Ben Richter

GSEC Certification
Version 1.3
Original Practical Submission
SANS Cyber Defense Initiative West Track – San Francisco
February 7, 2002

The security of electronic data and intellectual property

The globalization of the economy and greater labor productivity resulted from advances in information technology. Today people, businesses, and governments are able to communicate information instantaneously to anywhere on the globe. This information could be a personal, an important trade secret giving your firm a competitive advantage, or top-secret military information, what would result if this information fell into the wrong hands? Invariably in this digital age it is invariably easy to make perfect copies of digital media such as software and music. How can corporations and authors protect their intellectual property from piracy? The electronic transmission of information is no longer a convenience, but a way of conducting day-to-day business and living our lives; ensuring that these electronic transactions are secure and privy to only those who need access to them is of the utmost necessity.

There are two primary options to decide between in which the problem of insecurity can be addressed. Encryption, where you would encode the data, so that even if the enemy gets a hold of the files containing the secret information they would not be able to decipher the message. The other option is steganography, in which data is encapsulated within other files, making a message appear innocuous to the point that the enemy has no suspicion that there is information hidden within the message. In actuality best practice would be to use both encryption and steganography to conceal data, that way if intercepted and the steganographic data revealed, the "enemy" would find random looking (enciphered) data, and believe that a secret was never sent. We will examine the topic of steganography in this document; exploring its history, some real world usage, an application system of steganography in digital watermarking, and some software tools that can be used to both extract and hide data.

Foundations and History

Network security, rather security of information as a whole is based upon certain foundation principles: Confidentiality, Integrity, and Availability. Confidentiality holds to keep information from being disclosed to anyone not authorized to access it. Integrity states that information should be kept from modification or otherwise corrupted either maliciously or accidentally. Availability is the ability of a system to keep running efficiently and to keep information accessible. Cryptography conforms to these principles in that ideally only people with knowledge of the secret key can see the decrypted information. Integrity can be verified through the use of checksums to see if data files are modified in any way. Availability can be assured by having backups of cryptographic information and redundant decryption capability if files are deleted or bogged down. Although cryptography and steganography are two different disciplines, there are many lessons to be shared and learned from each. In 1883 Auguste Kerckhoffs first laid out a major principle of cryptographic engineering - we must assume that the method to encipher data is known to the opponent, so security must lie only in the choice of key. (Katzenbessier Pg. 8) Meaning that security by obscurity is simply naïve. To believe that a hacker or spy would not develop knowledge of a communications system is absurd.

Both cryptography and steganography have existed for thousands of years, with the shared goal of concealing information from unauthorized people. Steganography stems from the time of the Ancient Greeks; derived from Greek it literally means covered writing. In his *Histories* Herodotus speaks of Histiaeus and how, around 440 B.C, he shaved the head of his most trusted slave and tattooed it with a message. When the slave's hair grew back, concealing the message he was sent on his way, the message contained instructions to incite a revolt against the Persians. Another example was when Demeratus warned Sparta of an upcoming invasion by Xerxes the Persian King. Demeratus' approach wasn't as painful, as he removed the wax covering a writing tablet, wrote the message on the wood underneath, and reapplied the wax to cover up the message. The tablet appeared blank, managed to get through Persian inspection without suspicion, was delivered, and prevented the invasion of Sparta for the time. (Seeing the Unseen - Johnson P.27) Steganography has been implemented continually throughout history, in times of war and political upheaval especially. During both world wars messages were hidden using invisible ink to mark dots above or below letters in a cover text. An interesting method that developed was microdot printing, in which messages were shrunk by photographic reduction and inserted on top of periods and commas in material such as magazines. (Katzenbeisser, Petitcolas Pg. 4) Some forms of steganography were easily discovered, so we need to examine which types are the strongest.

Steganography Types

When considering the form of steganography to use, one must be conscious to the sensitivity of the data being concealed, and the type of person attempting to attack the cover message. In general there are three types of attackers, passive, active, and malicious; each category of attacker reacts differently to the cover data that they intercept. A passive attacker intercepts the cover material and pores through it to ensure that no secret information is being passed. An active attacker is someone that intercepts cover material, modifies it so as to destroy the secret, (or enables it so that in future exchanges they can access a secret message) then sends the message on to the unsuspecting destination party. The third form of attacker acts maliciously, forging steganographic messages to fool other parties involved in communication. Depending upon whom might be trying to attack your method of concealing data; you would want to make sure that the method you choose is secure. (Katzenbeisser, Petitcolas Pg. 18, 35)

There are three general forms of steganography: pure, private key, and public key (like cryptography). Pure steganography is a system in which no secret information, like a key, needs to be exchanged before cover messages are sent between parties. In this form of steganography, only both parties must have access to both the embedding and the extraction algorithm in order to send secret messages. However, according to Kerckhoffs' principle mentioned earlier, we know that it is assumed that the enemy knows the method of concealing (enciphering) data, so the security of the data depends on the strength of the key. Since the whole basis behind pure steganography is the secrecy of the algorithms, or security by obscurity, we know that this method isn't very secure. (Katzenbeisser, Petitcolas Pg. 20)

Private key steganography is similar to symmetric key cryptography in that a person conceals the secret message using a key. They send the message and the other party would extract the secret message using the inverse of the original process. Ideally, only people with access to the secret key would have access to the secret messages. This method of course assumes that we are able to exchange the secret key securely before communication begins. If there was no way to securely convey the key, it could be derived from the cover object, but then this method defeats the purpose of concealing the data, as an attacker could also derive the key from the cover data. (Katzenbeisser, Petitcolas Pg. 22) Both pure and private key steganography are vulnerable to a man-in-the middle attack, where someone would have access to the secret message, and if they wanted to could alter the messages between both parties.

Public key steganography is again similar to public key cryptography, where embedding and extracting of secret messages are done using key pairs. One key is public knowledge, this is used to embed the secret information, and the private key is used to extract the private message from the cover material. One of the best ways to secure this system is to combine public key encryption with public key steganography. This way one could encrypt a secret message and the resulting ciphertext, which ordinarily looks like random characters, could then be steganographically hidden. This way if an attacker suspects that data is hidden and figures out a way to extract the data, they wouldn't be able to figure out if there was even a message hidden at all, since the extracted message would appear to be random characters. (Katzenbeisser, Petitcolas Pg. 23) In order to get around the problem of malicious and active attackers with steganographic key exchange, one could use digital certificates signed by private key hashes to verify whom the messages were actually sent from. In these steganographic systems information is kept confidential through the secret key in private and public systems, while the secret embed and extract algorithms keep information secure in pure steganography.

Security of Steganographic systems

The purpose of employing a steganographic system is to conceal information. If an attacker can even prove that there is a secret message within a cover material then the system has been defeated. The easiest way in which to do this is if the attacker has access to both the original cover material and the steganographic cover, and by comparing the two. A way to foil this is through the continued use or creation of different cover material. The question of the type of attacker intercepting the messages arises. What happens to the embedded secret message if the attacker alters the cover material in some way? If a secret can be changed or altered, so that after extraction the message is incomprehensible then the steganographic system is not robust. A steganographic system is robust if the secret message can't be changed without drastic changes to the cover material. A robust system creates a supraliminal channel, which makes it impossible to modify the secret message without significant changes to the cover object (Katzenbeisser, Petitcolas Pg.33). If a part of the cover is changed, we could look to another region of the cover object to extract our message. It is the robust steganographic systems that provide us with the integrity we need to ensure that secret messages aren't modified accidentally or maliciously. Robust systems also serve to keep the secret information available to those who need to access it, with information spread throughout the entire cover message, even after significant alteration, users should be able detect the hidden message. The only problem with a system being more robust is that it is not as secure, because a secret message is embedded into significantly more portions of the cover data. We can say a steganographic system is secure even against malicious attackers if:

- Messages are hidden using a public algorithm and secret key, with the key providing non-repudiation of the sender.
- If only the person with the key can detect, extract, and prove the existence of a hidden message.
- If the enemy knows the contents of a secret message, they should not be able to figure out the contents of another message with that information solely.
- It is computationally infeasible to detect hidden messages. (Katzenbeisser, Petitcolas p.34, 35)

In the real world steganographic systems can serve many purposes for communicating confidential information, unfortunately this scheme for communicating data is available to all.

Real World Usage

"Hidden in the X-rated pictures on several pornographic Web sites and the posted comments on sports chat rooms may lie the encrypted blueprints of the next terrorist attack against the United States or its allies. It sounds farfetched, but U.S. officials and experts say it's the latest method of communication being used by Osama Bin Laden and his associates to outfox law enforcement." (Kelley, Terror Groups)

Unfortunately for the United States and our agencies, we most likely did not devote enough resources to discovering such files, as New York experienced the devastation perpetrated by Al-Qaeda on September 11. It has been said for about the past year that Bin Laden and his followers have mastered the use of steganographic tools; in fact rumor has it that a module on Steganography is even taught at the terrorist training camps (McCullagh – Bin Laden). The terrorists are said to use these tools to convey instructions for jihad and martyrdom on the Internet to their followers, many of these messages have been discovered, hidden in mp3 music files and within images on pornographic sites. In order to discover these files communication firms and individuals have created several steganalysis programs that work to break down files into component parts to discover their messages or to brute-force attacks their steganographic passwords. (Manjoo)

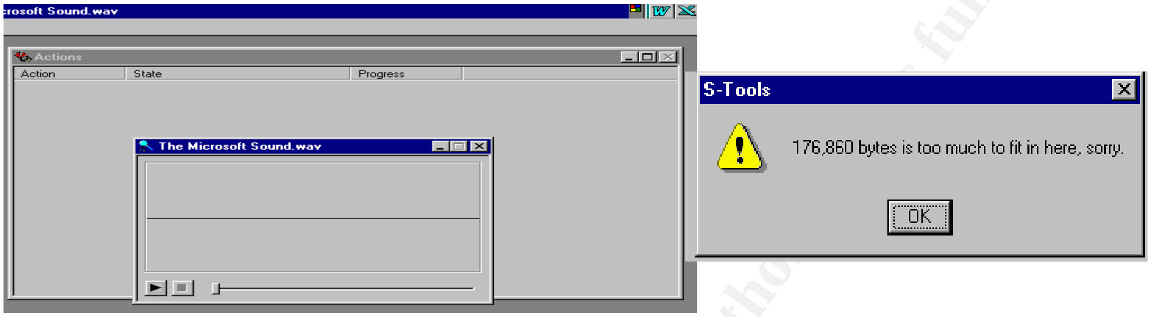
Some of the encryption and steganographic schemes employed have been quite difficult to crack as in the case of Ramzi Yousef, the convicted mastermind of the World Trade Center bombing in 1993, used encrypted files to hide details of a plot to destroy 11 U.S. airliners. Philippines officials found the computer in Yousef's Manila apartment in 1995. U.S. officials broke the encryption and foiled the plot. Two of the files, FBI officials say, took more than a year to decrypt (Kelley, Terrorist Groups).

Steganalysis - Defense

Detecting, extracting, and rendering useless a hidden message within a cover material is known as steganalysis. The first goal of steganalysis is detecting a hidden message. When information is embedded into a cover file, such as an image (.JPEG), some color distortion, or degradation usually occurs. Normally if you choose a cover file significantly larger than the embedded file, the degradation isn't noticeable to the human eye (Hidden Info, Johnson and Jajodia Pg.2). If this is the case one would have to use statistical analysis to determine if the file had been altered, in comparison to the original cover file, also known as a known-cover attack. The amount of degradation that is apparent depends upon the type of cover file and the steganographic tool used to embed the file. Once a hidden message is found you can try and embed another bit of information into the cover and attempt to disable the original secret. If you figure out how to extract the information, you could act maliciously, alter the embedded data for your own benefit, and send the message on its merry way. Besides image files, steganography can also be employed in .WAV files, in the next section we will look at how we can use a steganographic tool to hide data.

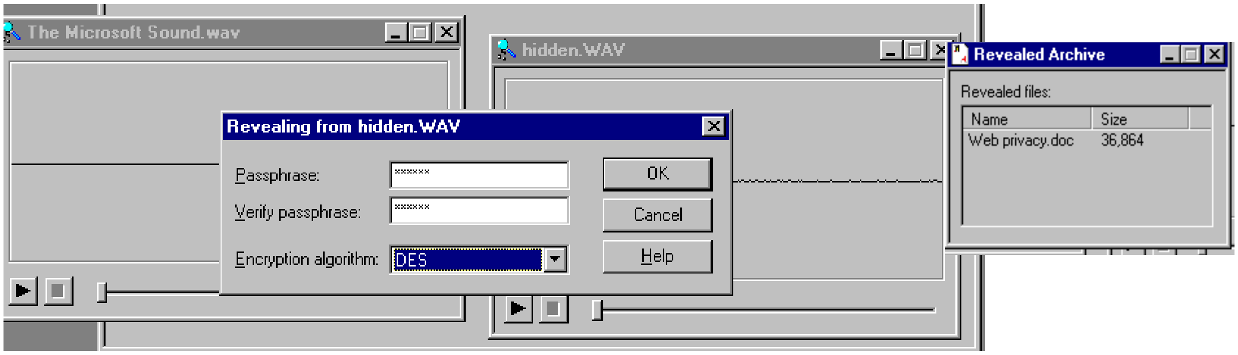
Steganography Tools

There are many steganographic tools available on the web that perform embedding in a variety of mediums, from MP3s to picture formats like JPG, BMP, and GIF. Some tools use GUIs such as Steganos Security Suite and JP Hide and Seek, and some use the command line like Hide4PGP (StegoArchive). The tool that I evaluated was S-Tools version 4, I was first exposed to S-Tools at the SANS conference in San Francisco. S-Tools has a graphical user interface with a drag and drop feature that allows you to manipulate and hide data very easily. Simply drag the BMP, GIF, or WAV file into the application window that you wish to use as a cover file. Here I chose to use the Microsoft Sound.WAV, see below (Brown, S-Tools). Next drag and drop the file, which you wish to embed within. S-Tools is user friendly it even tells you what size file a must be smaller than in order to be embedded, (this can be found in the lower right hand corner). Just to test S-Tools, I chose a file larger than the recommended size, a .PDF, to see what would happen; a pop-up box appears, above, telling the user that the file is too big, and to use a smaller file. Once you choose a file, a .DOC this time, that is under the proper size, you have the option to encrypt the file using a variety of symmetric key schemes: IDEA, DES, Triple DES, or MDC. After encryption is complete in the application window appears the cover file containing the secret document, here hidden.WAV. Extracting the file is just as easy, right-click on the cover file and click reveal, up pops a box prompting you for your secret pass phrase. If you choose the correct pass phrase and algorithm, another box pops up, below (Brown, S-Tools), with the name of the file embedded, right click on the file name to see its properties or to save it. Another feature of S-Tools, which I think to be necessary, is the option of compressing the embedded file. If you compress the embedded file (after tinkering with the level of compression), the cover file and original file will be close to the same size in bytes, so not to raise any suspicion. Second, since there seems to be a reason you are hiding a file, compression makes it more difficult for steganalysis to occur, because compression reduces the amount of signatures found spread within the cover material.



176,860 bytes is too much to fit in here, sorry.

So how is the file hidden within the cover file? WAV files in Windows are either 8 or 16



Passphrase: ******

Verify passphrase: ******

Encryption algorithm: DES

Name	Size
Web privacy.doc	36,864

bit. As we learned from the Security Essentials course, 8-bit values can range between 0 and 255. S-Tools embeds the file within the least significant bits of the cover material, or the right most bits. If we altered the left most bits, according to binary math, the value of the 8-bit number would significantly change. For example we have 10011010 as our 8-bit number, its value is 154. If we changed the right most bit, the bit value would equal 155, if we changed the left most bit, the bit-value would equal 26. A change this radical, if spread through enough bits, might allow the human perception system to pickup the change in sound and raise suspicion. However, since we only change the least significant bits, humans are not able to hear a difference between the original and cover/embedded file. To help stymie attackers, S-Tools uses a pseudo random generator to choose which bits in the cover file to hide the secret message. This is called a random interval method, where the secret message is spread over the entire cover media in a random fashion (Brown, S-Tools). We have seen what steganography can contribute to illicit or secret operations, but what about a practical application for today's economy? One such application is digital watermarking.

Digital Watermarking

Along with hiding messages within cover files, steganography has lead to developments in digital watermarking. Digital watermarking is similar to its paper counterpart in that it serves to frustrate digital counterfeiters. Digital watermarking and steganography are closely related in that both place data within files, however they differ in that steganography aims to keep the data hidden, where a digital watermark becomes part of and aims to protect the cover material from piracy (Seeing the Unseen - Johnson P.29). Pirates know that digital watermarks are present in cover material, they aim to remove or disable the watermark so that the cover material can be copied and freely distributed. Due to the fact that pirates aim to remove or disable digital watermarks, watermarks must be made very robust and strong against attack. Even if watermarks are disabled or distorted there should be a way to prove that the watermarks were there.

Watermarking and steganography share the same ideal setup principles, where both systems contain an embedding and extraction system. Watermarks can be any information that identifies that the cover material is the property of some person or organization. The mechanisms that protect watermarks are their keys and publicly scrutinized algorithms, which is in accordance with Kerckhoffs' principle. Robust watermarking systems rely on three general principles: Imperceptibility, Redundancy, and Keys. (Katzenbeisser, Petitcolas Pg. 101,102)

Watermarks can be visible to the human eye, like on currency or checks when held up to the light or they can be invisible. It is said that invisible (imperceptible) watermarks have an advantage over visible (perceptible) watermarks, because attackers know where in the cover material these watermarks are. The locations of invisible watermarks are not easily known, as they are usually spread throughout a cover media, making an attack against a specific location difficult. However, invisible watermarks can be altered more easily, because if more data is embedded into the cover, like a counterfeit watermark, it isn't readily visible to sight. One solution to this problem is to incorporate a more perceptible watermark, so that if an attack is made on a watermark, a part of the visible media is distorted or corrupted (Johnson – Watermark Recovery Pg.2). Redundancy is the feature most associated with robustness; watermarks are usually distributed over the entire area of the cover material. This allows for the recovery of a watermark after attack, even from the smallest portions of cover media. Being able to recover a watermark from attack, and ensuring a watermark is strong but perceptible is important, however if you do not keep the method of implantation and extraction secure then what is the point? This is why several cryptographic keys or key pairs are used to make sure a watermark is secure; if you are securing intellectual property potentially worth millions, and someone figures out how to read the watermark, they can remove your watermark, and insert their own. How then can you prove that this media is of your creation?

Watermark Attacks

In general there are four types of attacks one can impose on a watermark: robustness attacks, presentation attacks, interpretation attacks, and legal attacks (Craver, Yeo, Yeung Pg. 48). Robustness attacks are the attacks designers have in mind when creating watermarking schemes. The purpose of this type of attack is to degrade the quality of the cover media enough to render the watermark useless, or to remove it. Some methods used in robustness attacks are to compress files, blur images, or even crop the area where the watermark is present. (Katzenbeisser, Petitcolas Pg. 150) Presentation attacks try to manipulate the cover material in such a way that the watermark detection/recovery system fails to even recognize a watermark is present. A common method in presentation attacks is to stretch the digital media, an image, to possess a different shape, without degrading the quality of the media. The slight change of shape causes the media to be out of spec for a watermark detector, i.e. web spider, causing it not be configure correctly and miss the watermark during inspection. (Katzenbeisser, Petitcolas Pg. 160)

Interpretation attacks are engineered by inserting an additional watermark into a cover medium. The attacker's objective is to make it impossible to differentiate between which watermark, the fake inserted one or the original, is the original one. To apply this method an attacker would need to have access to a watermark detector; the attacker can apply little changes to the image, (like inserting his watermark) while testing to see if the detector can find the original watermark. This process can go on by continued strengthening of the attacker's watermark until the original watermark goes undetected (Katzenbeisser, Petitcolas Pg. 155). The other type of attack does not actually manipulate a watermarked file, legal attacks are basically the distribution of copyrighted material in a country that does not enforce intellectual property laws. (Katzenbeisser, Petitcolas Pg. 170) The ultimate purpose of digital watermarks is to prove that your intellectual property is actually yours. Attacks such as the above can occur resulting in your property being stolen and illegally copied, but this scenario would likely occur only with knowledgeable attackers. The presence of visible and imperceptible watermarks would deter casual users from piracy.

Real World Usage

There has been no larger intellectual property controversy in the past couple of years than in the music industry. If the music industry was more embracing to digital media distribution, a watermarking scheme could have been in place to secure intellectual property. Unfortunately all the protests and lawsuits tying up the court system has resulted in a loss of perhaps hundreds of millions of dollars to digital piracy. If an appropriately strong digital watermarking scheme had been established the RIAA (Recording Industry Association of America) and recording artists would have had nothing to complain about. They would be able to track all users who legally and illegally possessed their works, taking whomever to court and collecting their royalties deserved.

Another interesting development related to watermarking is the development of the Concealogram algorithm. Concealogram is not a digital watermark, (it is more of a hard-coded steganography tool) however it is part of the printed version of the digital media, in which a two-dimensional barcode is encrypted inside a halftone image; this can be read by scanning the image with a regular optical scanner. This barcode is similar to ones you would find on products in supermarkets, but is able to hold an incredible amount of data because the information is stored in a binary system. Concealogram is created very robustly, in which it exists within the whole image, avoiding recovery problems caused by damage to part of the image. This way a scanner can still be able to pickup the barcode information such as a name, social security numbers, and fingerprint (Hershman – Steganography Next Generation). The spread of information and digital media across the Internet is a potential boon for the global economy, figuring out a way to secure intellectual property is a necessity to ensure that the creators receive the credit that is due to them.

Watermark attacking tool

There are software tools that can be used to attack watermarking systems. The tool that we will examine is named StirMark. StirMark 3.1, in its current form is a command line program that runs a variety of tests against digital watermarks to try and disable them. StirMark applies distortions to images by moving the corners of an image a random distance in any direction; these procedures crop, rotate, resize, or stretch the image. The goal of StirMark is to prevent a watermark from being detected - known as a presentation attack in an image while at the same time maintaining image quality.

To use StirMark you go to the command prompt, then to the directory where StirMark and the picture you want to attack is located. The command "StirMark -?" brings up a manual describing all of the different options of attacks that can be executed by StirMark, such as

-q, where you can set the level of quality of the output jpeg file.

-d(float) where you specify the maximum byte value that a pixel's color can stray from the original RGB color values.

-o(float) or –I(float) is the distance in which the corners can be moved outside or inside the original corner image positions

A command line entry that encompasses all the tests would be "StirMark –T(output file) (file attacked)" – this command executes all the cropping, rotating, resizing, and stretching modifications to the image (for specifics see the readme file) and saves the output in either a JPG, or .PPM file. To view the results use a photo-editing program, to compare between the original watermarked image and the attacked image, you might not be able to find a difference. The overwhelming success rate of StirMark and other such tools these tools proves that the current watermarking systems that are available are not robust and strong enough to attack (Petitcolas, Anderson, and Kuhn), (Petitcolas – Watermarking Schemes). All of this will change with time, the modern economy depends on electronic communications to spread products and ideas, and therefore investments will be made to improve the protection of their intellectual property.

Conclusion

In today's global economy it is important that governments, businesses, and people can exchange confidential information securely and receive proper credit for their intellectual property. Steganography and digital watermarking are interesting and developing fields that offer several methods and tools to conceal and to protect information. Both steganography and cryptography aim to keep information secure. Further development in each field and the use of their synergies can better secure digital information for tomorrow in accordance with the principles of confidentiality, integrity, and availability.

References:

Katzenbeisser, Stefan, Petitcolas, Fabien A.P. Information Hiding techniques for steganography and digital watermarking. Boston, London: Artech House, November 2000. Pages 8,18,20,22,23,33-35,101,102, 150, 155, 160, 170.

Craver, S., B.-L. Yeo., and M. Yeung. “Technical Trials and Legal Tribulations.” Communications of the ACM, Vol. 41, no7, July 1998: 44-54

Kutter, Martin, Petitcolas, Fabien A. P. “A fair benchmark for image watermarking systems,” To in E. Delp et al. (Eds), Proceedings of Electronic Imaging '99, Security and Watermarking of Multimedia Contents, vol. 3657, San Jose, CA, USA, 25-27 January 1999.

Brown, Andrew. “S-Tools v4” Stegoarchive.com, 1997. http://www.stegoarchive.com

Gibson, Owen. “Hidden web codes could be linked to Bin Laden.” Guardian Unlimited, October 10, 2001. http://www.guardian.co.uk/Archive/Article/0,4273,4274369,00.html (January 15th, 2002)

Hershman, Tania. “Steganography, Next Generation.” Wired Magazine, December 21, 2001. http://www.wired.com/news/conflict/0,2100,49213,00.html

Johnson, Neil F. “An Introduction to Watermark Recovery from Images.” *IEEE Computer*, February 1999: 26-34. http://ise.gmu.edu/~csis (January 10th 2002)

Johnson, Neil F., Jajodia, Sushil. “Exploring Steganography: Seeing the unseen.” *IEEE Computer*, February 1998: 26-34. http://ise.gmu.edu/pub/r2026.pdf (January 10th 2002)

Johnson, Neil F., Jajodia, Sushil. “Steganalysis: The Investigation of Hidden Information,” *IEEE Computer*, September 1998: 26-34. http://www.jjtc.com/pub/it98jjgmu.ps (January 10th 2002)

Kelley, Jack. “Terror groups hide behind Web encryption.” USA Today, Updated June 19, 2001. http://www.usatoday.com/life/cyber/tech/2001-02-05-binladen-side.htm

Kelley, Jack. “Terrorist instructions hidden online.” USA Today, Updated June 19, 2001.http://www.usatoday.com/life/cyber/tech/2001-02-05-binladen.htm

Manjoo, Farhad. “Hidden Messages: Any There?“ Wired Magazine. November 8, 2001. http://www.wired.com/news/technology/0,1282,48235,00.html

McCullagh, Declan. “Bin Laden: Steganography Master?” Wired Magazine, February 7, 2001. http://www.wired.com/news/politics/0,1283,41658,00.html

McCullagh, Declan. “Secret messages come in .WAVs” Wired Magazine, February 20, 2001. http://www.wired.com/news/politics/0,1283,41861,00.html

Petitcolas, Fabien A.P. "The Information Hiding Homepage, Digital Watermarking & Steganography" Monday, 28 January 2002 http://www.cl.cam.ac.uk/~fapp2/steganography/

Fabien A. P. Petitcolas. "Watermarking schemes evaluation". *I.E.E.E. Signal Processing*, vol. 17, no. 5, pp. 58–64, September 2000. http://www.cl.cam.ac.uk/~fapp2/publications/ieeespm00-evaluation.doc

Fabien A. P. Petitcolas, Ross J. Anderson, Markus G. Kuhn. "Attacks on copyright marking systems," in David Aucsmith (Ed), Information Hiding, Second International Workshop, IH'98, Portland, Oregon, U.S.A., April 15-17, 1998, Proceedings, LNCS 1525, Springer-Verlag, ISBN 3-540-65386-4, pp. 219-239. http://www.cl.cam.ac.uk/~fapp2/publications/ih98-attacks.pdf