Content-Aware Steganography: About Lazy Prisoners and Narrow-Minded Wardens

Richard Bergmair[1] and Stefan Katzenbeisser[2]

[1] Computer Laboratory, University of Cambridge
[2] Institut für Informatik, Technische Universität München
{rbergmair,skatzenbeisser}@acm.org

Abstract. We introduce content-aware steganography as a new paradigm. As opposed to classic steganographic algorithms that only embed information in the syntactic representation of a datagram, content-aware steganography embeds secrets in the semantic interpretation which a human assigns to a datagram. In this paper, we outline two constructions for content-aware stegosystems, which employ, as a new kind of security primitive, problems that are easy for humans to solve, but difficult to automate. Such problems have been successfully used in the past to construct Human Interactive Proofs (HIPs), protocols capable of automatically distinguishing whether a communication partner is a human or a machine.

1 Content-Aware Steganography

In his 1984 landmark paper [23], Gustavus Simmons illustrated what is now widely known as the *prisoners' problem*: Two accomplices in a crime, Alice and Bob, are arrested in separate cells. They want to coordinate an escape plan, but their only means of communication is by way of messages conveyed for them by Wendy the warden. Should 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 no one has ever escaped from. Simmons' solution to the prisoners' problem is phrased in an interesting way: Alice and Bob "will have to deceive the warden by finding a way of communicating secretly in the exchanges, i.e. of establishing a 'subliminal channel' between them in full view of the warden, even though the messages themselves contain no secret (to the warden) information" [23]. In other words, Alice is trying to convey a particular piece of information which is represented as a single *datagram*. This datagram is available to both Wendy and Bob—but it contains different *information* to Wendy than to Bob.

Informally speaking, a subliminal channel is one that transmits datagrams that have at least two possible interpretations. Each datagram is intentionally given an obvious interpretation (the cover) that is innocuous to Wendy, and a non-obvious interpretation (the secret) that is suspicious to Wendy, and thus cannot be transmitted in plain sight. The security of the stegosystem usually relies on some assumption of an advantage that Bob has over Wendy, when it comes

J. Camenisch et al. (Eds.): IH 2006, LNCS 4437, pp. 109–123, 2007.

to the *interpretation* of the message: Bob can interpret the message with regard to its secret meaning, while Wendy can only interpret the message as the cover.

In the past, many stegosystems have been constructed, most of them using images, digital audio, or video as cover. Consider for example a simplistic LSB scheme for image-based steganography in which the cleartext message is written into the LSBs of an image without any further cryptographic concealment. The datagram has an obvious interpretation, which is visual perception by a human user of the pattern that appears on screen when it is opened in their favourite image viewer. It also has a non-obvious interpretation, which is to extract the LSBs and view their concatenation, say, in a hex-editor. Under the assumption that Alice constantly sends Bob bitmap images that Wendy is not willing to wade through with a hex-editor, this simplistic system might be attributed some kind of security. However, Wendy will probably try to *automatically* analyze all datagrams exchanged between Alice and Bob to gain knowledge of a subliminal channel. This notion of automaticity in steganalysis has probably received too little attention in the past, which is why we shall, in this paper, take the challenging point of view that a stego object should not be considered perfectly secure as long as its *semantics* are prone to *automatic interpretation by a machine*.

Due to recent progress in the field of steganalysis (see for example [17]), LSB substitution techniques must be considered completely insecure today. To understand why LSB steganography was compromised, it is important to bear in mind that a bitmap image is not just a sequence of bytes, but rather a representation for some specific semantic content. It could, for example, be a vector drawing consisting of uniformly colored geometric shapes. If a set of pixels can be identified as representing, say, an oval shape colored in a certain tone of blue, and half of these pixels deviate in their color by the LSB, this might give us some evidence of steganography taking place. A 24-bit bitmap might also be a photograph taken by a digital camera with a CCD that leaves noise with special characteristics in the images [20]. If these characteristics cannot be found in the LSBs of the image, then again we have gained evidence to suspect that steganography is taking place.

We believe the way in which LSB substitution has been compromised is stereotypical for how the steganography vs. steganalysis battle is usually fought, namely by steganalysis exploiting the false assumption made by steganography that a meaningful digital object can be specified *solely* in terms of syntactic properties. Stegosystems are usually broken by exploiting *semantic* inconsistencies introduced into the cover when hiding a secret. This is a limitation which is inherent with every steganographic system that takes a cover and applies modifications in order to obtain a stego object: an attacker that possesses a more accurate semantic cover model than the embedder can break the system easily. Thus, a security vulnerability is necessarily opened in any steganographic system whose participants are computers that employ state-of-the art cover models, as soon as the state of the art improves.

In this paper, we propose an alternative view of steganography, which takes semantic aspects into account and hides information in the *semantics* (rather

than the syntactic representation) of a datagram sent over a channel. We call such systems *content-aware steganography*. At the heart of the paradigm lies the assumption that Wendy the warden is a *computer* (and not a human), while Alice and Bob are both humans. Given the massive increase in communication over the last years, this is an assumption which seems to be justified, as large-scale manual steganalysis is not possible.

A content-aware stegosystem chooses stego objects in such a way that both the human sender and receiver can easily assign a secret semantic interpretation to the transmitted datagrams, whereas for a computer (such as Wendy) it is inherently difficult to perform the same task. In extending the analogy of Alice and Bob, we may think of the prisoners as being "lazy" when sending or receiving subliminal messages: as humans they can trivially assign and infer a secret semantic interpretation to a stego object. (Thus, one can view content-aware stegosystems as implementing a special supraliminal channel [16]). On the other hand, the warden Wendy is "narrow-minded" in the sense that her inherent limitations as a data processing device do not allow her to infer the secret interpretations of stego datagrams. We have to stress at this point, that it is not the intention of the present contribution to compete with current notions of steganographic security, but rather to complement them by suggesting *content-awareness* as a new security property that should hold for a secure system in addition to the well-established ones.

Content-aware stegosystems are constructed in such a way that a successful steganalytic attack would require solving an Artificial Intelligence problem that can currently not be tackled with state-of-the-art algorithms. We will show that Human Interactive Proofs (HIPs), which were recently developed to distinguish humans from computers in security applications, readily lend themselves to the construction of such content-aware stegosystems.

The rest of the paper is organized in the following way. Section 2 gives a thorough explanation of the new steganographic paradigm we propose, motivating it from a principal and conceptual point of view and Section 3 gives a generic construction of a content-aware stegosystem which draws its security from a Human Interactive Proof. These two sections are embedded in this paper in such a way that the more technically minded reader may choose to skip them, but will still be able to follow the rest of this paper. Sections 4 and 5 introduce two practical content-aware stegosystems, one that hides steganographic content in audiovisual content and one that uses natural language texts as covers. Finally, Section 6 will review related work in light of the new paradigm.

2 On Data and Information

Traditionally, stego objects have been treated as meaningless objects, which is an assumption most probably stemming from cryptography: in the context of cryptography, access to a cryptogram leaves an eavesdropper without any knowledge. By virtue of its definition, a cryptogram does not carry any meaning beyond that which must be inferred by means of the decryption routine. A stego

object however, which has to resemble an innocuous cover in every respect, does carry such meaning. A stego object can only be identified as innocuous or suspicious after it has been *interpreted and assigned meaning*, which extends the cryptologic picture into a semantic dimension as we move on from pure cryptography to steganography.

Turning back to our intuitive picture of steganography, the essence of the new paradigm is that we are dealing with *data* in the context of cryptography, as opposed to steganography, which deals with *information*. The distinction between data and information is based on the degree of understanding an observer has about a given observation. In particular, we shall call an observation a piece of data if we see it in a purely symbolic way, void of inherent meaning but capable of being processed to make sense.

Once we commit to this conception of data and information, it becomes apparent that the role of *understanding* as a means to elevate a given observation from data to information and knowledge is quite crucial. Ackoff [1] notes that understanding is by virtue of its nature a cognitive process. It can only be automated to the degree to which computers succeed in simulating this process. Thus, any claim attributing a human level of information-processing capability to a fully computerized system must be presupposing a hypothesis whose confirmation has resisted decades of research in Artificial Intelligence: that biological cognition is a computational process. Thus we feel driven to the point of view, that computers may not be regarded as directly operating on information as such in any way. Of course, the success of computerized systems in supporting human-controlled information processing systems is undisputed. Yet, this does not contradict the view that computers are essentially limited in their domain of operation to simple data since information processing may still happen implicitly in a computerized system within the brains of its human users.

These ideas about data and information have a strong impact on data and information processing in the context of cryptography and steganography: In the new paradigm we have in mind, a joint coding and encryption scheme lies at the core of every stegosystem. The purpose of this scheme is to provide security for the transmitted *data*; in addition, it performs appropriate coding for the communication channel which is used to transmit subliminal information. In the sequel, we will refer to this core solely as the cryptosystem. In an outer layer, a steganographic operation extends the cryptosystem by semantic aspects: its purpose is to let Alice transmit meaningful pieces of *information*. The stego layer thus controls the semantic interpretation of a datagram and provides resistance against automated steganalysis.

Figure 1 depicts this idea of content-aware steganography. The inner area of the figure represents the cryptosystem: The message input to the encryption routine is treated as a piece of data. The encryption routine translates this message into a cryptogram which is another piece of data; the routines for decryption and cryptanalytic attack basically invert this mapping. The encryption routine does not need to take into account any semantics, since it can always reinterpret its input as a random choice of one element from a finite message space, regardless

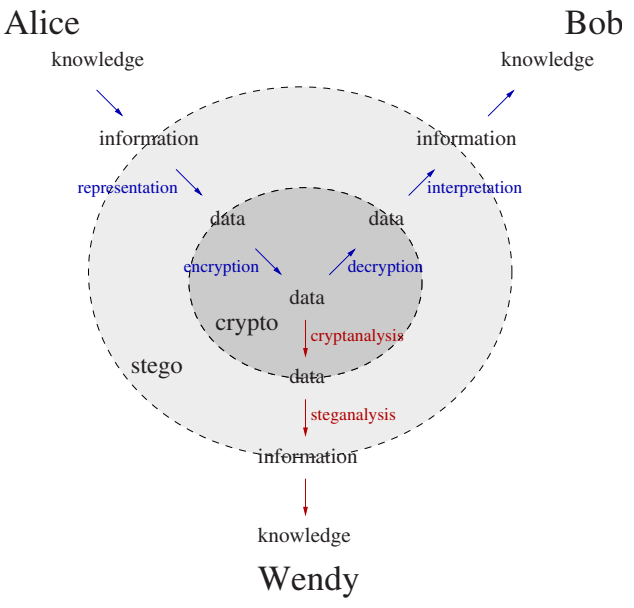

Fig. 1. Content-aware steganography

of whether this input is actually a representation for an image, a sound, or a text. The decryption routine and the cryptanalytic attack typically do not need to take into account any semantics either.

The outer area of the figure depicts the steganographic layer: The message that Alice actually wants to convey, is a piece of information. The act of representation degrades this information to data, so it can be run through the cryptosystem. The acts of interpretation or steganalysis, on the other hand *reassign meaning* to the data which is supposed to equal the original message, and therefore yield information again: the whole stegosystem essentially operates within the information domain. Clearly, the act of representation must take into account semantics, since Alice has exactly one piece of semantic content in mind when she represents it, and the acts of interpretation and steganalysis have to deal with semantics, since they have to reconstruct exactly that semantic content. The crucial requirement is that Wendy is unable (even after performing cryptanalytic attacks on the transmitted *data*) to correctly infer the secret semantics of the datagrams transmitted over the channel.

3 HIP: A New Security Primitive for a New Kind of Steganography

In this section, we propose a general construction for a content-aware stegosystem out of any Human Interactive Proof (HIP). Once we admit that Wendy is a computer and Bob is a human sitting in front of a computer, all we have to do

is to make the solution to the problem of determining the secret interpretation of the stego object depend on the solution of a problem that only humans can solve correctly.

Human Interactive Proofs (HIPs) [19,31,25], better known under the more specific model of CAPTCHAs (Completely Automated Public Turing tests to tell Computers and Humans Apart) [26], have only recently gained attention in the computer security community because of their usefulness in the fight against worms and spam and the prevention of web-service abuse, denial-of-service, and dictionary attacks. Essentially, an HIP allows a computer program to determine whether it interacts with another computer or a human. HIPs are based on complex Artificial Intelligence problems which computers cannot solve with the same speed and accuracy as humans.

Currently the best-known HIPs are OCR CAPTCHAs that display heavily distorted text to a user and ask them to type the text into an input field. Typically, humans have no problem in performing this task while an automated solution requires solving the complex problem of optical character recognition, which is still unsolved for heavily distorted text. The underlying assumption of the OCR CAPTCHA is that once a communication partner solves this challenge correctly, one can safely assume that it is a human.

for $k := 1, \ldots, n$ **do**
  The tester constructs a test/solution pair (t_k, s_k)
    such that $t_k \in T$ and $s_k \in S$
  The tester sends the test t_k to the testee
  The testee makes a choice h_k for a solution of t_k
  The testee sends h_k to the tester

  // The tester checks if testee could be a computer
  if $h_k \neq s_k$ **then**
    Do not draw any conclusions and stop
  end
Conclude that the testee is human

Fig. 2. n-round Human Interactive Proof

In general, a Human Interactive Proof involves a set of tests $T = \{t_1, t_2, ...\}$, a set of solutions $S = \{s_1, s_2, ..., s_{|S|}\}$, for $|S| \in \mathbb{N} \setminus \{0, 1\}$, and an algorithm that produces a random test/solution pair (t, s) where $t \in T$ and $s \in S$; everyone who answers s to t is considered to be a human. In theory, for an HIP to be secure, T must be countably infinite at least (otherwise there exists an algorithm that already contains the solutions to all problems hardcoded in the program file). In practice it is desirable that $|T|$ is as large as possible. We will assume that for each test $t \in T$ there is a set $C_t \subseteq S$ of candidate-solutions for t, which includes the correct solution s to t and a number of invalid solutions (thus, $|C_t| \geq 2$ for all tests t). Let $I_{C_t} : C_t \mapsto \{0, 1, ..., |C_t| - 1\}$ be a one-to-one mapping from the elements of a given set of candidate solutions to the smallest $|C_t|$ natural

for $k := 1, \ldots, n$ **do**
  Alice constructs a test/solution pair (t_k, s_k)
    such that $t_k \in T$ and $s_k \in S$
  Alice constructs a claim
    $c_k \leftarrow I^{-1}_{C_{t_k}}((I_{C_{t_k}}(s_k) + \overline{m}_k) \bmod |C_{t_k}|)$
  Alice sends the test/claim pair $\overline{e}_k = (t_k, c_k)$ to Bob
  Bob makes a choice h_k for a solution of t_k
  Bob computes $\overline{m}'_k \leftarrow (I_{C_{t_k}}(c_k) - I_{C_{t_k}}(h_k)) \bmod |C_{t_k}|$

Fig. 3. Content-aware stegosystem

numbers. For the sake of simplicity, we assume that all tests $t \in T$ have the same number b of candidate solutions, i.e. $|C_t| = b$ for all C_t.

Figure 2 shows how a Human Interactive Proof is performed. The tester enters a loop and constructs n test instances t_k together with respective solutions s_k. The tester shows the instances t_k to the testee. The testee provides solutions h_k for all instances; finally the testee is verified to be a human if they responded with the expected solutions in all n rounds (i.e., $h_k = s_k$ for $k = 1, \ldots, n$).

A secure Human Interactive Proof can be used as central primitive to construct content-aware stegosystems. In particular, we make the assumption that *sending a test instance of an HIP over a channel is not per se suspicious*. This assumption, which must be verified for each instantiation of the general construction presented in this section, is a direct extension of the general assumption of classic steganography that sending, for instance, images or pieces of literary text does not itself raise the awareness of Wendy. In practice we could, for example, assume that Wendy generally tolerates English language text being exchanged between Alice and Bob. We can then set up a stegosystem on the basis of a text-domain HIP, such as the word-sense disambiguation HIP [6]. Alternatively we could assume that Wendy tolerates images being exchanged. We would then use an image HIP such as the famous OCR CAPTCHA [26] or image recognition CAPTCHAs [14]. Sections 4 and 5 will discuss these two concrete constructions.

The general construction of a content-aware stegosystem from an HIP is shown in Figure 3. Once Alice wants to send a piece of information m to Bob, she fixes a datagram representation of m as an integer sequence of length n with elements between 0 and $b-1$, i.e., $\overline{m} = \overline{m}_1\overline{m}_2...\overline{m}_n$, where $\overline{m}_i \in \{0, 1, ..., b-1\}$. One can think of $\overline{m}$ as the radix-b expansion of a natural number smaller than b^n. Note that the construction can be straightforwardly generalized to the case of differing numbers of candidate-solutions $|C_t|$ by thinking of $\overline{m}$ as a mixed-radix expansion.

To send the message, Alice constructs n test instances t_k of the HIP together with corresponding solutions s_k. In addition, she constructs a *claim* which corresponds to a (possibly incorrect) solution to t_k, called c_k, computed as

$$c_k \leftarrow I^{-1}_{C_{t_k}}((I_{C_{t_k}}(s_k) + \overline{m}_k) \bmod |C_{t_k}|).$$

Thus, Alice uses the map $I_{C_{t_k}}$ to obtain the numerical representation of s_k and adds $\overline{m}_k$ to it; subsequently, she uses the inverse mapping to map the result back to a candidate solution. Finally, Alice sends both t_k and c_k to Bob. One can think of that as Alice claiming c_k to be the solution to t_k. If Bob is able to compute the correct solution to t_k (i.e., solve the HIP), he can reconstruct the secret message $\overline{m}$ precisely and thus can gain an understanding of the information m Alice sent.

Claim 1. (Decodability by humans) *Suppose that Bob is human and is thus able to solve all instances of the HIP correctly. After termination of the steganographic transmission, the message* $\overline{m}' = \overline{m}'_1\overline{m}'_2...\overline{m}'_n$ *received by Bob will be equal to the original message* $\overline{m}$ *submitted by Alice.*

Proof sketch: Consider the stego transmission of the k-th symbol. Since Bob is human, he is able to choose h_k in such a way that $h_k = s_k$ (otherwise he would fail to pass the HIP and thus not be considered human). Bob reconstructs the k-th message element by setting $\overline{m}'_k = (I_{C_{t_k}}(c_k) - I_{C_{t_k}}(h_k)) \bmod |C_{t_k}|$. Substituting c_k and letting $s_k = h_k$ results in $\overline{m}'_k = (I_{C_{t_k}}(I^{-1}_{C_{t_k}}((I_{C_{t_k}}(s_k) + \overline{m}_k) \bmod |C_{t_k}|)) - I_{C_{t_k}}(s_k)) \bmod |C_{t_k}|$, yielding to $\overline{m}'_k = \overline{m}_k \bmod |C_{t_k}|$. Since $\overline{m}_k < |C_{t_k}|$, we have $\overline{m}'_k = \overline{m}_k$, which means that Bob has correctly decoded the message. □

We now argue that the steganalysis problem for Wendy is hard. As mentioned above, at this point we rely on the general assumption that Wendy will find the transmission of HIP instances, i.e. the tuples (t_k, c_k) suspicious neither by themselves nor in the transmitted sequence; thus we assume the existence of an appropriate encoding function such that transmission of the coded tuples will be considered innocuous. This assumption must, of course, be verified in practice on a case-by-case basis. (In the subsequent sections we will outline two such encodings for a linguistic and an audiovisual HIP).

Wendy may apply cryptanalytic methods on the datagrams sent between Alice and Bob. These techniques may result in a "suspicion" $\overline{w}$, i.e., a datagram that she believes was exchanged covertly. However, due to our limited understanding of the underlying AI problem, Wendy, being a computer, will not be able to recover the sent datagram $\overline{m}$. The next claim asserts that if $\overline{m} = \overline{w}$, Wendy could pass the HIP, which contradicts the security of the HIP.

Claim 2. (Content-awareness) *Suppose that, after termination of the steganographic transmission, Wendy's suspicion* $\overline{w}' = \overline{w}'_1\overline{w}'_2...\overline{w}'_n$ *will be equal to the original message* $\overline{m}$ *submitted by Alice. Then Wendy would pass the HIP on the instances submitted over the channel.*

Proof sketch: We assume that Wendy has managed to guess $\overline{w}_k$ in such a way that $\overline{w}_k = \overline{m}_k$. Wendy can use that message to obtain a solution s'_k to the HIP instances t_k by letting $s'_k = I^{-1}_{C_{t_k}}((I_{C_{t_k}}(c_k) - \overline{w}_k) \bmod |C_{t_k}|)$. To see that this is really a solution to the HIP, we can substitute c_k and $\overline{m}_k = \overline{w}_k$ to obtain $s'_k = I^{-1}_{C_{t_k}}((I_{C_{t_k}}(I^{-1}_{C_{t_k}}((I_{C_{t_k}}(s_k) + \overline{m}_k) \bmod |C_{t_k}|)) - \overline{m}_k) \bmod |C_{t_k}|)$. This finally yields

$s'_k = I^{-1}_{C_{t_k}}(I_{C_{t_k}}(s_k) \bmod |C_{t_k}|)$ and thus $s'_k = s_k$. This means that Wendy can solve the HIP on those instances used to transmit the subliminal message. □

4 An Audiovisual Content Recognition Stegosystem

In order to show how the generic construction can be applied to a particular domain, we will develop in this section a stegosystem based on an audiovisual content-recognition HIP, similar to the image recognition CAPTCHA in [14]. We will first describe this HIP, and subsequently show how to turn this HIP into a content aware stegosystem according to the general construction of Section 3.

The original setup of the recognition HIP is as follows: Alice has available a database of images and/or sound files labelled by a human according to its semantic content. Alice could, for instance, use her digital album, containing photos, images and videos from her holidays, all of which were carefully hand-labelled by her in order to more easily find them on her computer; a label could, for instance, be "Charlie and me visiting Alcatraz". We assume that this database is private to Alice. Alice could, for instance, add new content to her album at a faster rate than she "uses them up" by publishing the content or using it for a HIP.

The HIP operates in the following manner. The test instance of the HIP consists of a number of items from her media database, including labels, which can either be incorrect or correct. Alice generates a test/solution-pair (t, s) by picking 8 media files from the database, and relabelling them to obtain a sequence

$$t = \langle (x_1, l_1), (x_2, l_2), ..., (x_8, l_8) \rangle,$$

where each x_i is a multimedia file and l_i is a label. One of the media files is correctly labeled with a label from Alice's private database, whereas all other labels are randomly picked (e.g., taken from different media items in the database). Alice presents the test instance t to Wendy, who has to choose the index of the correctly labeled file. If she passes the test, she is considered human (the test can be repeated in order to increase the level of confidence).

This HIP can be converted into a content-aware stegosystem by using the general methodology outlined in Section 3. First we need a way to encode the test instances of the HIP in such a way that they are not, by themselves, suspicious. For this purpose, Alice could generate an e-mail with a body that reads like "Hello Bob! Here are the images of my last vacation, you wanted to see.", and a number of attachments containing image files named after their labels. She can easily write this message body herself, so this need not be automated.

The receiver Bob solves the HIP and thus is able to find the index of the attachment which is correctly labelled. The only remaining issue for Bob is to recover the secret message bit from the message index. This can be done quite easily. For instance Bob could order the media files alphabetically according to their label and let the index of the correctly labelled file represent three bits from the message. Since Alice is free in her choice about the labels she can easily

choose them in a way that lets Bob recover the message she wants to transmit. However, Wendy cannot tell which of the images bears the correct label, since that would require her to *understand* the label, and *recognize* the content of the media-file.

This process can, of course, be iterated by sending more images (and arranging them in blocks of 8), which makes it possible to transmit messages that are longer than three bits. Additional unnecessary images could be used to conceal the length of the secret message.

5 A Linguistic Text Understanding Stegosystem

Recognition of audiovisual content is not the only AI-problem that lends itself to HIPs and consequently to content-aware steganography. In this section we will outline a content-aware stegosystem [5] that is based on a word-sense disambiguation HIP [6]. It can only be broken by an arbitrator who can understand the meaning of natural language text, which is an Artificial Intelligence problem that cannot be solved satisfactorily yet. Again, we start by outlining how Alice can use this problem to carry out an HIP to test whether Wendy is human, and then go on to extend the HIP to a stegosystem that enables her to pass on secret information to Bob if Wendy is a computer.

The HIP uses natural language sentences as test instances. Alice constructs a test-instance by writing down a sentence like

The radio station didn't want to ***send*** *the song.*

She designates one word within this sentence, which she looks up in a synonymy-dictionary like WordNet. This dictionary contains sets of words which can be used interchangeably in some context; note that these synonymy sets are not disjoint, as one word can have several different meanings depending on the context. For example, looking up the word *send* will give Alice information of the following form:

$$\begin{aligned}\mathrm{syn}(\mathit{send}, c_1) &= \{\mathit{air}, \mathit{broadcast}, \mathit{send}\}\\ \mathrm{syn}(\mathit{send}, c_2) &= \{\mathit{send}, \mathit{ship}, \mathit{transport}\}\\ \mathrm{syn}(\mathit{send}, c_3) &= \{\mathit{mail}, \mathit{post}, \mathit{send}\}\end{aligned}$$

Depending on the linguistic context c_s the word *send* is used in, it can be considered synonymous to a different set of words. Sometimes, *send* can be replaced by *broadcast* and sometimes by *post*. Since Alice is human and can fully understand the sentence, she can identify the context the word *send* is used in (in this example context c_1).

She now presents the sentence to Wendy, who has access to the synonymy set database, and asks her to choose the correct sense of the selected word from the database. If Wendy is a computer she will not know that a song can be aired or broadcast, but not shipped or transported by a radio station. Therefore all

Wendy can do here is to make a guess: If she could reliably choose the right set of replacements, she would have solved the problem of word-sense disambiguation, which has been of considerable interest to computational linguists ever since the first attempts at automatic machine translation were made in the 1950s. To this day, the performance of machines in word-sense disambiguation is nowhere near the performance of humans.

This HIP can be turned into a content-aware stegosystem in the following way. To transmit a secret message, Alice uses a natural language sentence as cover and selects one word (which is contained in at least two synonymy sets) in the sentence as before. The position of the word to be replaced is determined according to a key shared between Alice and Bob. The selected word is then replaced by a different word from an *incorrect* synonymy set; the replacement is selected according to the secret message. Alice could simply sort the elements of a synset in alphabetical order and assign the resulting indices as codewords. For example, by replacing the word *send* by words from the synonymy set c_2 we can encode one bit of information:

$$\textit{The radio station didn't want to} \left\{ \begin{array}{ll} - & \textit{send} \\ 0 & \textit{ship} \\ \mathbf{1} & \boldsymbol{\textit{transport}} \end{array} \right\} \textit{the song.}$$

Since Bob is a human, he can easily spot the incorrect word in the sentence. By looking up the index of the word in the shared synonymy set database, the secret message can be recovered. However, Wendy will be unable to distinguish a correct sentence from a sentence carrying secret information, as this would require her to solve the problem of word-sense disambiguation.

Again, it should be easy to embed a sequence of such HIP test-instances into an innocuous cover. This will simply be a sequence of sentences, i.e. natural language text, that can hardly be considered suspicious in itself.

6 Related Work

So far we have introduced from a conceptual side the paradigm of content-aware steganography, and have presented two examples of what an actual stegosystem based on this paradigm could look like. In this section we will discuss some stegosystems developed in the past, and analyze them from the point of view of content-aware steganography. In particular, we shall be interested in linguistic stegosystems.

The most widely cited contribution to linguistic steganography is perhaps that of Peter Wayner, who studied the use of n-gram language models [27] and probabilistic context-free grammars [28] as statistic language models by which an arbitrator identifies messages as containing natural-language. The assumption is that such data will generally be accepted by the warden, and therefore the same language model can be used to generate innocent looking stego objects.

Although Wayner's work is an important theoretical contribution to the field, his techniques cannot be directly applied to mimic natural language, since neither

n-gram models nor probabilistic context-free languages can be specified that handle languages remotely comparable in complexity to natural languages such as English. Practical techniques will therefore generally have to trade off some encoding efficiency, for example by using an embedding scheme where only single words in an innocuous piece of text are replaced by synonyms. This is what the systems by Chapman et al. [10,11,9,13,12], Winstein [29,30], and Bolshakov et al. [7,8] do. These systems basically suffer from the problem of word-sense ambiguity. Therefore they will make some substitutions that a human would never make, and will never make some other substitutions that a human would make. Other systems for linguistic steganography proposed in the past include those by Atallah et al. [2,3,24,4], by Chiang et al. [15], Nakagawa et al. [21], and Niimi et al. [22].

Another interesting variant was put forward by Grothoff et al. [18]. They proposed a stegosystem that mimics the output of statistic machine translation systems under the assumption that the arbitrator accepts such text. If we admit such an assumption, then, in our opinion, such a system should not be considered linguistic steganography any more, since all the languages that play a role in the steganographic protocol are then artificial. On the other hand, one might want to question this assumption. In this case it is important to note that the steganographic encoder used is essentially a statistical machine translation system itself: It operates on text that is publicly available in some language. The encoder translates the text into another language, embedding a secret along the way. The assumption that such output from a statistical machine translator is acceptable to Wendy can be motivated only by assuming that Wendy is cooperative, in that she wants to permit such a translator to be used somewhere in the channel between Alice and Bob. However, Wendy may also want to prohibit such traffic, and require Alice to send the source-text, and Bob to run the translator. Similarly, Wendy might whitelist a number of translations resulting from widely used standard-software and prohibit other translations from being exchanged. In our opinion the assumption that Wendy accepts poorly translated text should therefore be dropped, and the system should be considered as a linguistic stegosystem instead. However, in this case the system becomes conceptually very similar to Wayner's original scheme, except that hidden Markov models are used as language models, rather than probabilistic context-free grammars.

If we turn back to Wayner's original framework, we can highlight a number of vulnerabilities that should become obvious, once a content-aware point of view is taken. The natural language text which is assumed by Wendy as innocuous is generated and interpreted by humans. However the stegosystem generates and interprets messages by means of, say, an n-gram model, although n-gram models are not necessary and not sufficient as generators for the natural language actually spoken by humans. They generate sentences a human would never produce, and will never generate some sentences that a human would produce. Both of these clues, if observed by the arbitrator a statistically significant number of times, can, in principle, be used to break the scheme, since every piece of text

produced by the system comes from a well-known meta-model. The language model itself can be drawn from the meta-model by means of language learning techniques. N-grams can be learned by counting the occurrences of n-tuples of words (as done in code-breaking of substitution ciphers), Markov models can be learned by counting state-transitions in a finite-state automaton, and probabilistic context-free languages can be learned by counting rule applications in context-free derivations. It can be seen that these possible exploits display a universal pattern: as soon as a steganographic generator uses a computational language model to generate stego-objects, the model can be learned from data, and therefore the system can eventually be broken.

This supports the point of view that served as the conceptual point of departure in this paper: There are only two possible ways in which a linguistic stegosystem can be perfectly secure: (1) The system is content-unaware and therefore requires that Alice and Bob have a *perfect semantic model* that generates all and only the messages also generated by humans. However, this is hardly achievable. (2) The system is content-aware, and thereby turns the tables, so that it is now Wendy who must have access to a perfect semantic model during steganalysis. This can be done, as outlined before, by having humans take part in embedding and extracting the secret.

7 Conclusion

In this paper we have introduced the concept of content-aware steganography as a new paradigm of steganography, stemming from a shift in perspectives towards the objects of steganography. We pointed out that, in the predominant paradigm of steganography, the nature of these objects is that of data. We departed from the observation that systems relying on this paradigm are eventually broken on grounds of attacks that exploit the fact that the digital objects we encounter in everyday life are more than data—that they are meaningful and can be interpreted to give us information. This led us to abandon the point of view that steganographic objects can be characterized in terms of the data that represent them, and to take the new point of view that steganographic objects should be considered pieces of information as such.

To overcome the limitations of current steganographic systems, we introduced content-aware steganography, which hides secret messages in the semantic interpretation of a datagram. Finally, we introduced new content-aware steganographic algorithms that rely on Human Interactive Proofs as a security primitive: the steganalysis problem of the introduced schemes is directly related to a problem considered hard in Artificial Intelligence.

Acknowledgements. We would like to thank the anonymous reviewers for their suggestions on improving an earlier version of the paper. Richard Bergmair gratefully acknowledges financial support by an EPSRC studentship and a Cambridge European bursary and would like to thank the benefactors who made this possible.

References

1. Ackoff, R.L.: From data to wisdom. Journal of Applied Systems Analysis 16, 3–9 (1989)
2. Atallah, M.J., Raskin, V., Crogan, M., Hempelmann, C., Kerschbaum, F., Mohamed, D., Naik, S.: Natural language watermarking: Design, analysis, and a proof-of-concept implementation. In: Moskowitz, I.S. (ed.) Information Hiding: Fourth International Workshop. LNCS, vol. 2137, pp. 185–199. Springer, Heidelberg (2001)
3. Mikhail J. Atallah, Victor Raskin, Christian F. Hempelmann, Mercan Topkara, Radu Sion, Umut Topkara, and Katrina E. Triezenberg. Natural language watermarking and tamperproofing. In Fabien A. P. Petitcolas, editor, *Information Hiding: Fifth International Workshop*, volume 2578 of *Lecture Notes in Computer Science*, pages 196–212. Springer (October 2002)
4. Bennett, K.: Linguistic steganography: Survey, analysis, and robustness concerns for hiding information in text (May 2004)
5. Bergmair, R.: Towards linguistic steganography: A systematic investigation of approaches, systems, and issues. final year project, April 2004 submitted in partial fulfillment of the degree requirements for B.Sc (Hons.) to the University of Derby (2004)
6. Bergmair, R., Katzenbeisser, S.: Towards human interactive proofs in the text-domain. In: Zhang, K., Zheng, Y. (eds.) ISC 2004. LNCS, vol. 3225, Springer, Heidelberg (2004)
7. Bolshakov, I.A.: A method of linguistic steganography based on collocationally-verified synonymy. In: Fridrich, J.J. (ed.) IH 2004. LNCS, vol. 3200, pp. 180–191. Springer, Heidelberg (2004)
8. Calvo, H., Bolshakov, I.A.: Using selectional preferences for extending a synonymous paraphrasing method in steganography. In: Sossa Azuela, J.H. (ed.) Avances en Ciencias de la Computacion e Ingenieria de Computo - CIC'2004: XIII Congreso Internacional de Computacion, pp. 231–242 (October 2004)
9. Chapman, M.: Hiding the hidden: A software system for concealing ciphertext as innocuous text. Master's thesis, University of Wisconsin-Milwaukee (1997)
10. Chapman, M., Davida, G.I.: Nicetext system official home page. `http://www.nicetext.com`
11. Chapman, M., Davida, G.I.: Hiding the hidden: A software system for concealing ciphertext in innocuous text. In: Han, Y., Quing, S. (eds.) ICICS 1997. LNCS, vol. 1334, pp. 11–14. Springer, Heidelberg (1997)
12. Chapman, M., Davida, G.I.: Plausible deniability using automated linguistic steganography. In: Davida, G., Frankel, Y. (eds.) InfraSec 2002. LNCS, vol. 2437, Springer, Heidelberg (2002)
13. Chapman, M., Davida, G.I., Rennhard, M.: A practical and effective approach to large-scale automated linguistic steganography. In: Davida, G.I., Frankel, Y. (eds.) ISC 2001. LNCS, vol. 2200, Springer, Heidelberg (2001)
14. Chew, M., Tygar, J.D.: Image recognition CAPTCHAs. In: Zhang, K., Zheng, Y. (eds.) ISC 2004. LNCS, vol. 3225, Springer, Heidelberg (2004)
15. Chiang, Y.-L., Chang, L.-P., Hsieh, W.-T., Chen, W.-C.: Natural language watermarking using semantic substitution for chinese text. In: Kalker, T., Cox, I.J., Ro, Y.M. (eds.) IWDW 2003. LNCS, vol. 2939, pp. 129–140. Springer, Heidelberg (2004)
16. Craver, S.: On public-key steganography in the presence of an active warden. In: Aucsmith, D. (ed.) IH 1998. LNCS, vol. 1525, pp. 355–368. Springer, Heidelberg (1998)

17. Fridrich, J., Goljan, M., Hogea, D., Soukal, D.: Quantitative steganalysis of digital images: estimating the secret message length. Multimedia Systems 9, 298–302 (2003)
18. Grothoff, C., Grothoff, K., Alkhutova, L., Stutsman, R., Atallah, M.: Translation-based steganography. In: Barni, M., Herrera-Joancomarti, J., Katzenbeisser, S., Pérez-González, F. (eds.) Information Hiding, 7th International Workshop (IH 2005), Barcelona, Spain. LNCS, vol. 3727, pp. 219–233. Springer, Heidelberg (2005)
19. Hopper, N.J., Blum, M.: Secure human identification protocols. In: Advances in Crypotology, Proceedings of Asiacrypt '01 (2001)
20. Nasir Memon Mehdi Kharrazi, Husrev T.Sencar. Blind source camera identification. In: Proceedings of the National Conference on Image Processing (ICIP '04) (2004)
21. Nakagawa, H., Sampei, K., Matsumoto, T., Kawaguchi, S., Makino, K., Murase, I.: Text information hiding with preserved meaning – a case for japanese documents. IPSJ Transaction 42(9), 2339–2350 (2001)
22. Niimi, M., Minewaki, S., Noda, H., Kawaguchi, E.: A framework of text-based steganography using sd-form semantics model. IPSJ Journal, 44(8) (August 2003)
23. Simmons, G.J.: The prisoners' problem and the subliminal channel. In: Advances in Cryptology, Proceedings of CRYPTO '83, pp. 51–67 (1984)
24. Topkara, M., Taskiran, C.M., Delp, E.J.: Natural language watermarking. In: Delp, E.J., Wong, P.W.(eds) Security, Steganography, and Watermarking of Multimedia Contents VII, vol. 5681 (January 2005)
25. von Ahn, L., Blum, M., Hopper, N.J., Langford, J.: HIPs. `http://www.aladdin.cs.cmu.edu/hips/`
26. von Ahn, L., Blum, M., Hopper, N.J., Langford, J.: CAPTCHA: using hard ai problems for security. In: Advances in Cryptology, Eurocrypt 2003. LNCS, vol. 2656, pp. 294–311. Springer, Heidelberg (2003)
27. Wayner, P.: Mimic functions. Cryptologia XVI/3, 193–214 (1992)
28. Wayner, P.: Strong theoretical steganography. Cryptologia XIX/3, 285–299 (1995)
29. Winstein, K.: Lexical steganography. `http://alumni.imsa.edu/~keithw/tlex`
30. Winstein, K.: Lexical steganography through adaptive modulation of the word choice hash. `http://alumni.imsa.edu/~keithw/tlex/lsteg.ps`
31. Xerox PARC. In: First Workshop on Human Interactive Proofs (January 2002)