel (each pixel is represented by three bytes). The emphasized bits are the only bits that actually changed. The main advantage of LSB insertion is that data can be hidden in the least and second to least bits and still unnoticeable [3]. For example, the letter A can be hidden in three pixels. Assume the original three pixels are represented by the three 24-bit words. Inserting the binary value of A into the three pixels [Figure 3], starting from the top left byte, would result in [Figure 4].

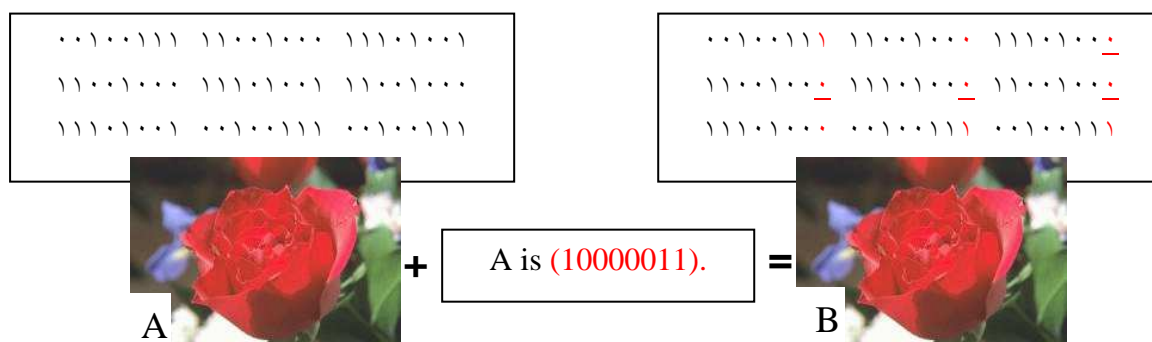


Figure 3: LSB example

Extracting messages in this technique is simple requires. It requires only identifying the proper LSBs of the covert host. Unfortunately, the steganalysis researchers found that this technique leaves a statistical signature, so they developed a tool used X^2 test to detect it easily [14]. In addition, it is extremely vulnerable to be destroyed simply e.g. re-saving for a BMP image can destroy the hidden information [3]. Further, this technique is not appropriate for JPEG and GIF format.

Pseudorandom permutations

A more sophisticated technique proposed in [2,11], where all cover bits can be accessed in the embedding process, the secret message bits can be distributed randomly over the whole cover. In the Encoding process a random number generator used and its output is used as indices where the embedded message bit is. The usage of a stego-key is important, because the security of a protection system should not be based on the secrecy of the

algorithm itself. Note that one index could appear more than once in the sequence, since we have not restricted the output of the pseudorandom number generator in any way. We call such case "collision". To overcome the problem of collisions, keep track of all cover bits which have already been used in a list. If during the embedding process one specific cover-element has not been used prior, add its index to the list and continue using it. If the index of the cover element is already included in the list, discard the element and choose another cover element pseudorandomly. At the receiver side, apply a similar technique. The basic steps of the pseudorandom permutations algorithm can be shown in [Figure 4]. This technique further increases the complexity for an attacker, since it is not guaranteed that subsequent message bits are embedded in the same order. As *LSB*, part of data that are randomly stored in LSB may be lost and this technique may produce a high noisy image if it stored huge bits in MSBs.

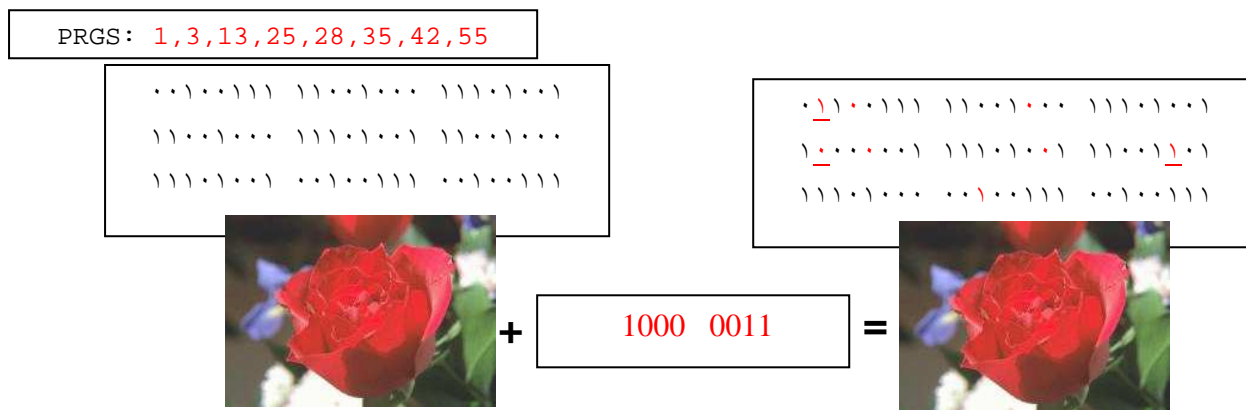


Figure 4: Pseudorandom permutations Technique example

4. Evaluation criteria of Image Steganography Techniques

Below is a list of criteria that measure the degree of success of an images steganography technique.

- **Level of Visibility (Perceptible or Imperceptible):**
Steganography techniques should embed information in such way that embed data don't leave a signs of steganography use as shown in [Figure5]. The visibility is directly influenced by the size of the secret message, the format and the content of the carrier image.
- **Detectability:**
The researchers of steganalysis develop a series of tools that detect using a specific steganography technique e.g StegSpy, Stegadetect, or Chi square that lead to investigate more and more techniques to avoid detectability problem.

- **Robustness vs. Payload:**

The embedded data should survive any reprocessing operation for the cover media which goes through and preserve its fidelity. A trade-off exists between the payload size (message size) and robustness. For example the LSB techniques have the capacity to hide larger amount of information in a cover image but a little reprocessing to the resulted image will destroy information completely, where embedding in blocks of pixels will be more robustness.

- **Domain Type:**

The techniques that use transform domain hide information in significant areas of the cover images but the main disadvantage of these is restricted to lossy format with different quality factor.

- **File Format Dependence:**

Some techniques employed to hide information depend upon specific characteristics to a carrier type or format while other techniques may work without relying on a specific file format.

5. An Appraisal of the techniques

The table below summarizes the adherence of various image steganography techniques with respect to the criteria of a successful technique.

TECHNIQUES		DOMAIN TYPE		FILE FORMAT DEPENDENCE		DETECTABILITY			PERCEPTIBILITY		ROBUSTNESS			CAPACITY		
		<i>Spatial</i>	<i>Transform</i>	<i>Yes</i>	<i>No</i>	<i>H</i>	<i>M</i>	<i>L</i>	<i>Visible</i>	<i>Invisible</i>	<i>H</i>	<i>M</i>	<i>L</i>	<i>H</i>	<i>M</i>	<i>L</i>
Least Significant Bit		X		X		X						X	X			
Pseudorandom permutations		X		X		X			X		X		X			
Patchwork technique		X		X			X			X		X	X			
Palette-based images Techniques	Using the palette order	X		X		X			X			X			X	
	Using the palette order	X		X		X			X			X			X	
	Using the image data	X		X		X			X			X			X	
Transform Techniques	Modulating the relative size of two DCT coefficients		X	X				X		X		X			X	
	Manipulating the LSB's of the DCT coefficients		X	X			X			X		X		X		

6. Steganography Tools

The table below summarizes the main features of few steganography tools.

No.	TOOL	Technique	Perceptibility level		DATA TYPE	Carrier File	File Format Dependence		Encryption	
			Visible	Invisible			yes	No	Yes	No
1.	EZStego	Palette rearrangement		X	Text	GIF	X			X
2.	Gif-it-up	LSB substitution		X	Any	GIF	X		X	
3.	Gifshuffle	Palette rearrangement		X	text	GIF	X		X	
4.	Hide and Seek	LSB substitution	X		Text	BMP	X		X	
5.	Hide 4PGP	LSB substitution		X	Text	BMP, WAV	X		X	
6.	S-Tools	LSB substitution	X		Any	BMP, GIF, WAV	X		X	
7.	Steganos 3 Security Suite	LSB substitution		X	Any	BMP VOC, WAV	X		X	
8.	Jsteg Shell	Using DCT		X	Any	JPEG	X		X	
9.	F5	Decrements and increments DCT coefficients		X	Any	JPEG	X		X	

7. Conclusion:

We have discussed several steganography techniques with an emphasis on image steganography. We develop a set of criteria to analyze and evaluate the strengths and weaknesses of the presented techniques. We discuss the requirements of more robust steganography techniques that takes advantages of the presented strengths and avoids the limitations. We have discussed and compared various steganography tools. We have developed a new steganography technique and we are currently on the process of testing it and experimenting with it.

References

1. W. Bender, D. Gruhl, N. Morimoto, and A. Lu. Techniques for data hiding. In IBM Systems Journal, Vol. 35, Nos. 3-4, pages 313-336, February 1996.
2. aura, T., "Practical Invisibility in digital communication, in information hiding" first international workshop, proceedings, vol. 1174 of lecture notes in computer science , Springer, 1996, pp. 265-278.
3. Johnson, N.F. and S. Jajodia. "Exploring Steganography: Seeing the Unseen." IEEE Computer Mag., February 1998.
4. Westfeld, A., and G. Wolf, Steganography in a Video conferencing system, in proceedings of the second international workshop on information hiding, vol. 1525 of lecture notes in computer science, Springer, 1998. pp. 32-47.
5. Pan ,H.K., Y.Y., Chen, and Y.C., Tseng, "[A Secure Data Hiding Scheme for Two-Color Images](#)", Proc. Fifth IEEE Symp. Computers and Comm., IEEE Press, Piscataway, N.J., 2000.
6. Westfeld, A., "[F5-A Steganographic Algorithm: High Capacity Despite Better Steganalysis](#)", 4th International Workshop on Information Hiding ,2001.
7. Curran, K. and K. Bailey, "[An evaluation of image-based steganography methods](#)" International Journal of Digital Evidence ,Vol – 2 , issue 2 , 2003.
8. Kessler, G. "[An Overview of Steganography for the Computer Forensics Examiner](#) ", Computer & Digital Forensics Program, Champlain College, Burlington, Vermont, February 2004
9. Rabah, K," [Steganography : The Art of Hiding Data](#)" Information Technology Journal ,Vol 3 No.3, 2004 , pp. 245-269 .
10. Bennett, K. "[Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text](#)" Technical Report, Center for Education and Research in Information Assurance and Security (CERIAS), 2004.
11. J.M. Rodrigues, J.R. Rios and W. Puech." [SSB-4 System of Steganography using Bit 4](#)". Proc. 5th International Workshop on Image Analysis for Multimedia Interactive Services, (WIAMIS'04), Lisboa, Portugal, April 2004.
12. R. Chandramouli, M. Kharrazi, N. Memon, "[Image Steganography and Steganalysis: Concepts and Practice](#) ", International Workshop on Digital Watermarking, Seoul, October 2004.
13. Westfield, A., and A. Pfitzmann. "Attacks on Steganographic Systems - Breaking the Steganographic Utilities EzStego, Jsteg, Steganos, and S-Tools - and Some Lessons Learned," Lecture Notes in Computer Science, 1768: 61-75 (2000).
14. Fridrich ,J, M. Goljan, and R. Du, "Detecting lsb steganography in color and grayscale images," IEEE Multimedia Special Issue on Security, pp. 22–28, October- November 2001.
15. Avcibas, I. , N. Memon, and B. sankur, "Steganalysis using image quality metrics." Security and Watermarking of Multimedia Contents, San Jose, Ca., Feruary 2001.
16. Fridrich, J., M. Goljan, , H. Dorin , "Steganalysis of JPEG Images: Breaking the F5 Algorithm". Information Hiding 2002, pp. 310-323.
17. Kessler, G. "An Overview of Steganography for the Computer Forensics Examiner " , Computer & Digital Forensics Program, Champlain College, Burlington, Vermont, February 2004
18. OutGuess Web Site ."Steganography Detection with Stegdetect ".URL: <http://www.outguess.org/detection.php>.Last accessed: 2004
19. Fridrich, J., Goljan, M., and Hogeia, D. " Attacking the OutGuess". In: Proceedings of the ACM Workshop on Multimedia and Security 2002, Juan-les-Pins, France, December 2002.