

Steganography - The Art Of Hiding Information

Martin Bober

January 8, 2009

This article introduces the reader into the field of steganography. In the first part, the idea behind steganography is presented with the help of the Prisoner's Problem. The second part shows how steganography can be done. In the final part, some real-world applications of steganography are presented.

1 What is steganography?

1.1 The Prisoner's Problem

To find out what steganography is imagine Alice and Bob sitting in a prison. They want to make an *escape plan* but all their communication is monitored by Willie the *warden*. Normally we would consider the use of cryptography, but in this case that would be a bad idea. If Willie notices Alice and Bob exchanging secret messages he will reinforce the security measures making their escape impossible. So Alice and Bob have to find a way to exchange secret messages which looks totally innocent to Willie. This is what steganography is about.

When Alice sends a message to Bob, she takes that secret message and embeds it into a unsuspecting cover message. The resulting message is called a stego message. If Willie can tell that there is a secret message hidden in the stego message, he wins. If he fails to do so, Alice and Bob (and steganography) win.

It is assumed that Alice and Bob had a chance to secretly exchange their steganography protocol so that Bob is able to find the secret message within Alice's stego message.

1.2 Variations

Normally Willie is just a passive listener to the stego message trying to detect a secret message. Also there are some variations of the Prisoner's Problem in which Willie can actively influence the stego message (for example distort it). In that case, his goal is to destroy any possibly existing secret messages. Of course he has to respect Alice's and Bobs civil right to exchange messages so that the noise Willie is allowed to add to their communication channel has an upper bound.

2 Hands on steganography

In this section some basic methods of steganography are presented. The secret message is assumed to be pure binary data (which can be anything from ASCII text to a jpeg image). Although the methods for injecting data into the cover message do not depend on the type of the secret message, they highly depend on the type of the cover message.

Basically, there are two different types of cover messages: binary data and text.

2.1 Injecting into binary data

Cover messages consisting of uncompressed binary data are very easy targets for steganography because hardly any *source coding* has been done. Therefore, a high percentage of irrelevant information is still left in the data. Irrelevant information can be used to hide the secret message in. Waveform audio files (*.wav) and bitmap image files (*.bmp) are good examples for uncompressed binary data.

Injecting the secret message is done by *tweaking* the *least significant bit* (LSB) of certain bytes. The result is a slight change in a pixel's brightness or a slightly changed volume of an audio sample. Both are very hard to notice for Willie, because (if the implementation is properly done) he cannot tell if the LSB is the result of Alice's manipulation or of totally normal noise in the cover data.

Bob has to be able to recover the secret message from the stego message, i.e. he has to know which bits have been tweaked by Alice. Therefore Alice is bound to her understatement with Bob about which bits she has to tweak. Unfortunately, not all bytes are appropriate for changing their LSB. For example, a single brighter pixel in a large area of pixels that all have the same brightness would make Willie suspicious. Therefore it would be nice for Alice to be able to choose the most appropriate byte to tune out of a larger set of bytes. This can be achieved

with *parity* checks.

Using parity checks, Bob does not treat the LSB of a certain byte as the secret message. He calculates the parity of a certain set of bytes. In this case, Alice can flip a single Bit out of the hole set of bytes to inject the secret message, making it easier to find a bit where manipulation is the least *obvious*.

2.2 Injecting into text

Text files have hardly any irrelevance left. All their bits have the same (high) significance. Tweaking the LSB of a ASCII-coded letter would result (in the best case) in an unlikely spelling mistake, making it easy for Willie to decide that there is a secret message in the stego message. Therefore the methods explained above cannot be applied.

What can be done is to hide the secret message in the irrelevance of human language instead of in the irrelevance of noise. There are several sentences that basically say the same. The secret message can be encoded in the choice of the sentence out of the set of synonymic sentences.

To do so, a hash function is needed that assigns either a one or a zero to every possible sentence depending on the choice of words and their order in the sentence. Alice and Bob have to share this hash function so that Bob is able to decode the secret message.

Encoding can be done using a special *word processor*. After the input of every sentence, the word processor passes the sentence to the hash function which decides if the sentence represents a one or a zero. If the result does not match with the current bit of the secret message, the word processor gives Alice a signal so that she can rephrase the sentence until the hash function returns the desired result.

3 Applications for steganography

Applications for steganography can be divided into two groups. One group using a passive warden in the Prisoner's Problem and another group using an active warden.

A Selected Translations

warden - Warter; *escape plan* - Plan fur einen Ausbruch; *source coding* - Quellcodierung (Informationstheorie); *to tweak sth.* - etw. anpassen; *least significant bit* - niederwertigste Bit einer Binarzahl; *parity* - Paritat (Prufen, ob Anzahl der Einsen in einer Binarzahl ungerade ist); *obvious* - auffallig; *word processor* - Textverarbeitungs-Software; *hostile surveillance* - feindliche Uberwachung; *to deploy sth.* - etw. herausbringen; *citizens* - Burger; *conspiratorial* - konspirativ; *to obscure sth.* - etw. verschleiern; *watermark* - Wasserzeichen; *fingerprint* - Fingerabdruck; *to determine sth.* - etwas ermitteln; *to prosecute sth.* etw. (rechtlich) verfolgen

On the one hand, scenarios with a passive warden are the classic applications for steganography. Whenever one has to communicate secretly under *hostile surveillance*, steganography is the weapon of choice. In history, steganography has often been used *to deploy* information to agents in a foreign country through public available information sources such as newspaper articles or photographs.

Today, the desire of many governments to monitor the communication of their *citizens* is growing and the use of cryptography is often considered as a *conspiratorial* act. Therefore a more subtle way of *obscuring* the true content of ones communications like steganography may be desired.

On the other hand, scenarios with an active warden are relatively new. Applications using these scenarios like *watermarking* and *fingerprinting* are mainly applied by intellectual property rights owners. Watermarking means embedding a secret message (such as a program's identity) in the output of a program.

For example, a student edition of a program for editing digital images could watermark each image with its ID. If the manufacturer of the program finds a picture with such an ID in a commercial context, he could sue the author for violating the licence agreement (non-commercial use only.) In that case, the program would be Alice the manufacturer would be Bob and the user would be the active warden who tries to destroy the watermark in the image.

Fingerprinting is similar to watermarking except that each file has its own unique ID. For example, fingerprints can be attached to music files bought in a on-line music store so that the origin of illegal copies of that file can be *determined* and *prosecuted*.

References

- [1] Christian Cachin. An information-theoretic model for steganography. In *Proceedings of 2nd Workshop on Information Hiding*.
- [2] Fabien A.P. Petitcolas Ross J. Anderson. On the limits of steganography, 1997.