Information Security Journal: A Global Perspective
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t768221795

Notestega: Notes-based Steganography Methodology

Abdelrahman Desoky [a]
[a] Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, USA

Online Publication Date: 01 January 2009

To cite this Article Desoky, Abdelrahman(2009)'Notestega: Notes-based Steganography Methodology',Information Security Journal: A Global Perspective,18:4,178 — 193
To link to this Article: DOI: 10.1080/19393550903076841
URL: http://dx.doi.org/10.1080/19393550903076841

Information Security Journal: A Global Perspective, 18:178–193, 2009

ISSN: 1939-3555 print / 1939-3547 online
DOI: 10.1080/19393550903076841

Taylor & Francis
Taylor & Francis Group

Notestega: Notes-based Steganography Methodology

Abdelrahman Desoky
Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, USA

ABSTRACT The wide use of Notes in business, science, education, news, etc., renders Notes attractive steganographic carriers and allows the communicating parties to establish a covert channel that is capable of transmitting messages in an unsuspicious way. The presented Notes-based Steganography Methodology (Notestega) takes advantage of the recent advances in automatic notetaking techniques to generate a text cover. Notestega neither exploits noise (errors) to embed a message nor produces a detectable noise. Instead, it pursues the variations among both human notes and the outputs of automatic notetaking techniques to conceal data. Virtually, it is accomplished in three steps. First, Notestega generates a number of legitimate various notes of the same input. Second, based on a predetermined protocol it picks a particular note, for example, note number 1, 2, or 5. Third, Notestega substitutes some of the text in the selected note with another text taken from the unpicked notes. Such text substitution is carefully done to avoid the introduction of a suspicious pattern while embedding a message. Unlike machine translation and automatic summarizer, automatic notetaking can embed nondirectly related elements to its output including linguistic elements, for example, sentences, words, or abbreviations, and nonlinguistic elements, for example, lines, stars, arrows, or symbols, and thus the generated note-cover (text-cover) has ample room of concealing data. The presented implementation and steganalysis validation of Notestega demonstrate distinct capabilities of achieving the steganographic goal, adequate room for concealing data, and a superior bitrate to contemporary text steganography approaches, which is roughly 7.777%.

KEYWORDS information hiding, linguistic steganography, steganography, test steganography

INTRODUCTION

Steganography is the science and art of camouflaging the presence of covert communications. The origin of steganography is traced to early civilizations (Kipper, 2004; Davern & Scott, 1995). Ancient Egyptians communicated covertly using the Hieroglyphic language, a series of symbols representing a message. The message looks as if it is a drawing of a picture although it may contain a hidden message that only a specific person who knew what to look

Address correspondence to Abdelrahman Desoky, Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, MD.
E-mail: abd1@umbc.edu

for can detect. The Greeks also used steganography, "hidden writing," from which the name was derived. Fundamentally, the steganographic goal is not to hinder the adversary from decoding a hidden message but to prevent an adversary from suspecting the existence of covert communications (Johnson & Katzenbeisser, 1999). When using any steganographic technique, if suspicion is raised, the goal of steganography is defeated regardless of whether or not a plaintext is revealed (Johnson & Katzenbeisser, 1999; Kessler, 2004; Martin, Shapiro, & Seroussi, 2005). Contemporary approaches are often classified based on the steganographic cover type into image, audio, graph (Desoky & Younis, 2008; Desoky & Younis, 2006), or text. When linguistics is employed for hiding data and generating the steganographic cover, an approach is usually categorized as linguistic steganography to distinguish it from nonlinguistic techniques, for example, image or audio. Linguistic steganography has become more favorable in recent years since the size of nonlinguistic-covers is relatively large and is burdening the traffic of covert communications (Martin, Shapiro, & Seroussi, 2005; Petitcolas, 1999; Bennett, 2004).

Most of the published steganography approaches hide data as noise in a cover that are assumed to look innocent. For example, the encoded message can be embedded as an alteration of a digital image or an audio file without noticeable degradation (Martin, Shapiro, & Seroussi, 2005; Petitcolas, 1999). Another example is hiding a message in a text-cover by modifying the format and style of an existing text (Johnson & Katzenbeisser, 1999; Bennett, 2004; Shirali-Shahreza & Shirali-Shahreza, 2006). However, such alteration of authenticated covers can raise suspicion, and the message is detectable regardless of whether or not a plaintext is revealed (Petitcolas, 1999; Bennett, 2004). The same applies to hiding the data in unused or reserved space for systems software, for example, the designated storage area of an operating system, the file headers on a hard drive (Anderson, Needham, & Shamir, 1998; ScramDisk, 2008), or in the packet headers of communication protocols, for example, TCP/IP packets transmitted across the Internet (Handel & Sandford, 1996). These techniques are vulnerable to distortion attacks (Kessler, 2004; Petitcolas, 1999).

A similar argument is made in the literature about linguistic steganography approaches such as null cipher (Kahn, 1996), mimic functions (Wayner, 1992; Wayner 2002), NICETEXT and SCRAMBLE (Chapman & Davida, 1997; Chapman & Davida, 2007; Chapman et al., 2001; Chapman & Davida, 2002), translation-based (Grothoff et al., 2005; Grothoff et al., June 2005; Stutsman et al., 2006), confusing approach (Topkara, Topkara, & Atallah, 2007), and abbreviation-based (Shirali-Shahreza et al., 2007). The vulnerability and concerns of these linguistic approaches, as explained in Section 2, can be summarized as follows. First, the linguistic cover may introduce detectable flaws (noise) such as incorrect syntax, lexicon, rhetoric, grammar, etc., when generating a text-cover. Obviously, such flaws can raise suspicion about the presence of covert communications. Second, the content of the cover may be meaningless and semantically incoherent and thus may draw suspicion. Third, the bitrate is very small. Since there is a limit on how many flaws a document may typically have, very large documents will be needed to hide few bytes of data. In fact, this applies to nonlinguistics approaches as well. Fourth, the bulk of efforts have been focused on how to conceal a message and not on how to conceal the transmittal of the hidden message. In other words, the establishment of a covert communication channel has not been an integral part of most approaches found in the literature. Fifth, while these approaches may fool a computer examination, they often fail to pass human inspection. A successful linguistic steganography approach must be capable of passing both computer and human examinations. These concerns have motivated the development of the Notes-based Steganography Methodology (Notestega), introduced in this paper.

Notestega overcomes the issues just mentioned above by manipulating the popular textual notes to camouflage both a message and its transmittal. Basically, Notestega exploits the variations among textual notes to conceal data by substituting notes' elements in a particular note. Such note can be fabricated in order to embed data without generating any type of suspicious pattern. The main advantages of Notestega are as follows. First, the high demand for using textual notes by a wide variety of people creates a high volume of traffic and averts suspicion in the presence of covert communication channels. Second, Notestega does not imply a particular pattern (noise) that an adversary may seek. Third, the concealment process of Notestega has no effect on the linguistics of the generated cover (note-cover). Therefore, a note-cover is linguistically legitimate and is thus capable of passing

both computer and human examinations. Fourth, Notestega can be applied to all languages. Fifth, textual notes have plenty of room for concealing data, as demonstrated later in the paper. The observed average bitrate. up to 7.777% in the current implementation experiments, is superior to all contemporary linguistic steganography approaches found in the literature. Sixth, Notestega is resilient to popular attacks and the hidden message is anti-distortion. Since the reuse and alteration of textual notes are a common practice can also pass comparison attacks. The implementation and steganalysis validation demonstrate that Notestega methodology is capable of achieving the steganographic goal.

The remainder of this paper is organized as follows. Section 2 discusses the related work and compares Notestega to the linguistic steganography techniques found in the literature. Section 3 explains the Notestega methodology in detail. Section 4 demonstrates the Notestega implementation. Section 5 presents the steganalysis validation of Notestega, with a Conclusion in Section 6.

RELATED WORK

The aim of this section is to discuss contemporary linguistic and nonlinguistic steganography approaches, with a brief overview of automatic notetaking.

Linguistic Steganography

Linguistic steganography approaches conceal data in a linguistic-based textual cover. Linguistic steganography approaches can be categorized as follows.

- **Series of characters and words:** During World War I, the Germans communicated covertly using a series of characters and words known as null-cipher (Kahn, 1996). A null-cipher is a predetermined protocol of character and word sequence that is read according to a set of rules such as: read every seventh word or read every ninth character in a message. Apparently, suspicion is raised because the user is forced to fabricate a text-cover according to a predetermined protocol, which may introduce some peculiarity in the text that draws suspicion and defeats the steganographical goal. In addition, applying a brute force attack may reveal the entire message.
- **Statistical based:** Wayner has introduced the mimic functions approach (Wayner, 1992; Wayner, 2002), which employs the inverse of the Huffman Code by inputting a data stream of randomly distributed bits to produce text that obeys the statistical profile of a particular normal text. Therefore, the generated text by mimic functions is resilient against statistical attacks. Mimic functions can employ the concept of both Context Free Grammars (CFG) and van Wijnaarden grammars to enhance the output. The output of regular mimic functions is gibberish, rendering it extremely suspicious (Petitcolas, 1999; Bennett, 2004). However, the combination of mimic functions and CFG slightly improved the readability of the text (Wayner, 1992; Wayner, 2002). Yet the text-cover still contains numerous flaws such as incorrect syntax, lexicon, rhetoric, and grammar. In addition, the content of the text-cover is often meaningless and semantically incoherent. These shortcomings may raise suspicion in covert communications.
- **Synonym based:** Chapman and Davida have introduced a steganographic scheme consisting of two functions called NICETEXT and SCRAMBLE that uses a large dictionary (Chapman & Davida, 1997; Chapman & Davida, 2007; Chapman et al., 2001; Chapman & Davida, 2002). NICETEXT uses a piece of text to manipulate the process of embedding a message in a form of synonym substitutions. This process preserves the meaning of text-cover (the original piece of text) every time it is used. The synonyms-based approach attracted the attention of numerous researchers in the last decade: Winstein (1999, 2008), Bolshakov (2004), Bolshakov et al. (2004), Calvo et al. (2004), Chand et al. (2006), Nakagawa (2001), Niimi et al. (2003), Bergmair (2004), Bergmair et al. (2004, 2007), Topkara et al. (2006), Murphy et al. (2007), and Atallah et al. (2001, 2002). Although the text-cover of the synonym-based approach may look legitimate from a linguistics point of view given the adequate accuracy of the chosen synonyms, reusing the same piece of text to hide a message is a steganographical concern. If an adversary intercepts the communications and oversees the same piece of text that has the same meaning over and over again with just different group of synonyms between communicating parties, he will question such use.
- **Noise based:** Grothoff et al. have introduced the translation-based steganographic scheme (2005, 2006) to hide a message in the errors (noise) that are

naturally encountered in a Machine Translation (MT). This approach embeds a message by performing a substitution procedure on the translated text using translation variations of multiple MT systems. In addition, it inserts popular errors of MT systems and uses synonym substitutions to increase the bitrate. Unlike synonyms-based steganography, linguistic flaws in a noise-based approach are not a concern unless they appear excessively. However, Grothoff et al. state that one concern is that the continual improvement of machine translation may narrow the margin of hiding data. In addition, the translation-based approach, as pointed out by Grothoff et al., cannot be applied to all languages because the fundamental structures are radically different. This generates severely incoherent and unreadable text (Grothoff et al., 2005, 2006). On the contrary, Notestega can be applied to all known languages without any exceptions, while the generated note-cover is linguistically legitimate.

Another noise-based approach has been proposed by Topkara et al. (2007) that employs typos and ungrammatical abbreviations in a text, for example, emails, blogs, forums, etc., for hiding data. Moreover, Shirali-Shahreza et al. (2007) have introduced an abbreviation-based scheme to conceal data using the short message service (SMS) of mobile phones. Due to size constraints of SMS and the use of the phone keypad instead of the keyboard, a new language called SMS-Texting was defined to make the approach more practical. However, these approaches are sensitive to the amount of noise (errors) that occurs in a human writing. Such shortcoming not only increases the vulnerability of the approach but also narrows the margin of hiding data. Conversely, Notestega neither employs errors nor uses noisy text to conceal data.

- **Nostega based:** Recently, the new paradigm in steganography research, namely Noiseless Steganography Paradigm (Nostega) has been introduced (Desoky, in press), in which the message is hidden in the cover as data rather than noise. A number of methodologies have been developed based on the Nostega paradigm. One of these methodologies is the Summarization-based Steganography Methodology (Sumstega) (Desoky et al., 2008). Sumstega exploits automatic summarization techniques to camouflage data in the auto-generated summary-cover (text-cover) that looks like an ordinary and legitimate summary. Another linguistic steganographic scheme that is also based on Nostega paradigm is the List-based Steganography Methodology (Listega) (Desoky, in press). Listega exploits the popular textual list of itemized data to conceal messages in a form of textual list.

It is worth noting that the presented Notestega methodology in this paper follows this new paradigm by exploiting both human-notes and automatic notetaking techniques to camouflage data without generating any suspicious pattern.

Nonlinguistic Steganography

`Nonlinguistic steganography approaches can be categorized based on its file type such as text, image, audio, and graph. Textual steganography, which is based on nonlinguistic techniques, hides data by a Textual Format Manipulation (TFM) (Petitcolas, 1999) process. TFM modifies an original text by employing spaces, misspellings, fonts, font size, font style, colors, and noncolor (as invisible ink) to embed an encoded message. However, comparing the original text versus the modified text triggers suspicion and enables an adversary to detect where a message is hidden. In addition, TFM can be distorted and may be discerned by human eyes or detected by a computer (Petitcolas, 1999; Bennett, 2004).

On the other hand, image steganography is based on manipulating digital images to conceal a message. Such manipulation often renders the message as noise. In general, image steganography suffers from several issues such as the potential of distortion, the significant size limitation of the messages that can be embedded, and the increased vulnerability to detection through digital image processing techniques (Martin, Sapiro, & Seroussi, 2005). Audio-covers have also been pursued. Example of audio steganography techniques include LSB (Cvejic & Seppanen, April 2004, August 2004), spread spectrum coding (Bender et al., 1996; Kirovski & Malvar, 2001), phase coding (Bender et al., 1996; Ansari, Malik, & Khokhar, 2004), and echo hiding (Ansari, Malik, & Khokhar, 2004; Gruhl, Lu, & Bender, 1996). In general, these techniques are too complex and, like their image-based counterpart, are still subject to distortion and vulnerable to detection (Johnson & Satzenbeisser, 1999; Martin, Sapiro, & Seroussi,

2005; Petitcolas, 1999; Cvejic & Sepptanen, 2004). The hidden message may become to a great extent a foreign body in the cover and thus makes those schemes vulnerable to detection. In addition, contemporary steganography schemes rely on private or restricted access to the original unaltered cover in order to avoid the potential of comparison attacks, which is considered a major threat to the covert communication. Basically, an adversary can detect the presence of a hidden message by comparing a particular image-cover or audio-cover to the original image or audio file and finding out that some alterations have been made.

Hiding information in an unused or reserved space in computer systems (Anderson, Needham, & Shamir, 1998; ScramDisk, 2008). For example, the Windows 95 operating system has around 31 KB unused hidden space that can be used to hide data. Another example is the unused space in file headers of image, audio, etc. that can be used to hide data. This depends on the size of the hard drive used. TCP/IP packets used to transport information across the Internet have unused space in the packet headers (Handel & Sandford, 1996). The TCP packet header has six unused (reserved) bits and the IP packet header has two reserved bits. There are tremendous packets are transmitted over the Internet can convey and transmit a secret data. However, these techniques are vulnerable to distortion attacks (Johnson & Katzenbeisser, 1999; Kessler, 2004; Petitcolas, 1999).

Recently, a Graph Steganography (Graphstega) methodology has been developed (Desoky & Younis, 2006; Desoky & Younis, 2008). Unlike other schemes, the message is naturally embedded in the cover by simply generating the cover based on the message. Graphstega camouflages a message as data points in a graph and thus the message is not detectable as noise. The approach is shown to be resilient to a wide range of attacks, including a comparison attack by untraceable or authenticated data. Similarly, Chestega (Desoky & Younis, in press) exploits popular games such as chess, checkers, crosswords, and dominos for concealing messages in an unaltered authenticated data. Graphstega and Chestega are nonlinguistic methodologies that follow the Nostega paradigm (Desoky, in press) discussed above. As indicated earlier, Notestega is a linguistic steganography methodology that also follows the Nostega paradigm.

Automatic Notetaking

The field of automatic notetaking has enjoyed significant advances in recent years. It is currently more active than ever and is promising more in the future. Unlike machine translation and automatic summarization (Mani, 2001; Jones, 2007), automatic notetaking can embed nondirectly related elements to its output including linguistic elements (e.g., sentences, words, abbreviations) and nonlinguistic elements (e.g., lines, stars, arrows, symbols). As will be discussed, such a feature makes automatic notetaking a flexible scheme and provides an adequate room of concealing data. According to Fister and Girju (2008), automatic notetaking techniques are investigated from two prospective: linguistics and psychology. Also, linguistics-based investigation has been very minimal. A milestone of the linguistic analysis of notetaking is traced back to 1985 by Richard Janda. Janda states that "the purpose in taking notes is normally to have a potentially permanent record of at least the salient points of a lecture." This approach, from notetaking analysis point of view, treats a note as a register of human language (Fister & Girju, 2008; Janda, 1985). Janda has considered the notetaking register for an adult talk versus a baby, and native versus nonnative speakers of a language. He has observed that the note talk has "no expressive, upgrading, or even clarifying processes."

Furthermore, Janda made an experiment on an adequate collection of textual notes from a wide variety of lectures, topics, and students (Janda, 1985). He observed systematic of ten various types of grammatical reductions that occurred. It is also argued that notes retain linguistic and nonlinguistic contents. The nonlinguistic contents of notes may include the use of arrows, mathematical notations, lines, and so forth. In addition, the nonlinguistic contents of a lecture may be transformed into linguistic contents, such as the symbol of "=" can be written in sentence using the word "equal" and so on. It is worth noting that the aspect of note analysis has later been investigated and led to significant advances in automatic notetaking (Shuy, 1998). There are also other interesting automatic notetaking approaches from the prospective of psychology mentioned in (Fister & Girju, 2008) and worth investigating. Nonetheless, from a steganographical point of view, such note variations can be employed as steganographic carriers to conceal data.

NOTESTEGA METHODOLOGY

The main idea of the Notestega methodology is to exploit the variations among human-notes and the outputs of auto-notetaking techniques to conceal data. One of the unique features of automatic notetaking that distinguishes it from both machine translation and automatic summarization techniques is that its output can be augmented with nondirectly related elements using both linguistic, that is, sentences, words, abbreviations, etc., and nonlinguistic elements, for example, lines, stars, arrows, and symbols (Fister & Girju, 2008). Such a feature enables great flexibility in concealing data in a note-cover (text-cover) and provides adequate room for that. Basically, Notestega manipulates human-notes and the parameters of automatic notetaking to generate legitimate various notes of the same inputs and then substitutes some of the textual elements in one of the generated notes to embed a message without generating suspicious pattern. The selected note for this procedure is pre-agreed upon by configuring, in advance, the Notestega system among communicating parties. For example, if the Notestega system can produce four different notes for the same input(s), then the system should determine in advance which one will be used. In other words, is it the note number 1, 2, 3, or 4? Note that Nostega system generates different notes of the same input(s) that makes it feasible to designate. Moreover, the demand of using notes in business, science, education, news, etc., renders note attractive steganographic carriers and averts an adversary's suspicion when a note-cover (text-cover) is transmitted among communicating parties.

To illustrate how Notestega can be used, consider the following scenario. Bob and Alice are on a spy mission. Before they start their mission, which requires them to reside in two different countries, they set the rules for communicating covertly using their professions as a justification. To make this work, they establish a business relationship as follows. Bob and Alice are students in two different schools but for the same course and they agree to use Notestega. This is like an online forum when students from different schools all over the world they discuss a particular subjects and exchange classnotes. Bob and Alice generate notes of real topics to make their covert communications more legitimate. When Bob wants to send a covert message to Alice, Bob either posts notes online for authorized classmates or online student friends or he sends them via email. These notes conceal data. Covert messages transmitted in this manner will not look suspicious because Bob and Alice are students and their interaction is legitimate. The use of notes, both in academic and nonacademic spheres, is natural given the time pressure for writing such documents experienced by people nowadays render it innocent. Furthermore, Bob and Alice are not the sole recipients. There are other nonspy students who send and receive such notes, further warding off suspicion. However, only Bob and Alice will be able to unravel the hidden message because they know the rules of the game.

Notestega Architecture

Notestega camouflages both a message and its transmittal on a legitimate textual note. As stated earlier, in the above example of Bob and Alice, using a particular topic gives legitimacy for camouflaging both a message and its transmittal. The core idea of Notestega methodology is basically camouflaging data in notes. Obviously, such steganographic cover in a form of note linguistically, logically, and scientifically is legitimate. The following is an overview of the Notestega architecture, which consisted of four modules as shown in Figure 1:

1. **Topic(s) determination** (Module 1)**:** Determines an appropriate topic(s) for achieving the steganographic goal. One of the major factors for employing a particular topic(s) is the use of note. The chosen topic(s) can be an academic subject, for example, Psychology, History, or Digital Design, or a nonacademic subject such as Real Estate, Driver Jobs, or Construction, Trading can be employed by the Notestega methodology. Module 1 is only involved in the stage of constructing Notestega system.
2. **Message encoding** (Module 2)**:** Encodes a message in an appropriate and required form for the camouflaging process (Module 3). The process of generating a note-cover, by Module 3, may influence the process of how a message should be encoded. Therefore, studying and analyzing the output of Module 3 may be necessary for implementing an effective encoder. For example, a message may be encoded by slicing its binary string into a particular

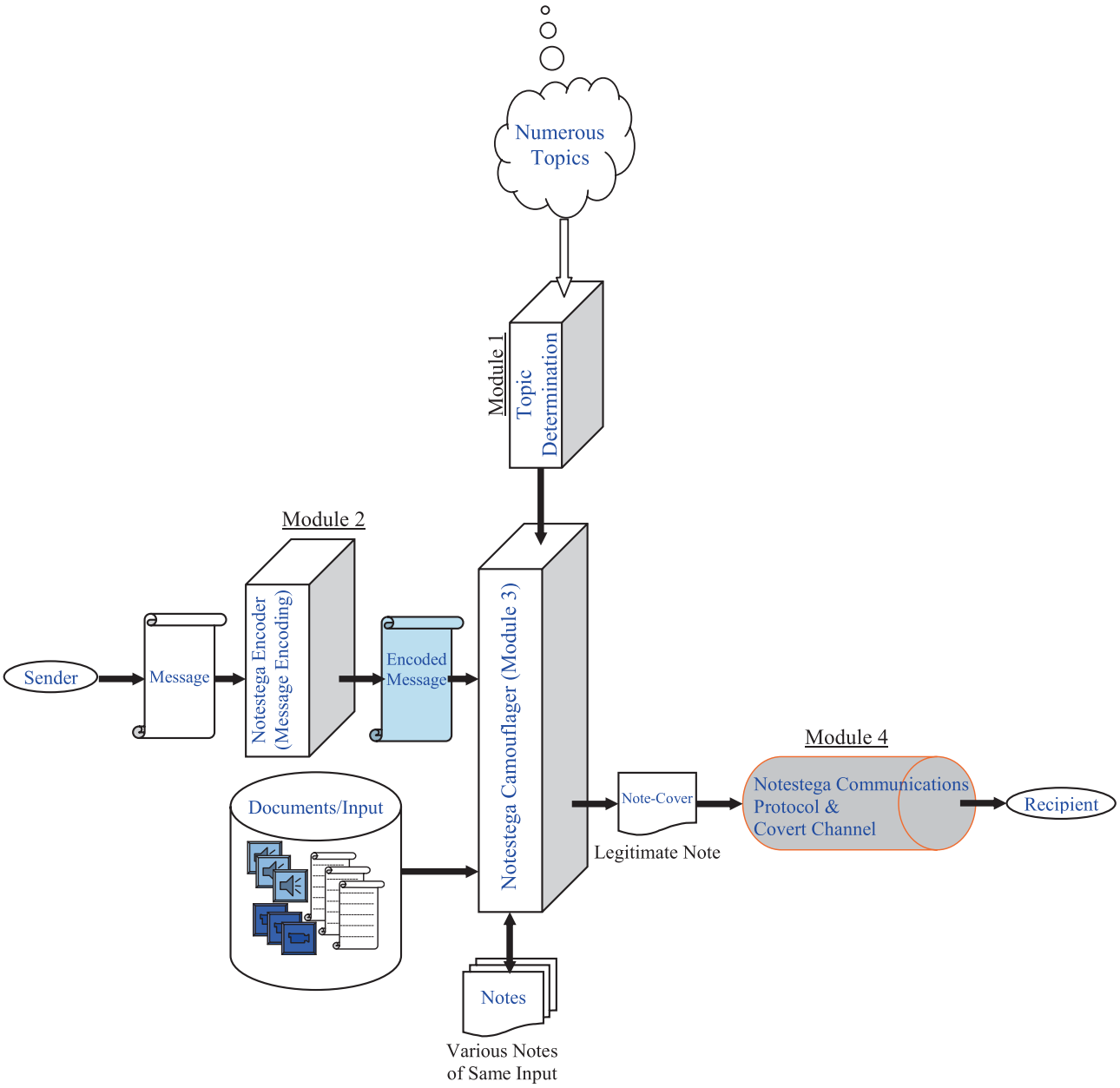

FIGURE 1 **Architecture of Notestega and the Communications Protocol.**

length of bits, for example, four or seven bits, as follows.

Message: "*Stop*"
Convert it to binary:
01010011011101000110111101110000
Then, slicing its binary string into four–bit groups:
0101 0011 0111 0100 0110 1111 0111 0000

3. **Message camouflager** (Module 3)**:** Generates the note-cover (text-cover) in which an encoded message, by Module 2, is embedded. Simply, Module 3 exploits human-notes and automatic notetalking techniques to embed the output of Module 2, the encoded message, in the generated textual note. This is accomplished in such a way that the note-cover looks legitimate like any ordinary note.
4. **Communications protocol** (Module 4)**:** Configures the basic protocol of how a sender and recipient would communicate covertly. Obviously, it includes the covert channel for delivering a note-cover to the intended recipient and the decoder scheme to unravel a hidden message.

Topic(s) Determination (Module 1)

The chosen topic(s) must be capable of concealing data. In other words, it must allow the process of embedding data without generating noise in order to achieve the steganographical goal. Since Notestega mainly manipulates textual notes and automatic notetaking techniques to camouflage messages, it can be applied to any topic that allows the use of notes. In addition, the chosen topic also has to fit the communicating parties and provide some ground for justifying the communications. For example, an uneducated person would not access, retain, exchange, or post atomic physics classnotes. Such communications can easily raise suspicion because such individual may use justifiable textual notes that match his background and his interest. Notestega naturally camouflages the delivery of a hidden message in a way that makes it appear legitimate and innocent. The scenario discussed in Section 3 demonstrates how the communications between Bob and Alice would not be unusual because their interest plays a role for camouflaging the delivery of note-cover. A legitimate reason for sending, receiving, accessing, or obtaining a particular material can legitimize the covert communications among communicating parties. Therefore, selecting the appropriate topic can play an essential role not only in camouflaging a message but also in transmitting a steganographic cover. In another words, selecting a justifiable topic is essential for establishing an appropriate covert channel for securing the steganographic communications.

Message Encoder (Module 2)

Notestega creates an encoded representation of a message and then camouflages it in a note-cover. The obvious constraint that Notestega imposes on the message encoder (Module 2) is to generate steganographic code that can be embedded in a note-cover. Given the availability of numerous encoding techniques in the literature (Johnson & Katzenbeisser, 1999) that can fit the presented methodology, the scope of the encoding process in this paper will focus on an example that illustrates how to meet the message encoding constraints. This example will be used in Section 4 in more details to demonstrate the applicability of Notestega. In the example, the encoding is done as follows. A message is first converted to a binary string. The string can be a binary of cipher text or a compressed representation. The binary string is then partitioned into groups of m bits. The value of m is determined based on the number "n" of different notes that are produced, as specified by the encoding parameters (Module 3 in Section 3.4). Basically, m is set to log n. If n= 4, that is, four different notes, the bit pattern 00, 01, 10, or 11 will be implied if an element in the note-cover uniquely matches that of the first, second, third or fourth generated note, respectively. Note that if the elements in the generated notes are not different from each other, then these elements imply null data bits. In other word, such elements will not be used to conceal data since Notestega employs only the variations among notes. Again, this encoding scheme is just for illustration and many alternate and more sophisticated schemes can be employed.

Message Camouflager (Module 3)

This module is responsible for generating a Notestega configuration on which the sender and receiver must preagree so that the hidden message can be extracted. There are numerous parameters to the auto-notetaking process that can be exploited to be steganographic carriers for concealing data. A parameter in this context means some input that a user may set to shape out the generated note. Examples of these parameters include the desired linguistics such as sentences, words, expression, abbreviations, etc., and the nonlinguistic elements such as lines, circles, symbols, and arrows. This is similar to two students who both have different notes for the same class. The generated note does not always look clear to everyone but it is clear to the one generated it or the one who is familiar with same topic. Unlike machine translation and automatic summarizer, automatic notetaking can embed nondirectly related elements to its output including linguistic elements (e.g., sentences, words, abbreviations) and nonlinguistic elements (e.g., lines, stars, arrows, symbols) (Fister & Girju, 2008) rendering the note-cover (text-cover) to preserve a plenty of room for concealing data. Simply, this module exploits mainly notes and the automatic notetaking techniques to generate set of notes for the same input. Each one of these notes

contains some uniquely different elements that will be used to embed a message. Then, based on a preagreed protocol Module 3 selects a particular note to serve as the original note. Finally, it employs these uniquely different elements to substitute the required elements from the original note (the selected one that untouched) to embed the encoded message from Module 2. As will be explained shortly, Notestega will use the various notes to camouflage the data in a note-cover.

Communications Protocol (Module 4)

The communicating parties configure the communications protocol of Notestega system, as shown in Figure 1, to communicate covertly by predetermining the following. The configuration includes the particular specifications of Notestega system used including its decoder and the covert channel for transmitting securely note-covers among communicating parties. Once communications protocol is agreed upon, the intended parties are ready to communicate covertly with each other using Notestega. First, the particular specifications of Notestega system used including its decoder. The first item is addressed by Modules 1, 2, and 3, which are discussed in the previous subsections. The second item is a particular covert channel that mainly defines how the cover will be delivered to the recipient without raising suspicion. Covert transmittal of the steganographic cover is very crucial to the success of steganography. At the core of the cover transmittal issue is how to prevent the association between the sender and recipient from drawing suspicion. For example, exchanging email messages would obviously imply a relationship between the communicating parties. Similarly, downloading files from a Website indicates an interest in the accessed material. With advances in monitoring tools for network and Internet traffic, profiles of user's access pattern can be easily established. An adversary most probably will suspect the presence of a hidden message, even if the content does not look suspicious, because of the observed traffic pattern and the lack of a justification for the interest in the contents of such traffic. For example, if a sender or recipient has pretended as a History undergraduate student and then sends or receives other suspicious documents such as classnotes of atomic physics, suspicious can easily be raised. Therefore, it is important to rationalize the sending and receiving of steganographic cover in order to avoid attracting any attention that may trigger an attack. Notestega enables an effective solution for the issue of legitimizing a cover transmittal. The use of a particular topic(s) allows establishing a covert channel in a form of legitimizing the association among communicating parties and thus sharing a note-cover would appear an ordinary practice. This is because the use of notes is very popular worldwide. Thus, the transmission of the note-covers via e-mail, posting them on Web pages, etc., is a natural matter that does not raise suspicion.

NOTESTEGA IMPLEMENTATION

Due to space constraints, only high-level approaches are used to illustrate the implementation of Notestega. As such, this section demonstrates the feasibility of Notestega methodology and its distinct capability of achieving the steganographical goal with higher bitrate than contemporary linguistic steganography approaches. It is worth noting that the focus of this section is on showing how Notestega achieves the steganographical goal rather than making it difficult for an adversary to decode an encoded message. Employing a hard encoding system or cryptosystem to increase the protection of a message is obviously recommended and straightforward using any contemporary encoder or cryptosystem. Similarly, employing compression to boost the bitrate can easily be accomplished by using one of the popular techniques in the literature. This section shows just few examples of possible implementations following the steps outlined in the previous section.

Notestega Configuration

This section first explains how Notestega modules are employed and configured to construct the overall Notestega system used by the communicating parties.

Determining Particular Topic(s) (Module 1)

In this paper, one topic of computer science field is employed, namely, the undergraduate class for Logic and Computer Design. Obviously, this topic is just an

example and any other topics may apply as stated in Section 3. Such topic is fairly popular among of computer science students and other professionals as well. Such topic demonstrates the capability of using not only linguistic elements but also nonlinguistic, for example, lines, stars, arrows, and symbols, and has no constrains for using textual notes render it suitable topic to for Notestega.

Notestega Encoder (Module 2)

Notestega encodes a message in a form that suits the camouflaging process. The steganographical code in this Notestega configuration works as shown in the Table 1.

Message Camouflage (Module 3)

Based on the output of Module 1, the note-cover is mainly a note of undergraduate class for Logic and Computer Design. Obviously, this topic is just used as an implementation example and so many other topics can be used. The camouflage module employs human-notes, automatic notetaking techniques, and uses popular Internet search engines such as google.com in order to accommodate the note-cover generation process. Modular 3 in this implementation generates four different notes and uses special characters from Microsoft Word 97, as shown in Table 2. The steganographic carriers are picked based on what matches the steganographic code value of an encode message (the bit string of a message). As will be shown in the examples below, the use of first two bits are used for the note style and 6 bits for special contents that are popularly used. This process does not impose any constrain on the employed implementation.

TABLE 1 **Details the Steganographic Code Used in this Paper. The First Two Bits are Employed to be Assigned for the Steganographic Carriers such as Styles, and Unique Different Elements of Various Notes**

Type of notetaking generation	1	2	3	4
Steganographic Binary Values For Style And Unique Different Elements	00	01	10	11
Special Embedding such as sentences, words, character, symbols, etc.	000000 – 111111			

TABLE 2 **Details the Steganographic Code Used in this Paper by Note Samples. The First Two Bits are Employed to be Assigned for the Steganographic Carriers such as Styles, and Unique Different Elements of Various Notes**

Type of notetaking generation	1	2	3	4
Steganographic Binary Values	00	01	10	11
Example Style	False and False = False	0 & 0=0	F & F=F	F and F=F
Example Special Embedding	♣, *, #, ➔,☺, etc. can conceal 6 bits from 000000 to 111111			

Communications Protocol (Module 4)

The chosen topic can play an essential role for legitimizing the discernable communications between sender and recipient such as the scenario of Bob and Alice in Section 3. In this example, a sender and recipient have a legitimate interest to the chosen topic which justifies the communicating parties to receive, send, obtains, etc., a textual notes that are related to such topic. Once the communications protocol is agreed upon, the intended parties are ready to communicate covertly with each other using Notestega. The following demonstrates examples of note-cover.

Samples of Note-Cover

The presented sample, is based on the chosen topic, demonstrates the robustness of Notestega. Table 3 demonstrates Sample 1 using Table 1 and 2. As observed, the note-cover below looks legitimate and an ordinary note. It is worth looking how other notes that do not contain a hidden message may look like as shown in Figure 2.

TABLE 3 **Details the Virtual Example of Notestega Methodology. In this Example, the Letter "X" Which is Binary String of its ASCII Representation is "01011000," will be Concealed**

Type of notetaking generation ➔	Notetaking generation type 2
Steganographic Binary Values	01
Example Style	0 & 0 = 0
Example Special Embedding	011000 = ➔

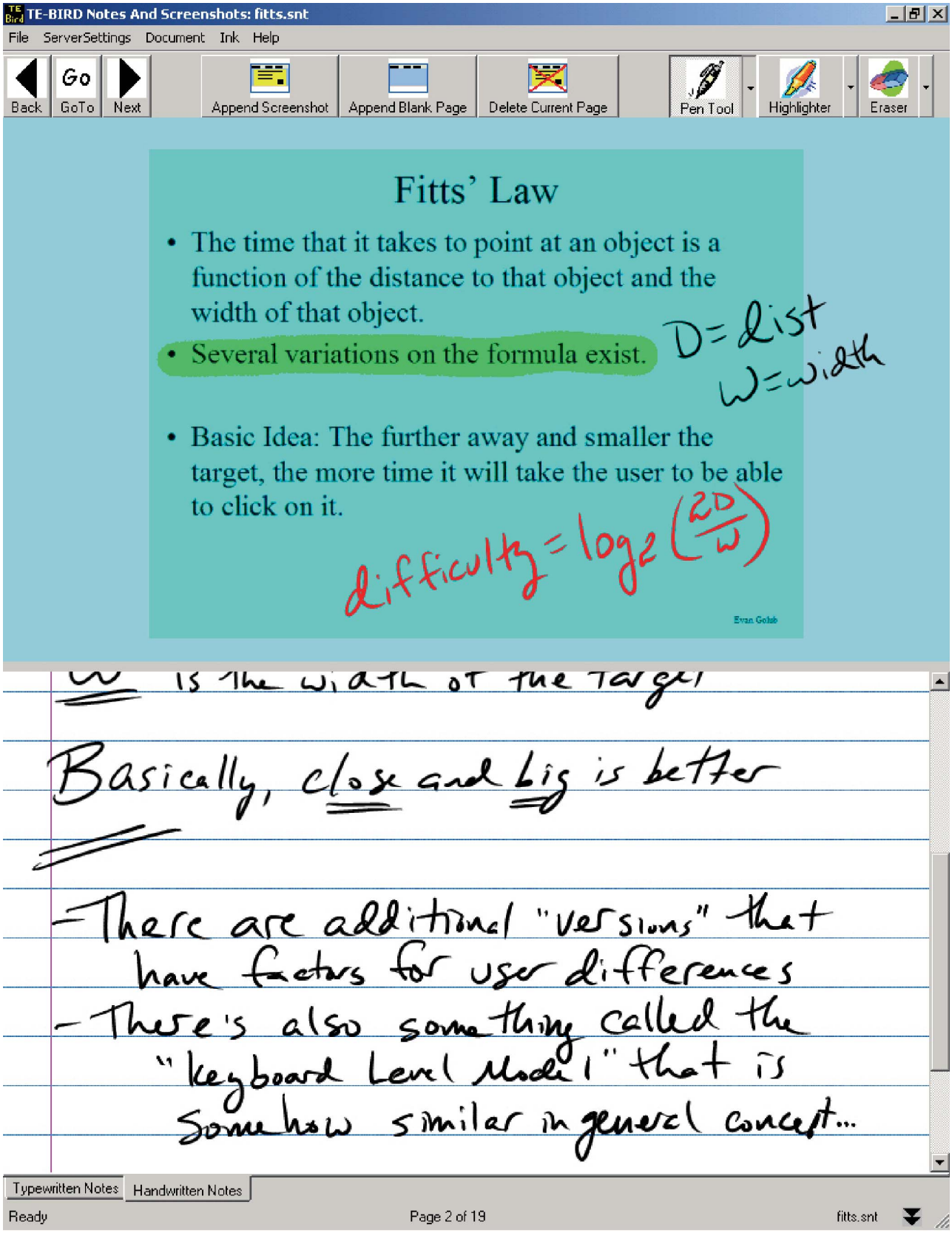

FIGURE 2 **Screenshots from (Notetaking System, 2008) that a Student Writes toward the Bottom of the Writing Surface. As Shown, More Room Is Automatically Created and Scrolled to.**

The following sample conceals the binary string "01011000" of the letter "X." Again, due to space constraints only high-level approaches are used to illustrate the implementation of Notestega. In, addition this can also be embedded among other class-notes of Logic and Computer Design.

Note-Cover Sample

Conceales 8 Bits :" → 0 & 0 = 0"

Bitrates

The aim of this section is to compare the bitrate of contemporary linguistic steganography approaches to that achieved by Notestega. The bitrate is defined as the size of the hidden message relative to the size of the cover. The average bitrate of the presented Notestega system used in this paper is roughly 7.777%. This can also be observed in the presented sample in this paper, which retains 9.090% bitrate. It is worth noting that the bitrate differs from one message to another, from one topic to another, and from one implementation to another as observed. To put this bitrate figure in perspective, the bitrate of contemporary linguistic steganography approaches has been investigated. The following reports on the findings, categorizing them based on the pursued approaches, while Table 4 provides a concise summary of these findings.

1. The statistical-based approach, namely mimic functions: An experiment has been conducted using 30 samples generated using Spam Mimic (Spam Mimic, 2007). An average bitrate of 0.90% is observed.
2. Synonym-based approaches:
   - For the NICETEXT scheme, the samples in (Chapman & Davida, 1997; Chapman & Davida, 2002) are used to estimate the bitrate, which is found to be approximately 0.29%.
   - The Winstein's scheme (Desoky, in press; Winstein, 2008) roughly hides about 6 bits per sentence, which yields a bitrate of approximately 0.5% based on the sentences listed in the these publications. However, this rate cannot be generalized since not every sentence in the text-cover conceals data. In addition, the size of sentences will affect the bitrate because there are short and long sentences. Nonetheless, the 0.5% figure is assumed given that it is based on the samples developed by the authors.
   - The capability of the scheme of Murphy et al. (Murphy & Vogel, 2007) again is reported as the number of bits per sentence. Based on the samples provided in their publication, the achievable bitrate is roughly 0.30% per sentence.
   - Nakagawa et al. (Fister & Girju, 2008) have provided two samples for their scheme. The samples achieve bitrate of 0.06% and 0.12%, respectively. However, it has been noted that when tried in a real application, only a bitrate of 0.034% could be reached.
3. Noise-based approaches:
   - The bitrate for the translation-based scheme reported in (Stutsman et al., 2006) is roughly 0.33%.
   - Based on the examples in (Topkara, Topkara, & Atallah, 2007), the confusing scheme approximately achieves a bitrate of 0.35%.
   - The linguistic techniques of the SMS-based methodology (Shirali-Shahreza et al., 2007) is said to be capable of hiding few bits in a file of several kilobytes, which yields an extremely low bitrate.

TABLE 4 The Bitrate of Contemporary Linguistic Steganography Approaches

Approach	Bitrate	Comment
Mimic functions (Wayner, 1992, 2002)	0.90%	Based on 30 samples generated at www.spamimc.com
NICETEXT (Chapman & Davida, 1997, 2002)	0.29%	Based on the samples in the cited papers
Winstein (1999; Desoky, in press)	0.5%	Based on the samples in the cited papers, and also confirmed in [41]
Murphy & Vogel (2007)	0.30%	Average per sentence (as reported in Murphy & Vogel, 2007)
Nakagawa et al. (2001)	0.12%	As reported in (Nakagawa et al., 2001), Bitrate achieved in real application is only 0.034%
Translation-based (Stutsman et al., 2006)	0.33%	Noted by the authors in the cited papers
Confusing (Topkara, Topkara, & Atallah, 2007)	0.35%	Based on the samples in the cited papers

Comparing the achieved bitrate by Notestega, which is roughly 7.777% versus the bitrate achieved by the contemporary linguistic approaches in Table 4, it is obvious that Notestega achieves much more superior bitrate than all comparable approaches, making it an effective steganography approach. The high bitrate also enables the use of reasonable cover sizes, a major concern for all steganography approaches linguistic and nonlinguistic.

STEGANALYSIS VALIDATION

The aim of this section is to show the resilience of Notestega to possible attacks. Again the success of steganography is qualified with its ability for avoiding an adversary's suspicion of the presence of a hidden message. It is assumed that an adversary will perform all possible investigations. In addition, the adversary is aware of Notestega, as a public methodology, but he does not know the Notestega configuration that the sender and recipient employ for their covert communication.

Traffic Attack

One of the possible attacks an adversary may pursue is to inspect the communications traffic of images, graphs, audio files, etc., in order to detect the existence of covert communications if occurred. For example, the intelligence community has a number of tools at their disposal for analyzing traffic on the Internet, tracking access to Websites, monitoring checked-out literature from public libraries, and so forth. The main goal of a traffic attack is to detect unusual or questionable association between a sender and recipient. Traffic analysis intuitively can identify who communicates with whom. The relationship between the communicating parties will be then qualified based on the contents of the message. Traffic attacks can be a threat for most contemporary steganographic techniques regardless of the steganographic cover types (e.g., image, graph, audio file, text) used. In the context of Notestega, the topic of the cover is checked rather than its validity and the consistency of its contents. If someone sends, receives, and accesses some materials without a legitimate reason for doing so, for example, a pretended uneducated person receives a atomic physics class notes from one of his friend, obviously suspicion can easily be raised and further investigation may be warranted. The additional investigations will involve a thorough analysis of a steganographic cover, as detailed in the following subsections.

Traffic analysis is deemed ineffective with Notestega. Notestega camouflages the transmittal of a hidden message (note-cover) to appear legitimate and thus suspicion is averted. Basically, Notestega ensures that the involved parties establish a covert channel by having a well-plotted relationship with each other rendering the communications traffic innocent and to look like any ordinary communications. Analyzing the traffic between them will not reveal any questionable association and will not trigger any further investigation because Notestega requires the communicating parties to use innocent topics that are highly discussed by a wide variety of people. Such topics create a high volume of traffic that makes it impractical for an adversary to investigate all traffics. The voluminous traffic allows the communicating parties to establish a covert channel in order to transmit a note-cover without drawing attention, rendering Notestega an attractive steganographical methodology. Finally, it is noted that if further investigation, on a note-cover, is triggered by traffic analysis, they would not be successful. In Notestega, differentiating between a note-cover that contains a hidden message and another peer textual note without a hidden message is not possible.

Contrast and Comparison Attacks

One of the intuitive sources of noise that may alert an adversary is the presence of contradictions in a note-cover. Examples of these contradictions include finding an excessive amount of repetition or wrong information. Also, if a note-cover contains errors, it is not expected to be severe and numerous. Such contradictions may raise suspicion about the existence of a hidden message, especially when they are present in the same document. Automating the generation of a note-cover through the use of automatic notetaking is feasible, due to recent advances in the filed, which makes the cover resilient to this type of attacks. As demonstrated in Section 4, the use of an Internet search engine, for example, Google, eases and supports the note-cover generation process because of the availability of the tremendous notes online. Meanwhile, noise in the context of comparison attacks reflects alteration of authenticated or previously used documents. The goal of the adversary is to find any incorrect and inconsistent data that may imply the manipulation of contents of a note-cover to embed a hidden message. However, since

reusing and modifying textual notes are common practices, comparison attacks are deemed ineffective.

Linguistics Attacks

Linguistics examination distinguishes the text that is under attack from normal human language. Distinguishing the text from normal human language can be done by examining meaning, syntax, lexicon, rhetoric, semantic, coherence, and any other feature that can help in detecting or suspecting the existence of a hidden message. These examinations are used to determine whether or not the text that is under attack is abnormal. Generally, the text used in textual notes is not sophisticated documents, and it is easy to retain the textual normality of note-cover. In addition, the produced textual notes meets the expected properties of a normal human language because it is initially generated by human and any alteration is done is also human-based which is thus does not generate any noise (linguistic flaws). As a result, the generated cover as demonstrated in the implementation section is normal text. Furthermore, if there are errors in the note generator engine, it should not be a concern for two reasons. First, it applies to all other textual notes that contain no hidden messages. Second, nothing is concealed in errors. In addition, an engine error of such note generator is most likely fixable. Therefore, Notestega is capable of passing any linguistic attack by both human and machine examinations.

On the other hand, a statistical attack refers to tracking the profile of the used text. A statistical signature (profile) of a text refers to the frequency of words and characters used. An adversary may use the statistical profile of a particular topic of documents that contains no hidden message and compare it to a statistical profile of the suspected note-cover to detect any differences. An alteration in the statistical signature of a particular document may be a possible way for an adversary to detect a noise. It has been shown that tracking statistical signatures is an ineffective means for attacking linguistic steganography (Grothoff et al., 2005; Stutsman et al., 2006). Nonetheless, Notestega is resistant to statistical attacks because it is simply opt to use legitimate text that is generated initially and naturally by both human and automatic notetaking. In addition, the generated note-cover keeps the same profile of its other peer documents that contains no hidden message. Basically, most alterations introduced by Notestega do not produce any flaws (noise), as demonstrated in the implementation section, deeming statistical attacks on note-cover ineffective.

CONCLUSION

The presented Notes-based Steganography Methodology (Notestega) conceals data in textual notes. The high demand for textual notes by a wide variety of people allows the communicating parties to establish a covert channel to transmit hidden messages (note-cover) rendering textual notes attractive steganographic carriers. Notestega neither hides data in a noise (errors) nor produces noise in the cover text. Instead, it camouflages data in legitimate notes by manipulating notes in order to embed data without generating any suspicious pattern. The presented implementation achieves bitrate up to 7.777%. Such bitrate is superior to contemporary linguistic steganography approaches found in the literature, confirming the effectiveness of Notestega methodology. Furthermore, Notestega can be applied to all languages. The steganalysis validation has demonstrated that Notestega methodology is capable of achieving the steganographic goal.

REFERENCES

Anderson, R. J., Needham, R., & Shamir, A. (1998). The steganographic file system. In *Proceedings of the Second International Workshop on Information Hiding: Vol. 1525. Lecture notes in computer science* (pp. 73–82). London: Springer-Verlag.

Ansari, R., Malik, H., & Khokhar, A. (2004, May). Data-hiding in audio using frequency-selective phase alteration. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, 5(17–21), 389–92.

Bender, W. et al. (1996). Techniques for data hiding. *IBM Systems Journal*, 35(3 & 4), 313–336.

Bennett, K. (2004). Linguistic steganography: Survey, analysis, and robustness concerns for hiding information in text. CERIAS Technical Report 2004-13, Purdue University.

Bergmair, R. (2004, September). Towards linguistic steganography: A systematic investigation of approaches, systems, and issues. Final year project, The University of Derby.

Bergmair, R., & Katzenbeisser, S. (2004, September). Towards human interactive proofs in the text-domain. *Proceedings of the 7th Information Security Conference (ISC'04). Lecture notes in computer science*. Berlin, Heidelberg: Springer-Verlag.

Bergmair, R., & Katzenbeisser, S. (2007, September). Content-aware steganography: About lazy prisoners and narrow-minded wardens. In *Proceedings of the 8th Information Hiding Workshop. Lecture notes in computer science*. Berlin: Springer Verlag.

Bolshakov, I. A. (2004, May). A method of linguistic steganography based on collocationally verified synonymy. In J. J. Fridrich (Ed.), *Information hiding: 6th International Workshop: Vol. 3200. Lecture notes in computer science* (pp. 180–191). Berlin: Springer.

Bolshakov, I. A., & Gelbukh, A. (2004, June). Synonymous paraphrasing using wordnet and Internet. In F. Meziane & E. Metais (Eds.), *Natural language processing and information systems: 9th international conference on applications of natural language to information systems, NLDB 2004: Vol. 3136. Lecture notes in computer science* (pp. 312–323). Berlin: Springer.

Calvo, H., & Bolshakov, I. A. (2004, October). Using selectional preferences for extending a synonymous paraphrasing method in steganography. In J. H. Sossa Azuela (Ed.), *Avances en Ciencias de la Computacion e Ingenieria de Computo - CIC'2004: XIII Congreso Internacional de Computacion* (pp. 231–242), .

Chand, V., & Orgun, C. O. (2006, January). Exploiting linguistic features in lexical steganography: Design and proof-of-concept implementation. *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS '06)*. *IEEE*, 6, 126b.

Chapman, M., & Davida, G. (1997). Hiding the hidden: A software system for concealing ciphertext as innocuous text. In *Proceedings of the International Conference on Information and Communications Security: Vol. 1334. Lecture notes in computer science* (pp. 335–345). Beijing, P. R. China. Berlin, Heidelberg: Springer-Verlag.

Chapman, M., & Davida, G. I. (n.d.). Nicetext system official home page. Retrieved August 3, 2007, from http://www.nicetext.com

Chapman, M. et al. (2001). A practical and effective approach to large-scale automated linguistic steganography. In *Proceedings of the Information Security Conference (ISC '01): Vol. 2200. Lecture notes in computer science* (pp. 156–165). Malaga, Spain, Berlin, Heidelberg: Springer-Verlag.

Chapman, M., & Davida, G. I. (2002). Plausible deniability using automated linguistic steganography. In G. Davida & Y. Frankel (Eds.), *International Conference on Infrastructure Security (InfraSec '02): Vol. 2437*. *Lecture notes in computer science* (pp. 276–287). Berlin: Springer.

Cvejic, N., & Seppanen, T. (2004). Increasing robustness of LSB audio steganography using a novel embedding method. *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)*, Las Vegas, Nevada, April 5–7, 2004.

Cvejic, N., & Seppanen, T. (2004). Reduced distortion bit-modification for LSB audio steganography '04. *Proceedings of the 7th International Conference on Signal Processing (ICSP 04)*, Beijing, China, August 31–September 4, 2004.

Davern, P., & Scott, M. (1995). Steganography its history and its application to computer-based data files. *Internal Report Working Paper: CA-0795*. School of Computing, Dublin City University. Retrieved August 3, 2006, from http://computing.dcu.ie/research/papers/1995/0795.pdf

Desoky, A., & Younis, M. (2006, November). PSM: Public steganography methodology. Technical Report TR-CS-06-07, Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, MD.

Desoky, A., & Younis, M. (2008). Graphstega: Graph steganography methodology. *Journal of Digital A novel Forensic Practice*, 2(1), 27–36.

Desoky, A. (2008). Nostega: Noiseless steganography paradigm. *Journal of Digital Forensic Practice,* 2(3),132–139.

Desoky, A. et al. (2008). Auto-summarization-based steganography. *Proceedings of the 5th IEEE International Conference on Innovations in Information Technology*, Al-Ain, UAE, December.

Desoky, A. (in press). Listega: List-based Steganography Methodology. *International Journal of Information Security*.

Desoky A., & Younis, M. (2009). Chestega: Chess steganography methodology. *Journal of Security and Communication Networks*.

Fister, A., & Girju, R. (2008, May). Preliminary investigation toward an automatic notetaking system. In *Proceedings of the 5th Midwest Computational Linguistics Colloquium (MCLC)*, Michigan

Grothoff, C., et al. (2005). Translation-based steganography. Technical Report CSD TR# 05-009. Purdue University (CERIAS Tech Report 2005-39).

Grothoff, C. et al. (2005, June). Translation-based steganography. *Proceedings of Information Hiding Workshop (IH 2005)* (pp. 213–233). Berlin: Springer-Verlag.

Gruhl, D., Lu, A., & Bender, W. (1996, May). Echo hiding. *Proceedings of First International Workshop on Information Hiding*. *Lecture notes in computer science* (pp. 295–316). Berlin, Heidelberg: Springer-Verlag.

Handel, T. G., & Sandford, M. T. (1996). Data hiding in the OSI network model. *Information Hiding: First International Workshop, Proceedings: Vol. 1174. Lecture notes in computer science* (pp. 23–38). Berlin, Heidelberg: Springer-Verlag.

Janda, R. D. (1985). Note-taking English as a simplified register. *Discourse Processes*, 8(4), 437–454.

Johnson, N. F., & Katzenbeisser, S. (1999). A survey of steganographic techniques. In S. Katzenbeisser & F. Petitcolas (Eds.), *Information hiding* (pp. 43–78). Norwood, MA: Artech House.

Jones, K. S. (2007). Automatic summarizing: The state of the art. *Info. Processing Mgmt.*, 43(6), 1449–1481.

Kahn, D. (1996). *The codebreakers: The story of secret writing* (rev. ed.). New York: Scribner.

Kessler, G. C. (2004). An overview of steganography for the computer forensics examiner. *Forensic Science Communications. Technical Report*, 6(3) http://www.garykessler.net/library/fsc_stego.htm.

Kipper, G. (2004). *Investigator's guide to steganography* (pp. 15–16). Boca Raton, FL: CRC Press LLC.

Kirovski, D., & Malvar, H. (2001). Spread-spectrum audio watermarking: Requirements, applications, and limitations. *Proceedings of the 4th IEEE Workshop on Multimedia Signal Processing*, 219–224, Cannes, France, October 3–5, 2001.

Mani, I. (2001). *Automatic summarization*. Amsterdam, Philadelphia: John Benjamins Publishing Company.

Martin, A., Sapiro, G., & Seroussi, G. (2005). Is image steganography natural? *IEEE Transactions on Image Processing*, 14(12), 2040–2050.

Mikhail, J. A., Raskin, V., Crogan, M., Hempelmann, C., Kerschbaum, F., Mohamed, D., & Naik, S. (2001, April). Natural language watermarking: Design, analysis, and a proof-of-concept implementation. In I. S. Moskowitz (Ed.), *Information Hiding: Fourth International Workshop. Lecture notes in computer science* (pp. 185–199). Berlin, Heidelberg: Springer-Verlag.

Mikhail, J. A., Raskin, V., Hempelmann, C. F., Topkara, M., Sion, R., Topkara, U., & Triezenberg, K. E. (2002, October). Natural language watermarking and tamperproofing. In A. P. Fabien & F. A. P. Petitcolas (Eds.), *Information Hiding: Fifth International Workshop: Vol 2578. Lecture notes in computer science* (pp. 196–212). Berlin, Heidelberg: Springer-Verlag.

Murphy, B., & Vogel, C. (2007, January). The syntax of concealment: Reliable methods for plain text information hiding. *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents*. http://spiedl.aip.org/getabs/servlet/GetabsServlet?prog-normal&id=PSISDG006505000001650 50Y000001&idtype-cvips&gifs-yes.

Nakagawa, H., Sampei, K., Matsumoto, T., Kawaguchi, S., Makino, K., & Murase, I. (2001). Text information hiding with preserved meaning - a case for Japanese documents. *IPSJ Transaction*, 42(9), 2339–2350 [originally published in Japanese]. A similar paper by the first author in English retrieved on June 4, 2008, from http://www.r.dl.itc.u-tokyo.ac.jp/ nakagawa/academic-res/finpri02.pdf

Notetaking System. (n.d.). Retrieved October 15, 2008, from http://www.cs.umd.edu/~egolub/AVIAN/TE-BIRD

Petitcolas, F.A.P., Anderson, R.J., & Kuhn, M.G. (1999, July). *Proceedings of the IEEE, special issue on protection of multimedia content*, 87(7), 1062–1068.

ScramDisk. (n.d.). ScramDisk: Free hard drive encryption for Windows 95 and 98. Retrieved August 3, 2008, from http://www.scramdisk.clara.net

Shirali-Shahreza, M. H., & Shirali-Shahreza, M. (2006). A new approach to Persian/Arabic text steganography. *Proceedings of 5th IEEE/ACIS International Conference on Computer and Information Science (ICIS-COMSAR 2006*, Honolulu, Hawaii, July 10-12.

Shirali-Shahreza, M. et al. (2007, November). Text steganography in SMS. *Convergence Information Technology,* November 21–23, 1714–1719.

Shuy, R. W. (1998). *What we do with English when we take notes: Evidence from a civil lawsuit. Studia Anglica posnaniensia: International review of English studies*. Poznán, Poland: Adam Mickiewicz University Press.

Spam Mimic. (n.d.). Retrieved July 31, 2007, from http://www.spammimic.com

Stutsman, R. et al. (2006). Lost in just the translation. *Proceedings of the 21st Annual ACM Symposium on Applied Computing (SAC'06)*, Dijon, France, April 2006.

Topkara, U., Topkara, M., & Atallah, M. J. (2006). The hiding virtues of ambiguity: Quantifiably resilient watermarking of natural language text through synonym substitutions. In *MM&Sec '06: Proceeding of the 8th Workshop on Multimedia and Security* (pp. 164–174), New York: ACM Press.

Topkara, M., Topkara, U., & Atallah, M. J. (2007, January). Information hiding through errors: A confusing approach. *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents*, San Jose, CA, January 29–February 1, 2007.

Wayner, P. (1992). Mimic functions. *Cryptologia*, XVI(3), 193–214.

Wayner, P. (2002). *Disappearing cryptography* (2nd ed., pp. 81–128). : Morgan Kaufmann.

Winstein, K. (1999, January). Lexical steganography through adaptive modulation of the word choice hash. Secondary education at the Illinois Mathematics and Science Academy. Retrieved April 15, 2008, from http://alumni.imsa.edu/~keithw/tlex/lsteg.ps

Winstein, K. (Lexical steganography. Retrieved August 3, 2008, from http://alumni.imsa.edu/~keithw/tlex