Analysis of Stegosystems

Sam Britt, Tushar Humbe, Ben Perry, Sanchita Vijayvargiya

December 7, 2012

Introduction

Steganography effectively hides a message within plain sight, wherein a message being communicated is hidden inside another ostensibly innocent message. The message is imperceptible to the adversary that acts as a regulator within the communication channel. Only the sender and the receiver situated at either ends of the communication channel are capable of deciphering existence of the hidden message.

The concept of steganography has been in use since the times of ancient Greece. Communicating messages concealed underneath the wax of writing tablets and the technique of dotting successive letters are some of the ways in which the Greeks steganography. Pirate lore includes tales of individuals tattooing secret information, such as maps, on their head which would be covered by their hair. Throughout World War 2 the grill method was used by the spies. This method involved wooden templates which would be placed over seemingly harmless text, revealing the secret message. During the same period, the Germans devised microdot technology in which a picture could be printed clearly even after shrinking it to the size of a dot. At the time of the British Rule in India, freedom fighters, in order to communicate with each other used lemon extract to write hidden messages on a paper containing general information. The hidden message became visible only when the paper was heated by exposure to a candle or the sun.

Gustavus Simmons, in 1984, initiated the scientific research on steganography in "The Prisoner's Problem and the Subliminal Channel" TODO: cite [3] where he illustrated the concept of steganography with the help of Prisoner's Problem. Two prisoners, Alice and Bob, attempt to devise a plan to escape even though they are locked up in areas and are prohibited from engaging in private communication with each other. Both of the prisoners can correspond merely through a single communication channel which is

scrupulously monitored by the warden of the prison. The warden, Wendy, is the adversary in this example and is trying to intercept Alice and Bob's private communications. If Alice and Bob try to exchange messages that are not completely open to Wendy, or ones that seem suspicious to her, they will be put into a high security prison forestalling their plan to escape. Messages sent without any hidden message are known as covertexts. Messages sent on the channel containing a hiddentext are known as stegotexts. At this point, Alice and Bob make use of steganography, by sharing a secret method or key to hide hiddentexts in some stegotext. Only the two accomplices are able to decipher the hidden message based on their secrets, while Wendy remains completely oblivious of that message.

In modern digital steganography, the message is hidden in a digital file yielding a stegotext. This broad definition is useful because any message can be embedded in any sort of digital file whether it is text, photo, or other multimedia. Because this definition does not define the method in which hiddentext is stored in a stegotext we can define any method to do the job. However, not all steganographic methods are steganographically secure. Imagine Alice sending a text file to Bob and renaming the file extension. At first glance this could seem like a reasonable message for Alice to send. Upon further investigation Wendy will be able to identify the hidden message or the fact that there is a hidden message by comparing this file to other files of that extension.

In this paper we are going to go over what is needed in a steganographic model in the first section. After understanding the model that must be used in stegosystems we will look at the notion of steganographic security. Next we will look at steganalysis, the act of analyzing suspected stegotexts to verify the suspicion or obtain the message. Before concluding, we will look at stegosystems and discuss issues that these may have in terms of our notions of security.

Steganographic Model

There are variations in steganographic models, but most papers will agree that a good model needs to include three parties and a monitored communication channel. The three active members of the model are Alice, Bob, and Wendy. Prisoners Alice and Bob are trying to hide a message, hiddentext, from Wendy the warden. The warden monitors the communication on the channel and inspects many messages on the channel. Valid channel messages are known as covertext, these are messages that should not arouse

any suspicion from Wendy. Alice will run an embedding algorithm on her hiddentext and some covertext to generate her stegotext. Bob will run an extracting algorithm to obtain the hiddentext from the stegotext. These are all of the needed pieces in steganography. If Alice and Bob are able to pass a message which includes hiddentext through this monitored channel without Wendy becoming aware of the presence of hiddentext, then Alice and Bob have successfully sent a stegotext. If Alice and Bob are not trying to hide a message then they are just sending covertext. Figure 1 illustrates steganography in a similar manner to what is described above.

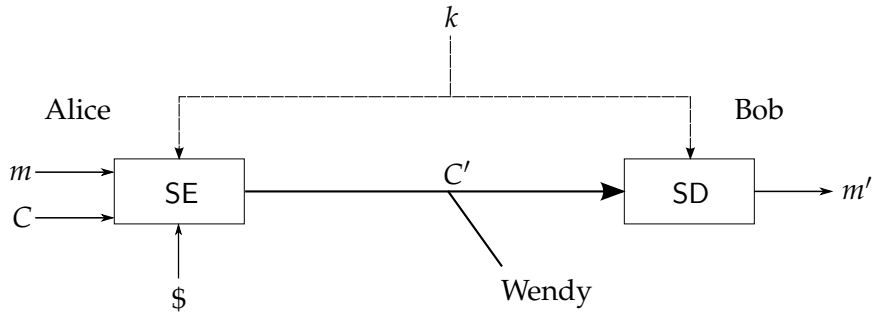

Figure 1: The basic steganographic model. Alice attempts to send message m to Bob, concealing it inside the stegotext C'.

In order for the steganographic system to function, the following elements should be present in the model:

C	Covertext: Has the potential to embed a hiddentext.
m & m'	Hiddentext: Text to embed in covertext.
SE	Embedding Algorithm: responsible for producing stegotext.
SD	Extracting Algorithm: responsible for extracting hidden text from the stegotext.
C'	Stegotext: Contains hiddentext embedded within.

The model above is also known as "The Prisoner's Problem". In order for Alice to send secret messages to Bob, she needs to hide them by using some covertext in such a way that the warden of the prison, Wendy, falls short of recognizing anything suspicious about the message. Wendy is the passive adversary in the stated case. She performs hypothesis testing to check for

anything suspicious by monitoring both the covertext and stegotext being sent over the communication channel.

The embedding algorithm SE performs the task of generating stegotext C' by combining C and m. The chief function of the extracting algorithm SD is obtaining the hidden message m' from C'. The extraction of the covertext C is of little importance at receiver end. In one case, while Alice is inactive, only the covertext C is sent over the communication channel to Bob. Embedding does not take place in this case and the passive adversary Wendy observes a rather innocuous text. In another case, whilst Alice is active, embedding takes place and the stegotext that is generated is sent over the communication channel to Bob. Both Wendy and Bob observe the stegotext C', but only Bob is able to extract the hidden message from C' with the help of the extracting algorithm SD that he possesses. In both the cases, Wendy is oblivious of whether Alice is active or not.

In the above model, an assumption has been made about Alice and Bob possessing the knowledge of the covertext and the stegotext. Moreover, the embedding algorithm must always work regardless of the information that C' holds. Another bold assumption that has been made in this case is that Bob is somehow aware of when Alice is active, even though he has no way of doing so. Although it does not affect the outcome, it does help in understanding the security properties of the steganographic system.

Steganalysis

The goal of steganography is to conceal the fact that two parties are exchanging confidential information. Unlike cryptography, where the attackers "wins" only if it deciphers and recovers the original message, an adversary against a steganography systems "wins" even if it realizes the existence of secret message, without having to recover the hidden text. Steganalysis deals with analyzing the channel to detect the presence of stego or hidden information. The primary step is to identify suspected stego media TODO: cite [1]. The suspected media then undergoes various transformation and/or modifications in order to reveal the existence of an embedded message.

The suspected media is not analyzed by itself without other factors. An adversary performing the steganalysis will also have knowledge of all communications within the channel, though an adversary may not have all that information all the time. There are two classes of steganography attacks can be used by the snooping adversary Wendy TODO: cite [1]. Wendy may employ a steganography only attack in which he attacker has access to only

the stegotext containing the hidden message. The other class of attack is the known cover attack in which the attacker has the original covertext as well as the stegotext. The specific steganalysis attack will depend on what the hiddentext is contained in. In the next sections we will look at some common steganographic applications and methods for analyzing these applications.

Security Definitions

Primitives

Let P_X be a probability mass function with support χ, where X is a discrete random variable taking the values in χ. The *entropy* of X is

$$H(X) = E(-\lg P_X)\,,$$

where $E(\cdot)$ is the expected value (weighted average) function; that is,

$$H(X) = -\sum_{x \in \chi} P_X(x) \lg P_X(x). \tag{1}$$

Intuitively, the entropy of X is a measure of the number of bits of uncertainty in X. For example, suppose χ is the set of all n-bit strings, and $P_X(x) = 1/2^n$ for any $x \in \chi$; that is, every n-bit string is equally likely to be pulled from P_X. This would represent a distribution of maximum uncertainty, and it is straightforward to show that Eqn. (1) evaluates to n in this case. In fact, $H(X) = \lg|\chi|$ is an upper bound for H, where $|\chi|$ denotes the cardinality of χ.

The *minimum entropy* of a distribution P_X is defined as

$$H_\infty(X) = \min_{x \in \chi}\{-\lg P_X(x)\} \tag{2}$$

This can be understood as a measure of uncertainty for the "most probable" element in χ according to P_X. For example, if there is some element x_0 with $P_X(x_0) = 1$, then $H_\infty(X) = 0$ (there is no uncertainty in X). Suppose the most probable element x_0 has probability $P_X(x_0) = 1/2$. Intuitively, the uncertainty is unity; that is, we can guess that the next value of X will be x_0 to within a single coin flip. Indeed, evaluating Eqn. (2) for such a distribution shows that $H_\infty(X) = 1$.

Given two probability mass functions P_X and P_Y, both with support χ, the *relative entropy*, also called the Kullback-Leibler divergence, from P_X to

P_Y is defined to be

$$D(P_X \| P_Y) = \sum_{x \in \chi} P_X(x) \lg \frac{P_X(x)}{P_Y(x)}. \tag{3}$$

Intuitively, the relative entropy is a measure of the difference between P_X and P_Y, although it is important to note that it is not symmetric; that is, $D(P_X \| P_Y) \neq D(P_Y \| P_X)$. However, $D(P_X \| P_Y) = 0$ if and only if $P_X = P_Y$, and increases indefinitely as P_Y diverges from P_X.

A *channel* as defined by TODO: cite is a distribution on timestamped bit sequences; i.e., a channel $\mathcal{C}$ is a distribution with support $\{(\{0,1\}, t_1), (\{0,1\}, t_2), \ldots\}$, where each $t_i \leq t_{i+1}$. The intent is to model communication, where not just the content but also the timing of the communication may be relevant. Since a particular draw from the distribution $\mathcal{C}$ depends the history of previously drawn bits, define $\mathcal{C}_h$ to be the distribution conditioned on history h. Furthermore, it is useful to think of drawing from the channel in chunks of b bits at a time, so define $\mathcal{C}_h^b$ to be the distribution on the next b bits after the history h.

Security Definitions

Hopper, et. al TODO: cite define a stegosystem S as a pair of randomized algorithms $(\mathsf{SE}, \mathsf{SD})$. SE takes as input a shared key k, a hiddentext message m, and a message history h, and an oracle $M(h)$ that samples according to channel distribution $\mathcal{C}_h^b$, where channels are required to satisfy, for all h drawn from $\mathcal{C}$, $H_\infty(\mathcal{C}_h^b) > 1$. As output, $\mathsf{SE}_k^M(m, h)$ returns a sequence $c_1 \| c_2 \| \ldots \| c_\ell$ in the support of $\mathcal{C}_h^{\ell b}$. The decryption algorithm $\mathsf{SD}_k^M(c_1 \| c_2 \| \ldots \| c_\ell, h)$ returns a message m, which should be "correct"; that is, the same message encoded by SE, at least $2/3$ of the time.

The authors define the security of a stegosystem in terms of a game. The adversarial warden W is given access to $M(h)$, which returns draws from $\mathcal{C}_h^b$, and an oracle $\mathcal{O}$. The oracle $\mathcal{O}$ is either SE_k or a function $O(\cdot, \cdot)$, where $O(m, h)$ simply returns a draw from $\mathcal{C}_h^{|\mathsf{SE}_k(m,h)|}$. The warden also has access to randomness r. The warden's advantage against the steganographic secrecy under chosen hiddentext attack for channel $\mathcal{C}$ of stegosystem S is defined by Hopper et. al to be

$$\mathbf{Adv}_{\mathrm{S},\mathcal{C}}^{\text{ss-cha-}\mathcal{C}}(W) = \left| \Pr_{k,r,M,\mathsf{SE}} \left[W_r^{M,\mathsf{SE}_k(\cdot,\cdot)} \text{accepts} \right] - \Pr_{r,M,O} \left[W_r^{M,O(\cdot,\cdot)} \text{accepts} \right] \right|.$$

A stegosystem S is $(t, q, \ell, \varepsilon)$-*steganographically secret under chosen hiddenttext attack* for channel $\mathcal{C}$ (SS-CHA-$\mathcal{C}$) if, for any warden W making at most q queries totaling at most ℓ bits of hiddentext, and running in time at most t,

$$\mathbf{Adv}^{\text{ss-cha-}\mathcal{C}}_{\mathsf{S},\mathcal{C}}(W) \leq \varepsilon;$$

that is, the stegosystem S is insecure if an efficient warden can (with high probability) distinguish between the output of $\mathsf{SE}_k(m, h)$ and draws from $\mathcal{C}_h^{|\mathsf{SE}_k(m,h)|}$, even when given access to $\mathcal{C}_h^b$ through M. A stegosystem S is $(t, q, \ell, \varepsilon)$-*universally steganographically secret under chosen hiddenttext attack* for channel $\mathcal{C}$ (USS-CHA-$\mathcal{C}$) if it is $(t, q, \ell, \varepsilon)$-SS-CHA-$\mathcal{C}$ for any channel $\mathcal{C}$ that satisfies, $\forall h$ drawn from $\mathcal{C}$, $H_\infty(\mathcal{C}_h^b) > 1$.

Cachin TODO: cite takes an information theoretic approach to steganographic security. He considers the basic prisoner's problem, where Alice sends either an innocent covertext or a stegotext concealing a message to Bob over an open communication line. The warden eavesdrops on the line, and must decide whether the communication is a covertext or stegotext. Let P_C be the distribution of covertexts, and let P_S be the distribution of stegotexts; these distributions are known to the warden. Cachin defines the overall security of the system in terms of the relative entropy between P_C and P_S; namely, the stegosystem is *information-theoretic perfectly secure* if

$$D(P_C \| P_S) = 0,$$

and is *information-theoretic ε-secure* if

$$D(P_C \| P_S) \leq \varepsilon.$$

Intuitively, Cachin is claiming that a stegosystem is secure if the probability distributions of covertexts and stegotexts are "close," so that, given a message in the support of P_C and P_S, the warden has very little reason to believe it was drawn from one distribution over the other. Cachin goes on to analyze the decision from the framework of hypothesis testing.

Cachin's warden is a much weaker adversary than Hopper's, and correspondingly, Hopper's SS-CHA-$\mathcal{C}$ game provides a much stronger definition of security. Some key weaknesses in Cachin's definition:

- Cachin's warden receives a single message over the communication line, and must determine her decision. In contrast, Hopper's warden has access to an oracle that can be queried as much as is computationally feasible. Indeed, Cachin's model is roughly equivalent to the SS-CHA-$\mathcal{C}$ game when only a single query is allowed.

- Cachin's warden knows the distribution of possible messages chosen by Alice, but does not know the message. Hopper, however, allows for an interactive, chosen hiddentext attack.

- Hopper's channels are all conditioned on the *history* of previously drawn samples. And since his warden is allowed to specify arbitrary history to the oracles, a stegosystem that meets SS-CHA-$\mathcal{C}$ security should be secure for any valid history of communication on $\mathcal{C}$. On the other hand, Cachin's model, because it depends on static probability distributions, does not capture this sequential element of real communication. Even if P_C and P_S were themselves dependent on history, Cachin's model would only be able to claim security of the stegosystem for the particular history leading up to his experiment.

Though Cachin's definitions are weaker, they are not without merit. Hopper's security schemes require perfect oracles—oracles that can sample from $\mathcal{C}_h$ for arbitrary h, can be "rewound," etc. Given the application, the lack of such an available oracle may make implementation of security schemes (and proof of their security) difficult or impossible. In contrast, knowledge about P_S and P_C might be easier to determine for a scheme couched in Cachin's framework.

Steganographic Applications

Over the years, various steganography techniques have been used to send secret messages, embedded in innocuous media. The applications exploit the structure of the media to conceal sensitive information. Common modern techniques of steganography can be classified as image steganography, audio and video steganography, text steganography and network steganography.

Image Steganography

Hiding information in digital images is one of the most popular steganography techniques. A digital image is a representation of pixels. Pixels are typically made of three color channels, for example red-green-blue, and each color channel will have have an intensity assigned to it. One color channel is normally 8 bits wide, but the size can vary. Changing the least significant bit (LSB) of each color channel intensity will change the resulting pixel very slightly. Steganographic applications can encode three bits of binary data

into every pixel in an image. The change will likely not be detectable by the human eye. There are various variations in this techniques. The number of bits per pixel can be varied or several pixels could be used to hide one bit of information. Sometimes a fixed number of pixels, previously decided by the two parties, are skipped to provide more obscurity.

Still image steganography is susceptible to various known-cover attacks. Analyzing the cover image and the stego image can reveal the existence of hidden message. For example, the GIF image format uses only 8-bits per pixel and changing the LSB can result in large change in the intensity of the pixel TODO: cite[2]. Cmorik and Sumák TODO: [7] demonstrate various structural and visual techniques to reveal modification of the original image. For example, changing the image pixels to a black and white scale reveals modifications in the LSB, and thus the stegotext TODO: [7].

TODO: the image

If the location of the bit(s) in an image does not change then simple attacks can run against the stegosystem and win with ease. The simplest attack would be to look in those locations when calling the oracle with the all 1's message and the all 0's message. If the location of the bit(s) is unchanging we can expect that the two locations will be different even if the coverimage is new.

Text Steganography

Text steganography involves hiding messages in plain text or text documents using certain predefined schemes like selected characters, special symbols, extra white spaces, position of tag attributes in html documents etc.

Dulera, et. al TODO: [4] explain several techniques used in text steganography. For example, the alphabet space is divided into two parts, say letters with a curvature in a shape (B, C, Q), that represent bit "0", while others (A, M, T), represent bit "1". Alice and Bob will then exchange messages by embedding a bit in the first letter of the sentences. Thus if Alice wants to send "101" to Bob, the transmitted stegotext might look something like

Today is a holiday. Burger at Wendy's. My Treat!

Such a stegosystem fails Hopper et. als definition of security immediately. A warden could launch a simple attack: query the oracle on a hiddentext of all ones, and arbitrary history. If the oracle returns a text where every sentence begins with letter without curvature, accept, otherwise, reject. In the case that the oracle is the steganographic embedding algorithm, the warden

will always accept. In the case that the oracle is not, then the probability of accepting is the probability that a draw from the channel returns such a sentence, which is unlikely for normal English communication.

Such a scheme is exemplary of the kinds of heuristic schemes that rely so heavily on obscurity and not actual security. The success of such a scheme in practice relies completely on a warden that is perhaps not overly suspicious, or is unaware of the algorithms used in the encryption scheme.

Messages have also been embedded in HTML documents using the html tag attributes. This approaches exploits the structure of an HTML document as well as the properties of the media, that is the Internet, to hide and share information. TODO: Garg[5] demonstrates various approaches of concealing information in HTML documents. For example, if an html tag has the "class" attribute before the "style," it represents bit "0," else bit "1." Thus it exploits the fact that changing the order of the attribute does not change the appearance of the document. Also the popularity of the internet and the high percentage of tags in an HTML document, allows you to communicate secret messages with ease. However, the scheme fails under chosen hiddentext attack in the same manner as above.

Good Stego

Bad Stego

Conclusion

References