BETTERMENT OF BUSINESS COMMUNICATION WITH STEGANOGRAPHY AND CRYPTOGRAPHY

Divyakant T. Meva, Jaypalsinh A. Gohil, Amit K. Patel

ABSTRACT

One of the biggest problems that most companies have regarding security technology is that they don't develop a strategy for implementation. Many companies seem to have the attitude that implementing a single technology, such as firewalls, will make them secure. In reality, just putting a firewall in place is not enough: That firewall has to be designed and configured properly for it to do its job successfully, and it must be used in concert with other security measures. And all these security measures must be properly integrated with the current network.

Secure communication is no different from information security: You have to pick the right tools, implement them properly, and train users to use them in their daily work.

This paper will look at the kind of assessment involved in developing a secure communications strategy that might include stego and crypto.

Keywords: Steganography, Cryptography, Communication, public key steganography, private key steganography, Steganalysis

1. INTRODUCTION

Secure versus Secret

You may not have a master of secure communications on your payroll, but you need to develop internal expertise to make important communications secure in your working world.

As you begin developing a secure communications plan it is important to remember the distinction between secure and secret. Making something secure involves ensuring that no one can read or gain access to a piece of information.

When you go to bed at night you may first lock the house and set the alarm. You have made the house secure. In most cases, when we talk about secure communications we are referring to cryptography, which keeps people from being able to read information.

Secret or covert communication involves hiding the fact that anything sensitive exists at all. Generally, when people talk about secret or covert communication, they are referring to stego. it is also secure to a degree. The degree of security rests in how well the secret item has been hidden.

The difference between secure and secret communications reinforces the concept of defence in depth. Only by putting many different technologies together, creating a layered approach, will you be secure. In the area of secure communication, this approach would suggest the wisdom of using both crypto and stego in concert.

2. SETTING COMMUNICATION GOALS

There is a saying that if you define your target after you shoot an arrow, you will hit your target every time. Many companies approach secure communication this way. Let's put something together, and whatever protection we build will define our security goal. In the long run, this is not an effective approach.

You have to define your communication goals before you can plan your secure communication strategy.

What follows is a list of the goals an organization might want to achieve when using secret communication. For each goal I have noted whether it is achieved through using crypto, stego, or both.

Communication goes to the intended party (crypto).

If you send a message it is important that it goes to the person for whom it is intended. At a basic level this is done by specifying the proper destination, such as an IP address or email address. At a more sophisticated level, guaranteeing the identity of the recipient can be ensured by crypto and the use of private key encryption.

Communication is not modified in transit (crypto).

Validating the integrity of a message is critical to ensuring that a person-in-the-middle attack, where data is modified in transit, was not performed. Performing a digital signature with crypto is critical to making sure that data has not been modified.

Communication does not go through a hostile person (crypto/stego).

In a situation where you cannot control where packets of data actually go as they travel to the intended recipient, if the data is hidden and protected the impact of a hostile entity accessing the data is minimized.

Communication is not read by unauthorized people (crypto).

If someone can intercept data, the concern is whether they can read or access information that they should not have access to. By encrypting the information nobody will be able to read or make changes to it.

The fact of the communication is hidden from unauthorized people (stego/digital dead drop).

In some situations, even if someone cannot read the information that is being communicated, the mere fact that information is being communicated or that encrypted information is being transmitted can raise suspicion. By using stego with a digital dead drop the relationship between the two parties is hidden, in addition to the existence of the communication.

The true intent of the communication is not discovered (stego).

In some cases, the fact that there is communication between two parties may not be a concern, but the intent of that communication may be. This situation is ideal for stego because the two parties can communicate, but the true intention (message) of the communication is hidden.

The organization can prove in a court of law that communication was sent from a given person (crypto).

Non-repudiation (being able to prove in a court of law that a specific person sent a communication) is at the heart of e-commerce. Why would anyone use digital contracts or electronic signatures if they were not binding? Using crypto with digital signatures provides the means for non-repudiation.

3. ROLES OF CRYPTO AND STEGO IN BUSINESS

Though these technologies are more in use by the criminal element of our world, the growing interest in them in the corporate sector suggests that they will become an everyday part of information security as time goes by. How will they fit into your organization? Read on.

3.1 Why You Need Both Stego and Crypto

Whenever you consider two or more technologies to be used in your communications security strategy, you should determine whether they are complementary or competing. If they are competing, that means they both do the same thing and the technologies are redundant. If they are complementary, they provide different services; by putting them together, you obtain a more robust result.

Stego and crypto are definitely complementary technologies. They provide two different services to the data they are protecting. Stego hides the existence of the data, while crypto makes the contents of the data difficult to read.

3.2 Crypto and Stego in Business Today

I know of several organizations that have gone to great lengths to roll out encryption across their enterprise, only to use the same key for all their data.

To add insult to injury, they then make that key available to everyone in the organization. I have seen organizations put their key on a public server with no protection whatsoever. Organizations may use crypto during the transmission of data but then store the data in unencrypted form on a Web server that is accessible from the Internet. In short, not many businesses are using crypto, and those that are often use it incorrectly.

If the crypto side of the coin does not look so good, the stego side is even worse. Since September 11, 2001, the corporate security landscape is changing, but a large percentage of people still do not even know what steganography is.

Let us see some examples of steganography.

1. **CameraShy:** It is a program that hides data in the least significant bits of GIF images. What makes this program unique is that its graphical user interface (GUI) makes it a perfect tool for network stego. CameraShy actually looks and acts like a browser that enables you to scan any intranet or the Internet for files containing hidden data.
2. **Invisible Secrets:** It is a very powerful suite of tools that allow you to both embed hidden data in encrypted files and transmit it. Invisible Secrets can embed information win files in the following formats: JPEG, PNG, .bmp, HTML, and .wav.

3.3 How Crypto and Stego Make You More Secure

A lot of the issues relating to secure communication on the Internet (email, Web page contents, and file transfer, to name a few) would go away if all data was stored in an encrypted format and hidden in files on hard drives on workstations or servers. Attackers would face the challenge of first locating the information, then breaking the crypto to read it. Certain types of attacks, such as denial of service, would still occur, but corporate espionage methods that modify or access proprietary information could be foiled to a great extent.

Remember that there is no silver bullet when it comes to security, but crypto and stego used together provide a high grade of protection against the most serious attacks that occur when data is placed on a network or transmitted via the Internet.

3.4 Developing a Strategy

The first step in developing a secure communications strategy is to analyze the communication patterns in your organization, detailing what type of information is communicated and by whom.

Based on that information you can establish a data security ranking system with categories such as these:

- Highly confidential data, such as product plans.
- Private data, such as employee salaries.
- Internal data, such as product code names and release dates.
- Public data, such as product brochures.

With these categories of data in place, instruct each employee on how to use the ranking and to respect how each type of data is to be used, stored, shared, and communicated.

The next task is to go through a decision process about which form of secure communications tools might be used with each type of data.

Don't forget to train your employees and management in the use of the data-ranking system and secure communication tools. Let them know the seriousness of the need for security by instituting certain actions if the guidelines are not followed.

Set up a reporting process for breaches of information security so these can be reported and acted on to stop an attack in progress, minimize damage, or prevent future security breaches.

4. COMMON PROBLEMS WITH SECURE TECHNOLOGIES

In the spirit of David Letterman, Let us see the common questions which should be asked before deploying secure technologies.

This list can also be thought of as a checklist for a secure communication strategy.

1. **Did you determine which technology is a higher priority and deploy that technology first?**

   It is usually considered good practice to deploy one technology, make sure it works, and then deploy the next technology.

   If you deploy two technologies at the same time and there is a problem, troubleshooting the problem becomes harder. On the other hand, even though you are only deploying one technology at a time, if you are planning on eventually implementing multiple technologies, it is highly recommended that you test them together for compatibility before deploying them. Also, if your company is planning to use these technologies with business partners, it is usually a good idea to make sure that those partners use the same technologies or compatible technologies.

2. **Did you integrate the technology at the lowest level possible?**

   For example, it is usually better to implement encryption between layers 3 and 4 in a VPN rather than at the application layer. If you implement the technology at a higher level you have to deploy a version for every single application you use. If you integrate it at a lower level, all corresponding applications will also be able to take advantage of the same encryption.

3. **Did you use the techniques in the proper order?**

   When using crypto and stego together, it is recommended that you encrypt the information first and then hide it in a file. Because crypto has a signature and is detectable, performing crypto last will leave a signature someone can use to track down suspicious files. By using crypto and then stego, and then hiding the encryption in a file or network stream, you make it much harder for someone to detect it.

4. **Are your keys and passwords for encryption and stego properly protected?**

   Remember one of the golden rules of secure communication: The strength of a communication technology is based on the secrecy of the key, not the secrecy of the algorithm. If someone can find your keys or the password/pass phrase that you used to protect your information, you have just defeated the whole purpose of using such a technique. It is critical that keys and passwords be very hard to guess, and even harder to find.

5. **Are your stego tools properly protected from detection or tampering?**

The goal of stego is to avoid detection, so if you are not supposed to be using stego but you leave stego tools on your system where someone can find them, he or she can figure out what you're up to.

6. **Are your users properly trained, and do they understand the technologies?**

This is a challenging process to institute at some companies. On one hand, you should not expect your users to have a thorough understanding of stego or crypto in order to use them; on the other hand, they should have an understanding of the value of decryption and encryption and appreciate that a private key must be kept secure.

7. **Are the technologies as transparent to the user as possible?**

Even though users should have an appreciation of the need for secure communication, you should make the technologies as transparent as possible to them. The less they have to learn and remember the less chance of error.

8. **Is your IT and administrative staff trained on the potential implications of stego and crypto?**

When you encrypt information or hide data in files or network traffic, it should be made known to the people that manage the servers and networks. Surprises are not a good thing when it comes to healthy networks. Make sure your technical staff is well trained and understands any potential implications these technologies might have—for example, as they use traffic analysis or intrusion detection systems. Because the data portion of encrypted files might look unusual, these systems may flag them. Also, because stego uses image files in many cases, if you want to generate a lot of stego, you may be saving a great many large graphics files that eat up server space.

9. **Did you test the tools before deploying them?**

The golden rule of technology is that you should always test before you deploy. After you test it, test it again. If you roll out a technology to 2000 computers and then find a problem, it can be very hard to rectify that problem. I recommend doing incremental rollouts. Once a technology is tested, roll it out to 10 people and make sure there are no problems, then roll out to 30 more people, then to 100, and so on. In this way, you minimize the chances of introducing errors to a large number of systems at once.

10. **Do you understand the inherent weaknesses in the tools and take measures to protect against them?**

No tool is perfect, and you need to understand the shortcomings of any techniques that you deploy.

Deploying a technique and thinking it is going to make you completely secure is naïve at best. Understanding the limitations and taking action to minimize the impact those limitations can have on your organization are the smart things to do.

5. PRIVATE-KEY STEGANOGRAPHY

Let us assume that Alice and Bob are allowed to share a secret key prior to imprisonment, or even to trade public keys. This gives them the opportunity not only to communicate covertly, but to defeat an active warden. In the former case, steganography consists merely of encrypting a message in such a way that the ciphertext appears statistically random, and embedding the bits of the text in a known subliminal channel. The embedded information, of course, must be made to have the same distribution as the channel noise in order to foil statistical tests.

In the presence of an active warden, it is not enough to embed a message in a known place. If Alice can subtly alter the bits in an image, it follows that Wendy could scramble those same bits with as little impact; erasing whatever was being sent via the subliminal channel. In this case it is possible to use what is referred to in as a "selection channel."

Essentially, the secret information shared between Alice and Bob is used to determine *where* the message is hidden. A cryptographically secure pseudo-random generator, seeded by a secret key, can be used to pick a subset of pixels in an image, for instance, to be used to conceal the data. If Wendy attempts to make subtle changes to the image, she may only be able to scramble a small percentage of the actual channel bits, since she doesn't know exactly where they are. This scrambling can then be fixed using an error-correcting code.

The sharing of keys before imprisonment, however, is a requirement that we would ultimately like to see removed. It allows a great deal of freedom on the part of Alice and Bob — indeed, if they share public keys before imprisonment, they can even defeat a malicious warden by signing their secret messages to prevent impersonation—but it is not reassuring to think that if two people ever need to communicate covertly, they must know so far in advance that they can trade secret keys before a real-world "warden" starts listening in.

6. PUBLIC KEY STEGANOGRAPHY

In private key steganography, the sender and receiver share the stego key. In cases where this assumption is not valid, it is still possible with public key cryptography, to send hidden messages that can be decoded only by the intended recipient. The main steps are

1. Encrypt the stego key with public key of the receiver.
2. Suppose that these encrypted data are n bits long, encrypt the value of n embed it in the first b video frames of a video file, where b is a constant.
3. Embed the n bits in the following n video frames by modifying the parity of one region in each frame.
4. Embed the secret message in the remaining frames using the stego-key.

This way the sender and receiver can communicate secretly if they have the same software. And don't have to share the value of n.

This approach can be extended to the case where the attacker tries to modify randomly selected bits in the stego cover. Clearly, if the attacker is free to stop the message from reaching the receiver, or to modify arbitrarily many bits, the embedded secret data would be lost. However if the attacker is allowed only bit modifications that leave the stego cover visibly unchanged, then only a limited number of bits can be modified in the attack. This destroys the part of embedded data but not all of them, which suggests the use of redundancy. Embedding the data as before but with a sophisticated error-correcting code, can result in a robust algorithm where secret data can be retrieved even after the attacker has randomly modified one bit per frame. The trade off is the loss of embedding capacity.

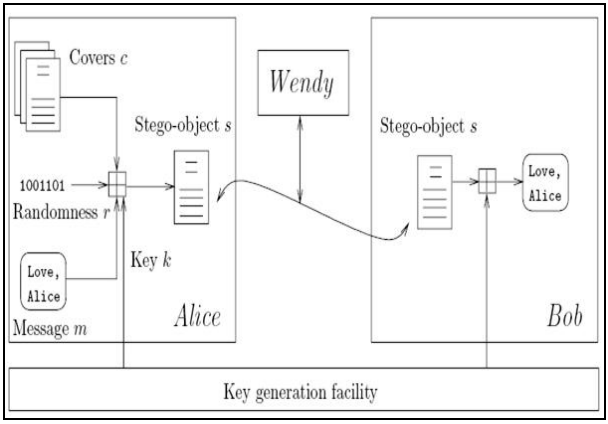

Fig. 1: Public Key Steganography

6.1 Products That Implement Steganography

In this section we'll look at several freeware, shareware, and commercial steganography programs and learn how they go about hiding data in a host file. The tools that you will work with in the following sections are as follows:

- S-Tools
- Hide and Seek
- JSteg
- EZ Stego
- Image Hide
- Digital Picture Envelope
- Camouflage
- Gif Shuffle
- Spam Mimic

Though there are hundreds of stego tools available, this section provides a sampling of insertion-, substitution-, and generation-based stego techniques.

Most of the other tools that are available today are similar to these tools in functionality and usability.

ENTRY NUMBER	R	G	B
0	24	104	155
1	41	100	65
2	24	120	179
3	33	83	49
4	82	132	90
5	65	125	90

Table 1: Sample Color Table

S-Tools

S-Tools is a freeware program with a drag-and-drop interface that runs on most versions of Windows 95 or later. It can hide data in GIF or .bmp image files or in .wav sound files. It can also perform encryption with IDEA, DES, Triple-DES, and MDC. Compressing files is also an option.

S-Tools offers the ability to hide multiple secret messages in one host file. For all of the file formats, it hides data in the three least significant bits of each byte of data.

Using S-Tools with Image Files

For image host files, S-Tools works by distributing the bits of the secret message across the least significant bits (also referred to as LSB) of the colors for the image. The method used for hiding data in images depends on the type of image.

For example, the .bmp format supports both 24- and 8-bit color, whereas the GIF format supports only 8-bit color. Also, 24-bit images encode pixel data using 3 bytes per pixel, 1 byte for red, 1 byte for green, and 1 byte for blue. The secret message is hidden directly in the three LSB of the pixel data. Note that 8-bit images are different from 24-bit images in that they are created with a reduced file size. These images use a color table (or palette) of 256 RGB values. This means the color table has 256 entries. The pixels are represented by a single byte, which specifies which RGB value to use from the color table.

To hide data in 8-bit images, S-Tools modifies the image to use only a 32- color palette instead of 256. The 32 colors are duplicated 8 times (32□8 = 256), to fill the color table with duplicate entries. S-Tools can then use the duplicate entries to store the secret message in the three LSB for each RGB entry. Because each color in the modified image can be represented in eight different ways, information can be hidden in any of the redundant representations. This is the method used most often by S-Tools because most images are stored as 8-bit to save space.

Using S-Tools with Sound Files

For sound files, data is placed directly into the three least significant bits of the file. This works with either 8-bit or 16-bit .wav files. Here's an example that shows how this works with S-Tools: Suppose that a sound sample had the following 8 bytes of information in it somewhere:

132 134 137 141 121 101 74 38

This is how this information would be represented in binary:

10000100 10000110 10001001 10001101 01111001

01100101 01001010 00100110

You want to hide the binary byte 11010101 (213) in this sequence. We simply replace the LSB of each sample byte with the corresponding bit from the byte you are trying to hide. The original sequence will change to:

133 135 136 141 120 101 74 39

In binary, this is:

10000101 10000111 10001000 10001101 01111000

01100101 01001010 00100111

The left-most byte is now 10000101, meaning that only the last bit changed to 1. This occurred because the data that needed to be hidden in that bit was 01, causing the last two bits of the original byte 00 to change. If the first two bits of the original byte had been 10, then both bits would have to change to get to 01.

If the two bits of the original byte were 01, then none of the bits would have to change.

S-Tools Step-by-Step

Using S-Tools is very easy because of its drag-and-drop interface. Because of this ease of use and the powerful hiding techniques it employs, S-Tools is one of the most popular stego tools available.

If you'd like to try S-Tools out, locate the file on the accompanying CD, install it, and then follow these steps:

1. Open the program from the Windows Start menu.
2. Drag the image in which you want to hide data into the S-Tools window from the Windows desktop or an open folder. The image appears in the S-Tools window, as shown in Figure 2.
3. Drag the file that contains the message you want to hide into the S-Tools window and place it over the host image. The dialog box shown in Figure 3 appears.
4. Enter the pass phrase, and click OK.
5. At this point, a new image will appear that looks identical to the original image, except that data has been hidden in it. Save the covert file; you might also want to destroy the original image at this point so that nobody can locate it and compare the two.

Fig. 2: S-Tools with an image loaded.

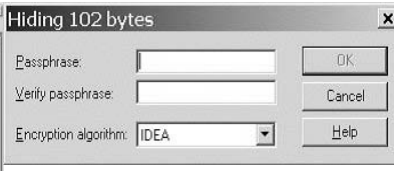

Fig. 3: S-Tools prompting the user for a pass phrase.

6.2 Steganalysis

A goal of steganography is to avoid drawing suspicion to the transmission of a hidden message, so it remains undetected. If suspicion is raised, then this goal is defeated. Steganalysis is the art of discovering and rendering such messages useless.

Attacks and analysis on hidden information may take several forms: detecting, extracting, confusing (counterfeiting or overwriting by an attacker, embedding counter information over the existing hidden information), and disabling hidden information.

The steganalyst is one who applies steganalysis in an attempt to detect the existence of hidden information.

In steganalysis, comparisons are made between the cover-object, the stego-object, and possible portions of the message. The hidden message in steganography may or may not be encrypted. If it is encrypted, then if the message is extracted, cryptanalysis techniques may be applied to further understand the embedded message.

In order to define attack techniques used for steganalysis, corresponding techniques are considered in cryptanalysis.

Attacks available to the cryptanalyst are *ciphertext only, known plaintext, chosen plaintext,* and *chosen ciphertext.[21]*

Parallel attacks are available to the steganalyst:

- **Stego-only attack.** Only the stego-object is available for analysis.
- **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 stego attack.** The steganography tool (algorithm) and stego-object are known.
- **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.
- **Known stego attack.** The steganography algorithm (tool) is known and both the original and stego-objects are available. [21]

7. CONCLUSION

From the above discussion, we can conclude that either cryptography or steganography is not enough for secure and secret business communication. We have to combine them for betterment of communication. You can choose one the possible ways for combining them, either public key or private key steganography. We have taken an introduction to the steg-analysis here. The tools are available in the market for steganalysis and cryptanalysis. The other important issue is where to implement steganography. These point can be discussed in future.

REFERENCES:

1. Cole, Eric, Hiding in Plain Sight:Steganography and the Art of Covert Communication, Wiley Publishing Inc.,2003, pp.217 – 222.
2. Salomon, D. Data Privacy and Security. Springer-Verlag, New York, Inc., 2003.pp.362
3. R.J. Anderson, “Stretching the Limits of Steganography.” in Information Hiding, Springer Lecture Notes in Computer Science v 1174 (May/June 1996), pp 39–48.
4. Craver, Scott, On Public-key Steganography in the Presence of an Active Warden, 1996
5. Fridrich, J., Goljan, M., and Hogea, D. (2002). Steganalysisof jpeg images: Breaking the f5 algorithm. In Proc.of In 5th International Workshop on Information Hiding.

6. Johnson, N. F. and Jajodia, S. (1998). Exploring steganography: Seeing the unseen. Computer, 31(2):26–34.
7. Kerckhoffs, A. (1883). La cryptographie militaire. Journal des Sciences Militaries, 9th series(IX):5–38.
8. Kharrazi, M., Sencar, H. T., and Memon, N. (2004). Image steganography: Concepts and practice. In WSPC Lecture Notes Series.
9. Menezes, A., van Oorschot, P., and Vanstone, S. (1996). Handbook of Applied Cryptography. CRC Press.
10. Provos, N. and Honeyman, P. (2003). Hide and seek: An introduction to steganography. IEEE SECURITY & PRIVACY.
11. Shannon, C. E. (1949). Communication theory of secrecy system. Bell Syst. Tech. J., 28:656–715.
12. Simmons, G. J. (1984). The prisoners' problem and the subliminal channel. In Advances in Cryptology: Proceedings of Crypto 83, pages 51–67. Plenum Press.
13. Westfeld, A. (2001). F5-a steganographic algorithm: High capacity despite better steganalysis. In Proc. 4th Int'l Workshop Information Hiding, pages 289–302.
14. Westfeld, A. and Pfitzmann, A. (1999). Attacks on steganographic systems. In Proc. Information Hiding 3rd Int' Workshop, pages 61–76.
15. William Stallings; Cryptography and Network Security: Principals and Practice, Prentice Hall international, Inc.; 2002.
16. Oded Goldreich; Foundations of Cryptography, China Machine Press, 2003.
17. Jae K. Shim, Anique A. Qureshi and Joel G. Siegel, The International Handbook of Computer Security,Glenlake Publishing Company, Ltd., Glenlake Publishing Company, Ltd., 2000.
18. Ma Shilong, Emad S. Ibrahim, and Hala A. Bayoumy, An introduction to Admire in China (Advanced Multimedia Interactive Real-time Environment), ICAIA' 2003, proceedings of the 11th International Conference on Artificial Intelligence Applications, Cairo, Egypt, February, 2003.
19. Emad S. Ibrahim and Hala A. Bayoumy, "Novel Authentication Approach Using The MSPC", the Proceedings of the 7th IEEE International Conference on Intelligent Engineering Systems, INES2003, Assiut - Luxor, Egypt, March, 2003.
20. http://www.jjtc.com/stegdoc/stegdoc.html
21. Stefan Katzenbeisser, Fabien A. P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking", Artech House, 2000