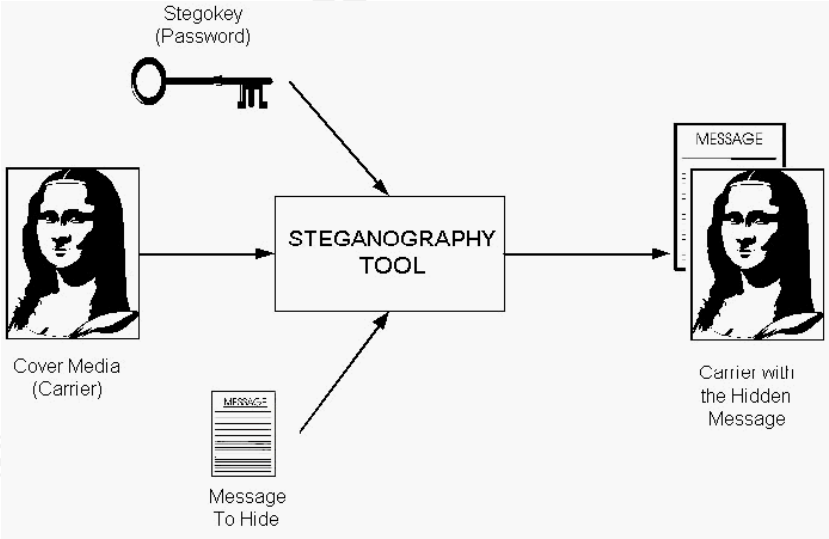
Global Information Assurance Certification Paper

Jim Bartel 1.2d

STEGANALYSIS
An Overview

In the field of research known as Information Hiding, there is a little known area called Steganography. Steganography, which literally means covered writing, has been around for very long time and examples of its use can be found as far back as the antiquities. It is a discipline that borrows much of its terminology and methodology from its cousin, cryptography. While Steganography and cryptography may parallel each other in many areas, their respective goals are different: the goal of cryptography is to protect the contents of a message while the goal of steganography is to hide the fact that the message even exists. Like cryptography, Steganography has an area of study that examines ways to detect and defeat it. It is known as steganalysis. Beginning with a quick review of steganography, this paper takes a brief look at the terminology associated with steganalysis, the parallels to cryptanalysis, as well as a few steganography tools and their signatures. Although there are many steganographic tools available for use with various mediums of communication, this paper limits itself to steganalysis of digital imagery.

Before there can be steganalysis, there has to be steganography. Below is an illustration of the steganographic model. Using a steganography application, a message is combined with cover media or a carrier to create a stego-medium or stego-carrier. This may require what is called a stego-key or secret password that is in addition to the original message. Each of these elements can be used in different ways to perform steganalysis attacks.



Cover medium + embedded message + stegokey = stego-medium

Figure 1. Steganography Model

Figure 1 illustrates how using a steganography tool, a message and a carrier are combined, along with a password, to create a stego-carrier.[2][3]

Next, we look at the terminology associated with steganalysis. As was mentioned earlier, Steganography borrows much of its terminology from cryptography. Take for example, one who uses cryptanalysis methods to decipher encrypted information is known as a cryptanalyst. One who uses steganalysis methods to detect and defeat hidden information is known as a steganalyst. Shown below are a few more examples that illustrate the parallels between cryptography and steganography.

Cryptography/Cryptanalysis	Steganography/Steganalysis
Combining plaintext and a cryptographic tool yields cipher-text.	Combining text with a steganographic tool yields a stego-object.
Plaintext and cipher-text are utilized when performing cryptanalysis.	The carrier, stego-object and hidden message may be used when performing steganalysis.
A cipher-text only attack is where only the cipher-text is known to the analyst.	A stego-only attack is where only the stego-object is available for attack.
A chosen plain-text attack is where a portion of the plain-text, which corresponds to a portion of the cipher-text, are available for analysis.	A chosen stego attack is where the Steganography tool (algorithm) and the stego object are known.

Here are other attacks available to the steganalyst:

- **Known cover attack.** The "original" cover-object and stego-object are both available.
- **Known message attack.** At some point, the hidden message may become known to the attacker. Analyzing the stego-object for patterns that correspond to the hidden message may be beneficial for future attacks against that system. Even with the message, this may be very difficult and may even be considered equivalent to the stego-only attack.
- **Chosen message attack.** The steganalyst generates a stego-object from some Steganography tool or algorithm from a chosen message. The goal in this attack is to determine corresponding patterns in the stego-object that may point to the use of specific Steganography tools or algorithms. (This is the most powerful attack.)
- **Known stego attack.** The Steganography algorithm (tool) is known and both the original and stego-objects are available.

Detection

In order to use imagery to hide information, stego tools must first manipulate the original image. This manipulation causes distortion in the original image's properties, which can

be difficult to detect visually. While it may not be readily observed by the human eye, this distortion can be detected by other means and eventually used to as a signature to determine the tool or hidden message.

One signature that can be easily detected is exaggerated noise in an image. This is a common characteristic of tools that use the LSB to carry information when applied to 8-bit color images. If the palette of the image is not manipulated as part of the process, many of the pixels will exhibit a color shift as a result of the LSB change and if they are not already adjacent to a similar color, they will show up as noise. It is for this reason that 256 gray-scale images are recommended by makers of steganography software for use as a cover over color images. Gray-scale images are very good covers because the shades gradually change from color entry to color entry in the palette. An example of the noise problem appears in the figure below. The image on the left is the original carrier, the center image shows the results of trying to hide a 9k text file, and the final image on the right is a stego image with the same 9k text file and shows what results are possible with gray-scale images.

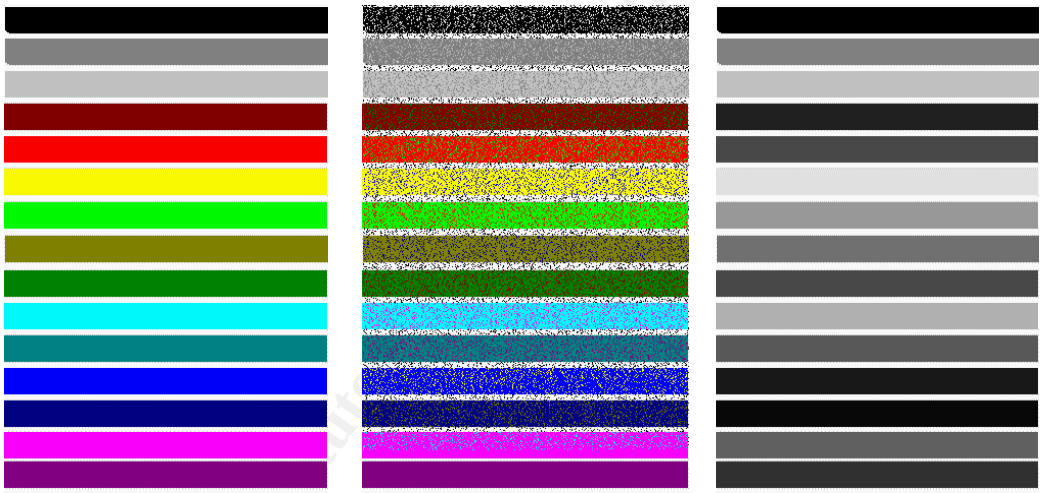


Fig. 2. Original 8-bit cover image (left), and the 8-bit stego image (center), and the same message using a gray-scale image as a carrier (right). Created using *Hide and Seek v4.1*

Another visual clue to look for is padding of the image. In version 4.1 of Hide and Seek, the user is limited to use carrier images that are 320x480 pixels in size and 256 colors. If the image is smaller than the required size, Hide and Seek will add pixels to the image (padding) to get the required dimension (see Fig 3.). Conversely, if the image is too large, Hide and Seek will crop the image as needed. In version 5.0, the user is not limited to a single size. Instead, the images must meet specific dimensional requirements: 320x200, 320x400, 320x480, 640x400 and 1024x768. Version 5.0 still utilizes padding in some circumstances to satisfy its dimensional requirements. Later versions do not have the same constraints on size as with previous versions.

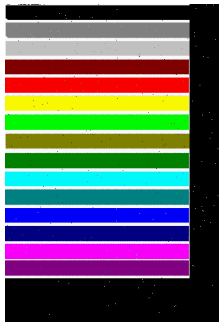
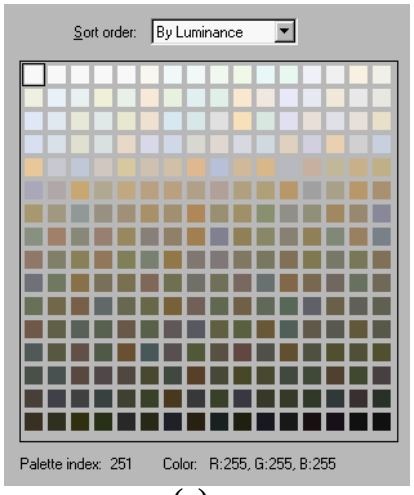


Fig. 3. Example of padding method
Created using *Hide and Seek v4.1*

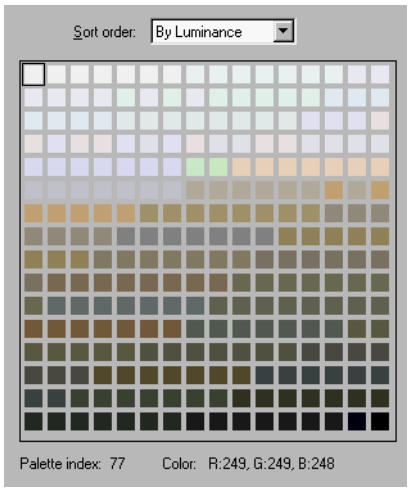
While LSB encoding is very popular due to its simplicity and there are many tools using this technique, it is extremely vulnerable to attack. Manipulation, filtering, cropping, noise, rotation, will all render the message unusable. For example, cropping a single row of pixels from the stego-image in fig. 2 made the message irretrievable.

Detection beyond simple visual observation requires more detailed analysis. Known cover and known stego attacks can be used to look for signatures. These methods involve using many different images and comparing the originals to the stego-images. The intent is to determine the existence of a signature for a specific tool that can later be used when examining real-world images. Note that in some cases, a recurring predictable pattern is not produced even though there are noticeable differences between the two images.[3]



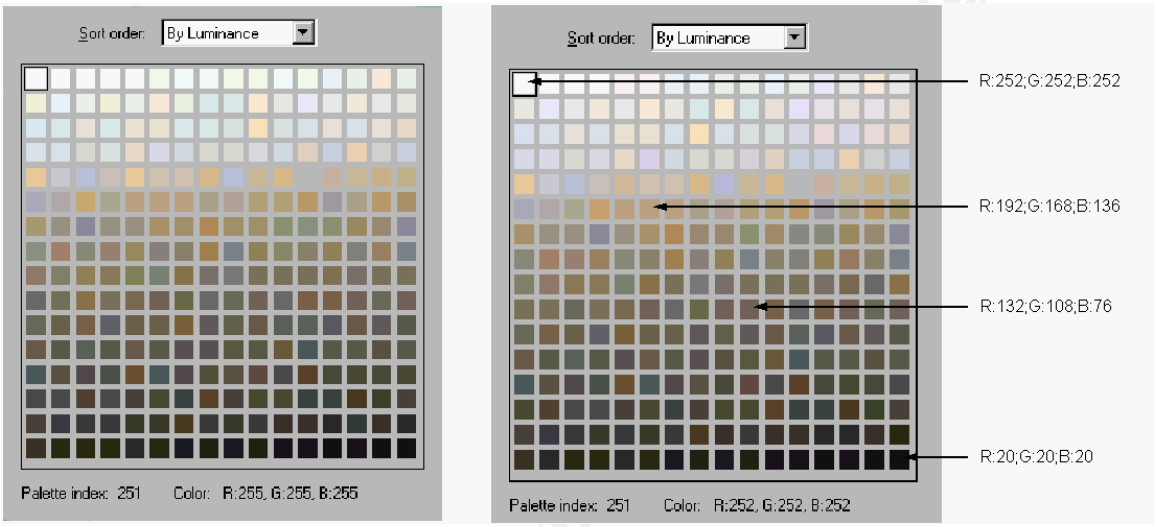
(a)

As mentioned earlier, some tools manipulate the color palette to hide information. One example of this is the addition (padding) of black palette entries in a 256-color photo. Since most 256-color photographs will generally have entries that are *near* black, the existence of an unusual number of black entries in a palette would be an indication of palette manipulation.[2] Palette manipulation is very effective at hiding visual distortions and tools such as S-tools and Mandelsteg produce very good "on paper" results but under close examination, they can produce unnatural patterns between color values.[1] Figure (b) to the right illustrates the palette manipulation as a result of using S-Tools. Figure (a) is the original palette. Another method of manipulation reduces the actual number of colors found in the color palette. For example, when Syscop processes a gray-scale image, it reduces the total number of colors to only the colors needed in the final stego-image. The result of using this tool on a GIF image which has a color index of 256, will be that the index will now show only those few colors used by Syscop. Even if a preferred gray-scale image is used as a carrier, it still has a 256 color index (FF or 255) for white and (0) for black, and the result will be a similar lack of colors in the color index.



(b)

Some tools have characteristics that are unique among stego-tools. In versions 4.1 and 5.0 of *Hide and Seek*, the color palettes have unique characteristics that have yet to appear anywhere else. For example, all color palette entries are divisible by 4 for all bit values and the maximum is 252.[1]



a b

Fig 4. Hide and Seek palette signatures. (a) original palette (b) manipulated palette. Note that all entries in (b) are divisible by 4.

Disabling information.

Detecting the existence of hidden information defeats Steganography's goal of imperceptibility and we've seen how steganalysis can be applied to achieve this. However, other methods exist that can produce results that are much more difficult to detect. Alternatives to passive attacks that focus on detection are available, and they are much easier to use. These methods fall into a category called active attacks and their purpose is to destroy or disable the information rather than detect it. The advantage of using active over passive attacks is that they are easier to implement since the presence of steganography is not a requirement for their use. Listed below are some of the methods used in active attacks and their descriptions.

Blur – Smoothes transitions and decreases contrast by averaging the pixels next to the hard edges of defined lines and areas where there are significant color transitions.
Noise – Random noise inserts random colored pixels to an image. Uniform noise inserts pixels and colors that closely resemble the original pixels.
Noise reduction – Reduces noise by adjusting colors and averaging pixel values.
Sharpen – Opposite of blur. Increases contrast between adjacent pixels where there are significant color contrasts, usually at the edge of objects.

Rotate – Moves an image around its center point in a given plane.
Resample – Resampling involves an interpolation process to minimize the "raggedness" normally associated with expanding an image.
Soften – Applies a uniform blur to an image to smooth edges and reduce contrasts. Causes less distortion than blurring.[1]

So how does Steganography hold up against the active attacks? In [1], the authors subjected several well-known stego and watermarking tools to tests to determine the survivability of hidden information when subjected to various image processes and conversions. The results were that few steganography tools survived even minor processing or conversion to JPEG. Watermarking tools fared much better with varied results.

What's on the Horizon?

When compared to cryptography/cryptanalysis, little is known about Steganography/steganalysis and for that matter, Information Hiding in general. This is rapidly changing however. In 1992 there were only two papers written on watermarking and the first academic conference on Information Hiding didn't occur until 1996. In 1998, there were 103 papers on watermarking and the fourth academic conference will take place in April of this year. As both the legal and illegal uses of Steganography increase, industry and government have become involved in supporting and funding steganalysis research to develop automated blind detection steganalysis tools that can go out across the Web and seek out steganography. (To date, there has been some success with the use of JPG files). This paper has taken a look at some of the manual tools and methodologies of one small area of steganalysis that are currently in use. It will probably not be long before the tools and methods discussed here are considered just as archaic as those methods used in the antiquities.

Comment

Anyone who is doing research for their SANS project on Steganography, or anyone else who just wants more information on the subject and are not finding what they want on line, I strongly recommend the two books: ***Information Hiding: Steganography and Watermarking – Attacks and Countermeasures*** and ***Information Hiding Techniques for Steganography and Digital Watermarking***. Both are full of detailed information that I was not able to locate on the Web and they are listed as references for this paper. Like most good reference books, they are not cheap but both were excellent sources.

REFERENCES

1. Johnson, N. F., Z. Duric, and S. Jajodia, *Information Hiding: Steganography and Watermarking – Attacks and Countermeasures*, Boston, Massachusetts: Kluwer Academic Publishers, 2001.
2. Katzenbeisser, S., F. A. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, Boston Massachusetts: Artech House, 2000.
3. Johnson, N.F. and S. Jajodia. *Steganalysis of Images Created Using Current Steganography Software*, Information Hiding: Second International Workshop, 1998, URL:http://ise.gmu.edu/~njohnson/ihws98/jjgmu.html
4. Lin, E.T., and E. Delp. *A Review of Data Hiding in Digital Images*, URL: ftp://skynet.ecn.purdue.edu/pub/dist/dellp/pics99-stego/paper.pdf.
5. Johnson, N.F., and S. Jajodia. *Steganalysis: The Investigation of Hidden Information*. 1998, URL: http://ise.gmu.edu/~njohnson/pub/it98jjgmu.ps
6. Johnson, N.F. and S. Jajodia. *Exploring Steganography: Seeing the Unseen*. 1998, URL:http://www.jjtc.com/pub/it98a.htm.
7. Petitcolas, F.A., R. Anderson and M. Kuhn, *Information Hiding-A Survey*, 1999, URL: http://debut.cis.nctu.edu.tw/~yklee/Research/datahiding/REF/ProcIEEE1999-7-1.pdf
8. Sellers, D., *An Introduction to Steganography*, URL:http://www.cs.uct.ac.za/courses/cs400w/nis/papers99/dsellars/stego.html.
9. McCullagh, D., *Secret Messages Come in .WAVS*. URL: http://www.wired.com/news/politics/0,1283,41861,00.html.

Last Updated: July 22nd, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Trenton SEC401	Trenton, NJ	Aug 21, 2017 - Aug 26, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor