Global Information Assurance Certification Paper

Francis Ansuh
GSEC Version 1.4

Steganography: Not Just a Tool For The Bad Guys

Abstract

Recent advances in technology has seen an explosion in the use of computers and the World Wide Web as a means of trading and general exchange of information. Video, audio and other forms of art are now becoming available in digital form and it has become very necessary to protect authors from copyright infringements and, theft of intellectual property as technology allows perfect copies to be made. Steganography and digital watermarking lends us a hand in working towards the solution to these problems.

This work gives a brief overview of steganography and shows how easily it can be performed by almost everyone. It also portrays digital watermarking as a subset of steganography and its potentially positive applications in modern day technology. A brief mention is also made of the implications of the use of steganography by people with ill intentions. The paper concludes on the need for the development of more robust watermarking and steganalysis techniques.

Introduction

Steganography is the art and science of hiding information in ways that prevent the detection of the hidden message. The term as borrowed from Greek literally means, "Covered writing". In contrast to cryptography, which scrambles a message so it cannot be understood steganography hides the message so it cannot be seen. A message in ciphertext may arouse suspicion on the part of the recipient but an "invisible " message created with steganographic metho ds will not. Steganography gives real meaning to the notion of "security through obscurity". In addition to hiding the message, most steganographic methods also encrypt the message so even if the presence of the message is detected you will still need a pa ssphrase to decipher the message. Steganography adds an extra layer of security and is thus complementary to cryptography.

It is an ancient art that has been performed throughout history in a variety of ways. For example, around 440 BC Histiaeus a Roman general shaved the head of his most trusted slave and tattooed it with a message. After the hair had grown back and the message was covered, the slave was sent to the message's intended recipient who subsequently shaved the head to reveal the secret messag e [1]. Herodotus also tells us of another incident when a message was hidden on wax -covered tablets (the then writing medium). Demeratus, a Greek notified Sparta of an imminent invasion by Xerxes, King of Persia. To avoid any suspicion, he scrapped the wax off the tablet and wrote the message on the underlying wood. The tablet looked exactly like a blank one and did not arouse any suspicion on inspection.

The use of steganography flourished again in the twentieth century. Early in the century during the Bo er war, Lord Baden -Powell (founder of the Boy Scout

Movement) was employed by the British to mark positions of Boer artillery bases. In order to avoid suspicion if he was caught, he worked his maps into drawings of butterflies. This appeared innocent to the casual observer but certain markings on the wings of the butterflies were actually positions of Boer military installations [4].

During the 2nd world war, there was an intensive use of steganography and experimentation of different steganographic methods. Invisible inks offered a common form of writing. Common sources for invisible inks are fruit juices, vinegar, milk and urine, all of which when heated darkens. With an invisible ink, a seemingly innocent letter could contain a very different message written between the lines [2]. Some invisible inks were also made of chemicals and the messages were developed like the processing of a photographic film in order to retrieve it.

Some other methods of concealing messages include Microdots, Character shifting arrangements, Digital signatures, Covert channels and Spread spectrum communications.

As Johnson [4] puts it, "microdots are photographs the size of a printed period having the clarity of standard-sized typewritten pages". The Germans used microdots during WWII. The first was found masquerading as a period on a typed envelope carried by a German agent in 1941. J Edgar Hoover the FBI director described this technology as "the enemy's masterpiece of espionage" [4], [18].

A covert channel is described by Lampson [19] as a communication channel that is neither designed nor intended to transfer information. Unused space in TCP/IP packet headers can be used to transmit information.
Unused space in operating systems can also be taken advantage of to store information covertly. For example drives formatted as FAT 16 under Windows 95 operating system (MS-DOS compatible) without compression typically stores data in 32 KB clusters. This means that even if a file is 1KB in size, the resulting 31 KB will be wasted because of the way storage space is allocated. The "extra" space can be used to store information without showing up in the directory [8], [14].

Modern day steganography has taken on a whole new meaning with the advent of computers and the Internet. The ease with which one can perform steganography and the existence of the World Wide Web raises a lot of questions for security professionals and law enforcement agencies. There is little wonder that many people see steganography as a dangerous tool for terrorists and other people with ill intentions to perform illicit acts [3]. Could terrorists be using steganography as a means of communication? Could it be used as a tool for corporate espionage by disgruntled staff? Could our loved ones use it as an aid to cheat on us from our home PCs? I leave the answer to this last question to your imagination.
Supposing the answer to all the above questions is yes, is there a possible means of detecting and controlling these activities?

Modern Steganography

There are 2 basic components needed to perform steganography, these are the secret message and the cover medium or container. The message may be plaintext or ciphertext or anything that can be embedded in a bit stream. Should a

"ciphermessage" be used one will als o need a stego-key (compare with passphrase in cryptography). Stego-meduim is the result of embedding the message in the cover medium [12] (**figure 1).**

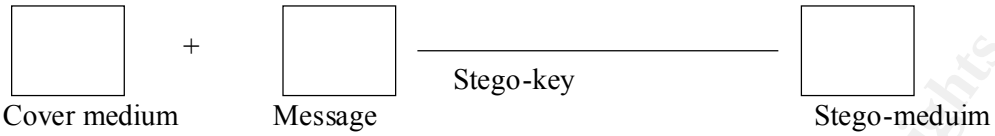

Figure 1.An illustration of the components of steganography. Please note that the stego-key may not always be needed.

Once a cover medium is selected, a technique for embedding or hiding the message must be decided on. Some of the common approaches of hiding information in digital media include:

- Least significant bit insertion
- Masking and filtering
- Algorithms and transformations.

Each of the above techniques can be carried out with varying degrees of success to different digital media. "Masking and filtering" and "Algorithms and Transformations" are sometimes referred to as techniques of the Transform Domain.

Please note that steganography can be pe rformed using different forms of digital media such as sound files, text files, image files. The rest of this work will focus mainly on image files as the cover medium.

Techniques in Concealment

Least Significant Bit (LSB) Insertion

This is the simples t form of embedding message in a cover file. The least significant bits of the host file are replaced with data. Given a message data such as 11010010, the most significant bits (MSB) are those that lie to the far left and the least significant bits (LSB) lie to the far right. Making use of the limitations of human vision, the last one or two LSBs can be changed without having a significant change in colour. Changing the most significant bits however will have a big impact in colour difference. Unfortunatel y, this method is very vulnerable to even the slightest of image manipulations. Converting an image from a format like GIF or BMP to JPEG and then back could destroy the information hidden in the LSBs. For a brief look at file compression, see footnote. [1]

For a 24-bit image file, each pixel is represented by 3 bytes (24 bits) making up the image's raster data. Thus you can potentially store 3 bits in each pixel (if only 1 of the LSBs is used) or 6 bits if the 2 LSBs are used. The resulting stego -image will still

[1] FILE COMPRESSION: Two kinds of compression are lossless and lossy. Lossy compression as typified by JPEG (Joi nt Photographic Experts Group) format files offer high compression rates but may not maintain the original integrity of the image. The compression algorithm "losses" unnecessary image data and provides a close approximation but not an exact duplicate of th e original. Lossless compression maintains the original image data but unfortunately does not offer high compression rates as lossy. Examples of lossless compression formats are CompuServe's GIF (Graphics Interchange Format) and Microsoft's BMP (Bitmap) [4].

look the same to the human eye as the original cover image. For example, if the binary value of our message is 11010010, it can be hidden in 3 pixels (assuming no compression). The original raster data for 3 pixels (9 bytes) may be:
(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)
Inserting our message in the LSBs of the 3 pixels, starting from the top left byte will result in the following
(00100111 11101001 11001000)
(00100111 11001000 1110100**0**)
(11001000 00100111 1110100**0**)
The 2 underlined bits are the only ones that actually changed in the 8 bytes used. The resultant stego-image will look the same to the human eye as the original cover image [4], [5].

8-bit image files are not as forgiving when it comes to LSB insertions. Each pixel is represented by 1 byte (8 bits). Looking at our example above you will notice that there will not be a lot of bits to play around with without impacting a significant colour change. The success in using 8-bit image files for LSB insertions lie in the careful selection of the cover-image.

Some of the tools that make use of the LSB technique for embedding data are S-tools (Steganography-tools), EzStego, White Noise Storm, Steganos, StegoDos, etc. (to mention but a few.

Masking and Filtering
This technology hides information by marking the cover image in a similar way to paper watermarks. The message is embedded in significant areas of the cover image in such a way that the hidden message becomes an integral part of it. Digital watermarking tools employ this technique.

Algorithms and Transformations
These technologies also hide messages in the significant areas of the cover image. It is thus more robust than the LSB insertion method.
JPEG images use the discrete cosine transform (DCT) to achieve compression. In DCT algorithm the cosine values cannot be calculated exactly. Rounding errors or approximations may be introduced into the final result as calculations are repeated using limited precision numbers. Variances between the original data values and the restored data values depend on the method used to calculate DCT [5]. This explains why JPEG format is a lossy compression.
In addition to DCT, fast Fourier transformations and wavelet transformations can also be used to process images. Image properties such as luminance can also be made use of.

Patchwork algorithm and other similar techniques use redundant pattern encoding or spread spectrum method to scatter hidden information throughout the cover image. Patchwork algorithm selects random pairs of pixels, makes the brighter pixels brighter and the duller pixels duller. The contrast change in this pixel subset encodes one bit [4], [6]. In redundant pattern encoding there is a trade-off between message size and robustness. For example, a small message may be repeatedly painted across the cover

image and there is a high probability that the message (watermark) can still be read after the image has been cropped. A large message on the other hand may be painted only once across the cover image and will therefore be vulnerable to cropping (**figure 2).**

(A) Cover image has been painted repeatedly with a small -sized image (watermark) Note that the image "Francis K Ansuh" can still be read if the cover image is cropped.

(B) Cover ima ge has been painted only once with a big -sized image (watermark) of a red rose.

Figure 2. Above illustrates the trade -off between the size of the hidden message and robustness. Please note that for demonstration purposes only, the luminance of the hidden message has been manipulated to make it visible.

Watermarking

Digital watermarking like steganography describes the techniques that are used to secretly convey information by embedding into digital media. Watermarking may be

perceptible or imperceptible to the human eye. Visible watermarking can be compared to traditional paper watermarks and logos seen on TV broadcast stations. Visible watermarks are technically not considered as steganography (hence no mention will be made of it in the rest of the paper).
Digital watermarking is considered a subset of steganography in that some of the techniques used in embedding messages are the same. These techniques, mostly of the transform domain are more robust to attacks such as cropping, compression and some image processing where least significant bits are changed. The only difference between digital watermarking and steganography is primarily one of intent. In digital watermarking the cover is the object of communication whereas in steganography the hidden message is the object of communication [12]. Steganography typically relates to covert point-to-point communication between two parties and therefore does not necessarily need the robustness required in digital watermarking. Digital watermarking however needs an additional notion of robustness against manipulations that may attempt to remove it.
The hidden information in watermarking may typically be registration of ownership for copyright or a means of tracking information that has been distributed. The hidden message could also be an ad encouraging viewers to contact the owner if they like what they see [7].
It may be used to indicate the copy status of data. An example is the case of DVD systems where copy information is embedded as a watermark. Compliant playe rs will not playback or copy data that may carry a "copy never" watermark. Data that carries "copy once" watermark may be copied only once and no further copy will be allowed from that copy.
Digital watermarking may also be used for fingerprinting. Fingerp rinting is the embedding of serial codes that distinguish between distributed data sets. Embedded data in fingerprinting may be a unique code out of a series of codes that identifies the recipient of the data or a unique code that identifies the copyright owner {[8], pp100-105}. Thus, watermarking helps in tracking copyright violators while fingerprinting will help in their conviction.
One can understand from the above applications of watermarking, why there needs to be that additional requirement of robust ness.

Some Tools Of The Trade

The list of software packages used in performing staganographic manipulations is enormous and is growing everyday. Amongst these are Hide and Seek, SstegoDos, White Noise Storm, S-Tools for windows, Jpeg-Jsteg, Stealth, etc. Almost all these tools are freeware and can be downloaded from the Internet (a good starting place will be http://members.tripod.com/steganography/stego/software.html).
In [9], Johnson evaluates some of these tools and concludes S -Tools for windows v3 as the most versatile of those packages tested. I will use S -Tools v4 to demonstrate the ease with which steganography is performed in the next few paragraphs.

S-Tools for Windows

S-Tools 4 by Andy Brown [10] includes programs that will hide information in BMP and GIF image files, and audio WAV files. It supports 24 -bit image files and also has an array of encryption routines (IDEA, DES, 3DES and MDC) with many options. It has the option to compress the data to be hidden or store it in raw mode. To solve the problem of identical sets of data encrypting the same, S -Tools prepends some random garbage on to the front of each data. The random garbage together with the data is then encrypted, using the passphrase that is chosen, to generate the key.

S-Tools applies the LSB technique discussed earlier in hiding information. Instead of just spreading the information to be hidden in linear fashion across the available bits, it uses a cryptographically strong pseudo -random number generator (from the chosen passphrase) to determine the position of the next bit to use. For instance, if there are 100 bits available for hiding and you wanted to hide 10 bits, S -Tools will choose any random 10 bits (depending on the chosen passphrase). For example, 83,92,15,20,2,53,99,80,21,30 may be the sequence rather than 0,1,2,3,4,5,6,7,8,9[S -Tools documentation][10].

How to hide and Reveal files with S -Tools4

NB: I will assume that S-tools for windows is already installed on your PC.

After opening up the application, the cover image is first selected from windows file manager or windows explorer and dragged to the S -Tools window. The file to be hidden is also located and dragged from the explorer windo w and dropped on the opened cover file in the S-Tools window. It is as simple as that. You will then be prompted to choose the type of encryption (between IDEA, DES, 3DES and MDC) and the passphrase of your choice. An action window is displayed to show the status of the process. S-Tools does a good job of displaying the size of file that can be hidden in an opened cover file in order not to waste your time. After hiding the message the stego-image is displayed alongside the cover image for comparison (figure 3).

To save the stego-image, you right-click on it and choose "Save" or "Save as". If "save" is chosen, the file will be saved under the name, "hidden". Either way, you have to make sure the file name ends in BMP or GIF. S -Tools looks at this part of t he file name to decide whether to save the picture as a GIF or BMP file.

To retrieve a message from an image file that S -Tools is displaying, you right -click on the image and select "Reveal" from the context menu that appears. You will then be prompted for the passphrase and the type of encryption algorithm used. The reveal task appears in the actions window for you to check on its progress. A "Revealed Archive" window appears showing the name or names of available files. To read or open a file, you first select it from the archive window and right -click on it. You then choose "Save as" from the context menu that appears. The saved file can then be manipulated.

(A) Original / Cover Image

(B) Stego-Image containing embedded message (in this case, the Nursery rhyme "Mary Had A Little Lamb").

Figure 3. Images A and B (above) illustrates how innocent a stego -image may look compared to the original cover image. **Note**: To check the presence of the hidden message, use S-Tools4. Passphrase used =6767
Encryption algorithm used = IDEA

Applications Of Information hiding

Digital watermarking as already mentioned above has its potential uses in the law courts to protect intellectual property from copyright theft.
An emerging technique in the healthcare indus try is hiding of messages in DNA sequences [11]. This could be used to protect intellectual property in medicine and the biotechnology industry. For further reading on DNA -Based Steganography or Biologically Inspired DNA -Based Technologies you are referred to http://adsr13.mssm.edu/domains/dept/facultyInfo.epl?objname=physbio&user=bancrc_01.

The Digital Imaging and Communication in Medicine (DICOM) standard us ed in the healthcare industry usually separates image data from caption such as the name of patient, date of birth, and physician. The problem faced here is that sometimes the link between patient and image is lost thus; embedding the details of the patien t in the image file could be a useful safety measure [8].

Information hiding can also be used for keeping the privacy of research subjects where, medical records or census returns are de -identified for processing by researchers.

Discussion

As already seen from the above paragraphs, steganography is very easy to perform and has really gone mainstream these days. The question of whether terrorists have been using covert means of passing information to their colleagues by using the Internet has been circul ating for a while [3]. A lot of questions were raised about this following the September 11 th atrocities at New York.
Fear of the use of steganography by terrorists groups became apparent when the US government requested that the media refrain from broadc asting unreviewed video originating from the Middle East.

Unusual patterns usually stand out on files after steganographic manipulations. A stego-image may look innocent to the casual observer but may reveal the presence of "signatures" upon critical anal ysis. These "signatures" reveal the presence of hidden information and defeats the aim of steganography. An example of unique signatures of steganographic tools as applied to images can be found in [12].

There is the need for more research work in the aut omation of "signature" detectors that can crawl the World Wide Web for steganographic material. Such tools are very promising for future work in steganalysis and verifying watermarks [8].

Filters can be applied to Internet firewalls to detect packets that have information in supposed unused or reserved space.

Information hidden in text by line spacing may be difficult to detect by the casual observer but opening such a file with a common word processor will reveal any appended spaces and "invisible" chara cters [8].

Conclusion

Steganography has recently been given a new breadth of life as various governments are making moves to restrict the availability of encryption services. This has led people to study methods by which private messages can be embedded in seemingly innocuous cover messages. However, the main driving force in current academic research work in information hiding is the concern over protecting copyright. Recent advances in technology have made it very easy to make perfect copies of digital audio and video. This may lead to unauthorised large -scale copying and has become a great concern for the music, film, book and software publishing industries. Information hiding techniques as applied in digital watermarking and fingerprinting seem promising in solving these problems.
Even though there are various attacks that can be staged on watermarks there are also countermeasures to these attacks. Through the use of tools that test the strength and survivability of watermarks, limitations of current techniques have been understood and also new techniques are under development [13]. More research work is needed in future to automate steganalysis techniques, as it will be very useful to law enforcement authorities in computer forensics, information secu rity professionals and digital traffic analysis.

References

1. Herodotus, "The Histories", London, England: J. M. Dent & Sons, Ltd, 1992.

2. Zim, H. S. "Codes and Secret Writing ", New York: William Morrow, 1948.

3. Various articles on "Steganography and Terro rist activities" http://www.steganos.com/en/bin_laden_1.htm

4. Sellars, Duncan. "An Introduction to Steganography". http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html

5. Johnson, Neil F., Jajodia, Sushil "Exploring Steganography: Seeing the Unseen". http://www.jjtc.com/pub/r2026.pdf

6. Petitcolas, Fabien AP., Anderson, Ross J., Kuhn, Markus, G. "Attacks on Copyright Marking systems". http://www.cl.cam.ac.uk/~fapp2/publications/ih98 -attacks.pdf

7. Digimarc Corporation, "Di gimarc Watermarking Guide". http://www.digimarc.com/support/cswater.htm

8. Katzenbeisser, Stefan. , Petitcolas, Fabien AP. " Information Hiding Techniques for Steganography and Digital Watermarking" . Boston. London: Artech House, January 2000.

9. Johnson, Neil F. "Steganography". http://www.jjtc.com/stegdoc/index2.html

10. Brown, Andrew. S-Tools. http://members.tripod.com/steganography/stego/s -tools4.html

11. Clelland, T., Risca, V., Bancroft, C. "Hiding Messages in DNA Microdots" http://fulcrum.physbio.mssm.edu/cb /NATURE.pdf

12. Johnson, Neil F., Jajodia, Sushil. "Steganalysis of Images Created Using Current Steganography software". http://www.jjtc.com/ihws98/jjgmu.html

13. Johnson, Neil F. "An Introduction to Water mark Recovery from Images". http://www.jjtc.com/pub/nfjidr99.pdf

14. Johnson, Neil F., Jajodia, Sushil. "Steganalysis: The Investigation of Hidden Information. http://www.simovits.com/archive/it98jjgmu.pdf

15. Johnson, Neil F. " In Search of the Right Image: Recognition and Tracking of Images in Image Databases, Collections, and The Internet". http://www.jjtc.com/pub/csis_tr_99_05_nfj/

16. MSSM homepage (for further reading on DNA -based steganography) http://adsr13.mssm.edu/domains/dept/facultyIn fo.epl?objname=physbio&user =bancrc01

17. Steganography Software. http://members.tripod.com/steganography/stego/software.html

18. Kahn, David. "The Code Breakers". New York: The Macmilla n Company, 1967

19. Lampson, B. W., "A Note on the Confinement Problem". Communications of the ACM. Volume 16, (Number 10, 1973): pp. 613 -615

Last Updated: August 17th, 2017

Upcoming Training

Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
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
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
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 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 London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS New York SEC401**	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
SANS October Singapore 2017	Singapore, Singapore	Oct 09, 2017 - Oct 28, 2017	Live Event
SANS Phoenix-Mesa 2017	Mesa, AZ	Oct 09, 2017 - Oct 14, 2017	Live Event
SANS Tysons Corner Fall 2017	McLean, VA	Oct 14, 2017 - Oct 21, 2017	Live Event
CCB Private SEC401 Oct 17	Brussels, Belgium	Oct 16, 2017 - Oct 21, 2017	
SANS Tokyo Autumn 2017	Tokyo, Japan	Oct 16, 2017 - Oct 28, 2017	Live Event
Community SANS New Orleans SEC401	New Orleans, LA	Oct 23, 2017 - Oct 28, 2017	Community SANS
Community SANS Omaha SEC401	Omaha, NE	Oct 23, 2017 - Oct 28, 2017	Community SANS
SANS vLive - SEC401: Security Essentials Bootcamp Style	SEC401 - 201710,	Oct 30, 2017 - Dec 06, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event