Information Security Journal: A Global Perspective

ISSN: 1939-3555 (Print) 1939-3547 (Online) Journal homepage: http://www.tandfonline.com/loi/uiss20

Innocipher: A Novel Innocent-Cipher-Based Cryptography Paradigm—High Level of Security for Fooling the Enemy

Abdelrahman Desoky

To cite this article: Abdelrahman Desoky (2013) Innocipher: A Novel Innocent-Cipher-Based Cryptography Paradigm—High Level of Security for Fooling the Enemy, Information Security Journal: A Global Perspective, 22:2, 83-97, DOI: 10.1080/19393555.2013.774448

To link to this article: https://doi.org/10.1080/19393555.2013.774448

Information Security Journal: A Global Perspective, 22:83–97, 2013

ISSN: 1939-3555 print / 1939-3547 online
DOI: 10.1080/19393555.2013.774448

Taylor & Francis
Taylor & Francis Group

Innocipher: A Novel Innocent-Cipher-Based Cryptography Paradigm—High Level of Security for Fooling the Enemy

Abdelrahman Desoky
The Academia Planet,
Baltimore, MD, USA

ABSTRACT The recent advances in cryptanalysis techniques are the major threat to cryptography. A leakage of information about the cryptosystem used by either a fatal shortcoming or an insider enemy can easily defeat the cryptographic goal. An adversary may succeed in decrypting ciphertexts, while users of a particular cryptosystem unwarily continue using same vulnerable encryption techniques. Such major concerns motivate the development of a novel Innocent-Cipher-Based Cryptography Paradigm (Innocipher), which is presented in this paper. Innocipher focuses on high level security that protects private information through two phases. First, Innocipher system conceals the required data in a legible form of legitimate plaintext other than ciphertext, for example, legitimate text, graph, game, and image, that looks benign and legitimate. Second, it converts the output of the first phase, the encoded data in the legible form, into a ciphertext. The main advantage of the Innocipher paradigm is that if a worst case scenario occurred, which is an adversary succeeding in decrypting a cipher message, then an adversary will be fooled by getting a legible form of legitimate text. At this point, the adversary will stop any further investigation while an original message is not revealed. This fooling mechanism of Innocipher is the key-feature that enables a multilayer of security for protecting valuable information. The presented implementation and validation of Innocipher demonstrates the robust capabilities of achieving the goal of securing information in static stage and during data transmission to its legitimate recipient.

KEYWORDS secure communications, cryptography, information security, computer security, network security

1. INTRODUCTION

This paper promotes the novel Innocent-Cipher-Based Cryptography Paradigm (Innocipher). Innocipher allows communicating parties to secure their data in static stage and during data transmission over an insecure communications channel to a legitimate recipient. Innocipher achieves its goal through two modules. First, the Innocipher system encodes data into a legible form of legitimate plaintext (e.g., text, graph, game, image) that looks innocent. Second, it encrypts the output of

Address correspondence to
Abdelrahman Desoky.
E-mail: desoky@desoky.com

first phase, the encoded data in the legible form, into a ciphertext. The Innocipher paradigm takes advantage of the recent advances in steganography techniques to construct the first phase. Literally, steganography is the science and art of camouflaging the presence of covert communications (Desoky, 2012, 2010, 2009). The origin of steganography is traced back to early civilizations (Desoky, 2012, 2010, 2009). The ancient Egyptians communicated covertly using the Hieroglyphic language, a series of symbols representing a message (Desoky, 2012, 2010). The message looks as if it is a drawing of a picture, although it may contain a hidden message that only the intended recipient may obtain. The Greeks also used steganography, "hidden writing," where the name was derived. Fundamentally, the steganographic goal is not to hinder the adversary from decoding a hidden message but rather to prevent an adversary from suspecting the existence of covert communications. When using any steganographic technique, if suspicion is triggered, the goal of steganography is defeated, regardless of whether or not an original message in its plaintext form is revealed (Desoky, 2012, 2010, 2009). Contemporary approaches are often classified based on the steganographic cover type into graph, game, text, image, audio, and so forth. These steganographic covers are the objects that camouflage data in an innocent form.

Unlike steganography, cryptography by its nature advertises the fact that the outputs of its cryptosystems, namely ciphertexts, carry secret information. While steganography acts by its nature that the outputs of its stegasystems, namely steganographic cover (e.g., text, graph, game, image), are innocent material and have nothing to conceal (Desoky, 2012). Fundamentally, cryptography techniques depend on the claimed-strong techniques that challenge adversaries for decrypting a cipher message. Adversaries employ cryptanalysis techniques to investigate and study a cryptography method for obtaining the meaning of encrypted message without accessing an actual secret cryptosystem or its actual cryptokey (Koblitz, 1994; Stallings & Brown, 2008). Typically, this involves knowing how the system functions and finding out or calculating a secret key (Koblitz, 1994; Stallings & Brown, 2008). Generally, this is the practice of cracking the cryptosystem or ciphertext. For an adversary to encode a cryptosystem, a cryptanalysis must be performed on the unreadable text (ciphertext). Therefore, the cryptosystem used must be identified along with its specification and parameters used Koblitz, 1994). In steganography, it is strongly recommended and it is a common practice to encrypt the message before camouflaging it (Desoky, 2012, 2010). In other words, stegasystems conceals the ciphertext, not only to give the message more protection but also to pass as an innocent communication.

Conversely, Innocipher opts to encrypt a message after it is concealed in an innocent form, for example, plaintext. In other words, Innocipher encrypts a steganographic cover. This process will be done even if a plaintext was encrypted before it is concealed in a plaintext form. This process is done to overcome the following threats:

1. A cryptanalyst may succeed to decrypt ciphertext due to the recent advances in cryptanalysis techniques or due to similar reasons.
2. A leakage of information about the cryptosystem conceived by either fatal mistakes or an insider enemy can easily defeat the cryptographic goal.

These major concerns motivate the development of a novel Innocent-Cipher-Based Cryptography Paradigm (Innocipher), presented in this paper, to overcome such concerns. The main advantages are as follows:

A. Enables a multilayer of security, which elevates the security level.
B. If an adversary succeeds in decrypting a cipher message the actual message will not be revealed.
C. Fools the enemy.

The advantages of the presented paradigm arises if a worst case scenario occurred, which is if an adversary succeeded in decrypting a cipher message, then he will claim victory. At this point, the adversary will stop any further investigation while the message is not divulged.

The remainder of this paper is organized as follows. Section 2 discusses the related work. Section 3 explains the Innocipher paradigm in details and briefly demonstrates an example of an Innocipher system implementation. Section 4 discuses the possible attacks. Finally, Section 5 concludes the paper.

2. RELATED WORK

The aim of this section is to present an overview of contemporary cryptography and steganography approaches. Section 2.1 briefly describes the contemporary cryptography techniques. Then, contemporary steganography techniques are discussed in this paper according to the

following category. Section 2.2 describes the linguistic steganography, and section 2.3 highlights the non-linguistic steganography.

2.1. Cryptography

The output of all encryption systems is a ciphertext, which is the disguised form so that only the intended recipients can unravel the original message (e.g., plaintext, plain-image, plain-graph). Yet all contemporary cryptography systems, when decrypting the reverse mode of encryption process, a ciphertext will get a plaintext. Unlike all contemporary cryptography techniques, when an Innocipher system decrypts a ciphertext the output will be in an innocent legitimate form (e.g., text, graph, game), and the original message will still be protected and unrevealed. This is the major difference between all contemporary cryptosystems and the Innocipher system. Nonetheless, the following briefly describes contemporary cryptography approaches.

Cryptography is the science and art of converting plaintext, for example, legible-text, legible-graph, legible-game, and legible-image, into an illegible form of text that called ciphertext. Cryptography is traced back to circa 1900 B.C. when Egyptian scribes used nonstandard Hieroglyphic in an inscription (Desoky, 2012, 2010, 2009; Ibahim, 1992). The Greeks also used cryptography, where the name derives. However, during the Muslim Empire, Muslim Arabians "صفر" also invented numerous cryptosystems and contributed significantly to the field of cryptography. The cipher word is driven from the Arabic word and pronounced "Sifer," which means zero or null (Ibrahim, 1992).In World War II, the German Lorenz used a cipher machine, to encrypt secret messages (Kahn, 1996). Before the computer age, classic cryptography was widely used. A famous example of the classic cryptography is the technique that was introduced by Gaius Julius Caesar who was a Roman military and political leader during the Roman Empire era (Luciano & Prichett, 1987).He used a substitution-based cipher which was named after his last, Caesar cipher, in which each letter of plaintext is substituted by a letter some fixed number of positions further down the alphabet. Julius Caesar communicated with his generals during his military campaigns using this scheme by shifting of 3, which it is just like EXCESS-3 code in Boolean algebra (Luciano & Prichett, 1987).

Modern cryptography intersects with other disciplines, for example, mathematics, computer science, and engineering, which arose during the computer age. Symmetrical cipher refers to particular encryption techniques in which a communicating party, (the sender and recipient), either shares the same key or uses different keys that are relatively easy to be retrieved computably by only intended recipient as postulated (Koblitz, 1994; Stallings & Brown, 2008). In modern cryptography, the examples of symmetric-key ciphers are mainly, but not limited to, the block ciphers, stream ciphers, and to their related-based schemes.

The block cipher was invented by Leon Battista Alberti which takes an input of plaintext along with a key to generate a block of ciphertext that is in the same size of its input. Block cipher techniques divides an original message, plaintext, to a particular number of blocks and then it encrypts each block a number of times using a key. This procedure may differ from one implementation to another and from one particular algorithm to another. For instance, DES encrypts each block 16 rounds. Note that if a message is encrypted using block cipher techniques, it is most likely longer than a single block, so it requires a way to assemble the output block ciphers in its correct order. The Data Encryption Standard (DES) and the Advanced Encryption Standard (AES) are primarily block-cipher-based schemes that comply to the cryptography standards of the U.S. government. In spite of withdrawing DES after adopting AES as an official standard, DES and its variation, the triple-DES, remains popular enough and it is employed to secure data in numerous applications, for example, ATM, e-mail, and remote access (FIPS, 2007; Merkle, 1989; RFC, n.d.; NCUA, 2004). Also, as there are good quality versions of block ciphers, there are also vulnerable block cipher versions (Menezes, van Oorschot, & Vanstone, 1996; Desoky, 2003; Schneier, 1996). On the other hand, stream-cipher-based cryptosystems generate an arbitrary long key-stream (pseudorandom cipher bit stream) that is blended with a plaintext digit by digit somehow as a one-time pad using mainly XOR operation, for example, the widely used RC4 (Menezes et al., 1996). Block ciphers can be utilized as stream ciphers similarly in block cipher modes of operation (NIST, n.d.).

In other techniques such as hash-functions-based cryptosystems, a message of any length is converted to a short output, which makes it suitable for a digital signature applications (FIPS, 2007; Merkle, 1989; RFC, n.d.). There are hash-functions-based cryptographic schemes vulnerable, and there are others quite secure (Koblitz, 1994; Stallings & Brown, 2008). Examples of vulnerable hash-functions-based cryptosystems are MD2,

MD4, and MD5 (Dobbertin, "Cryptanalysis of MD4," 1996; Rivest, "MD4," 1992; Rivest, "MD5," 1992; den Boer & Bosselaers, 1993; Dobbertin, "Cryptanalysis of MD5," 1996; Rogier & Chauvaud, 1997; Muller, 2004; Knudsen & Mathiassen, 2005). Therefore, a hash function design competition is coming up by 2012, namely SHA-3, in order to pick up a new U.S. national standard. Note that the techniques of Message Authentication Code (MAC) are similar to the hash-functions-based cryptography except a key is used to authenticate the hash value of receipt (Menezes et al., 1996; Desoky, 2003).

One of the concerns of using symmetric-cipher-based cryptosystems is that the intended users require a secure and sophisticated key management in particular if there is a big number of users (Koblitz, 1994; Stallings & Brown, 2008). This may not only involve that communicating parties opt to share or swap different keys but also may include bartering cipher messages (Koblitz; Stallings & Brown).

Unlike symmetric-key cryptosystems, the public-key-based cryptography is capable of employing public keys to be publicly distributed. Note that each intended user hold a secret key which remains unused by others except only its owner. In detail, a sender will use the public key to encrypt a message, while a recipient uses its own secret-key to decrypt a ciphertext. In 1976, Whitfield Diffie and Martin Hellman introduced a public-key cryptosystem that was based on a discrete logarithm problem, which was named the asymmetric key system (Diffie & Hellman, June 1976, November 1976). The strength of Diffie-Hellman system is based on the claim is that the discrete logarithm problem is hard to be resolved by an adversary due to an absent information, and not only the secrete key. The system employs two different keys as follows. The first key is publicly used while the second key is secretly used (Diffie & Hellman, June 1976). Shortly after two years, precisely in 1978, Ronald Rivest, Adi Shamir, and Len Adleman presented RSA, which is also a public-key-based cryptosystem (Rivest, Shamir, & Adleman, 1978). Mathematically, the RSA's strength is fundamentally inherited from the assumption of the impracticality of factoring large prime numbers. A distinct public-key cryptosystem that is based on a discrete logarithm problem is the ElGamal scheme (El Gamal, "On computing," 1985; "Public key cryptosystem," 1985, 1984, "Public key cryptosystem and signature scheme," 1983, "Subexponential," 1983). The ElGamal system is capable of encrypting messages and verifying signatures. An up-to-date ElGamal cryptosystem does not leak information to an adversary to be used in future attacks Koblitz, 1994; Stallings & Brown, 2008; (El Gamal, "On computing," 1985; "Public key cryptosystem," 1985, 1984, "Public key cryptosystem and signature scheme," 1983, "Subexponential," 1983). In the mid-1980s, another public-key cryptosystem was developed based on numerical theory problems of elliptic curves (Washington, 2003).

The "cat and mouse problem" is always persistent in the area of security. More precisely, the race between cryptographers and attackers has never been an easy ordeal. This is due not only to the continual advances in cryptanalysis techniques but also to the high leakage of information about the cryptosystem conceived of by either a silly mistake or an insider enemy. The leakage of information about the cryptosystem used, for example, losing or giving secret key of a cryptosystem used to the enemy, can easily defeat the cryptographic goal. The conclusion is that an adversary may succeed in decrypting ciphertexts while cryptographers unwarily continue using same vulnerable encryption techniques. All contemporary cryptosystems' ciphers are reversible to their original messages by only legitimate users, as presumed. This means that in case an adversary succeeded by somehow to reverse a cipher to a plaintext, the cryptosystem used fail in protecting data.

These major concerns motivate the development of a novel Innocipher presented in this paper. Unlike all contemporary cryptosystems, the presented Innocipher system is capable of unraveling the original message even if an adversary successfully reversed a ciphertext, of an Innocipher system, to its plaintext. Simply, the first output of decrypting the ciphertext of Innocipher system is a plaintext that also needs to be "decrypted" (reversed) for an original data to be revealed. Innocipher schemes achieve the goal of securing data through two phases or stages, as follows. First, Innocipher system conceals the required data in a legible form, for example, plaintext, plain-graph, plain-game, and plain-image, that looks innocent and legitimate. Second, it encrypts the output of first phase, the concealed data in the legible form, into a ciphertext. The main advantage of Innocipher paradigm is that if a worst case scenario occurred in which an adversary succeeded in decrypting a cipher message to a plaintext, the message will remain unrevealed. At this point, the adversary will stop any further investigation. This fooling mechanism of Innocipher paradigm is the key feature that enables multilayer of security for protecting valuable information.

2.2. 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 a 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, 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 functions are 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, thus rendering it extremely suspicious (Desoky, 2012, 2010, 2009). However, the combination of mimic functions and CFG slightly improved the comprehension of the text (Wayner, 1992, 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 use a large dictionary (Chapman & Davida, "Hiding the hidden," 1997; "Nicetext system," n.d.; 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, n.d.); Bolshakov et al. ("Method," 2004, "Synonymous," 2004); Calvo and Bolshakov (2004); Chand and Orgun (2006); Nakagawa et al. (2001); Niimi et al. (2003); Topkara, Topkara, and Atallah (2007, 2006); and Murphy and Vogel (2007). Although the text-cover of 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 a use.

Noise based: Desoky (2012) has introduced the translation-based steganographic scheme 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 also uses synonym substitutions to increase the bitrate. Unlike synonyms-based steganography, linguistic flaws in noise-based approach are not a concern unless they appear excessively. However, Desoky states that one of the concerns 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 Desoky, cannot be applied to all languages because of the fundamental structures are radically different. This generates severely incoherent and unreadable text (Desoky, 2012). On the contrary, Listega can be applied to all known languages without any exceptions while the generated list-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, and forums, for hiding data. Moreover, Desoky (2012) has 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 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 shortcomings not only increase the vulnerability of the approach but also narrow the margin of hiding data. Conversely, Listega neither employs errors nor uses noisy text to conceal data.

Nostega-based: Recently, a new paradigm in steganography research, namely Noiseless Steganography Paradigm (Nostega) (Desoky, 2012, 2010, 2009, 2008)

is introduced. Nostega conceals messages in a cover as legitimate data rather than noise. A number of linguistic methodologies have been developed based on the Nostega paradigm. These methodologies are as follows. Summarization-based Steganography Methodology (Sumstega) (Desoky, 2012, 2010, 2009, 2008) exploits automatic summarization techniques to camouflage data in the auto-generated summary-cover (text-cover) that looks like an ordinary and legitimate summary. List-based Steganography Methodology (Listega) (Desoky, 2012, 2010, "Nostega," 2009, "Listega," 2009) manipulates itemized data to conceal messages in a form of textual list. Notes-Based Steganography Methodology (Notestega) takes advantage of the recent advances in automatic note-taking techniques to generate a text-cover (Desoky, 2012, 2010, "Nostega," 2009, "Notestega, 2009). Notestega embeds data in the natural variations among both human-notes and the outputs of automatic-notetaking techniques. Mature Linguistic Steganography Methodology (Matlist) exploits NLG and template techniques along with Random Series values (RS) to camouflage data without generating any suspicious pattern. Matlist employs a particular domain-specific subject such as finance, medicine, science, economics, and so forth (Desoky, 2012, 2010, "Nostega," 2009, "Jokestega," in press, 2008). The qualified domain-specific subject is based on a random series of binary, decimal, hexadecimal, octal, alphabetic, alphanumeric, or any other form. Unlike Matlist, the Normal Linguistic Steganography Methodology (NORMALS) neither generates noise nor uses noisy text to camouflage data (Desoky, 2012, "Comprehensive, 2010, "Nostega," 2009, "Normals," 2010). NORMALS employs Natural Language Generation (NLG) techniques to generate noiseless (flawless) and legitimate text-covers by manipulating the inputs' parameters of NLG system in order to camouflage data in the generated text. As a result, NORMALS is capable of fooling both human and machine examinations. NORMALS is capable of handling nonrandom series domains. Recently, another methodology called Automatic Joke Generation Based Steganography Methodology (Jokestega) is published that is also based on Nostega (Desoky, "Jokestega," in press. Basically, Jokestega pursues textual jokes in order to hide messages. Fundamentally, the Jokestega methodology takes advantage of recent advances in Automatic Jokes Generation (AJG) techniques to automate the generation of textual steganographic cover by concealing the required data using jokes variations.

It is worth noting that the presented Innocipher paradigm in this paper is based on this new Nostega paradigm to conceal data in a noiseless plaintext then encrypting that plaintext that contains hidden data. Obviously, this paradigm will protect valuable data even if an adversary succeeded in decrypting the ciphertext.

2.3. 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 Textual Format Manipulation (TFM) process (Desoky, 2012, "Comprehensive," 2010). 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 (Desoky, 2012, "Comprehensive," 2010).

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 distortion, the significant size limitation of the messages that can be embedded, and the increased vulnerability to detection through digital image processing techniques (Kahn, 1996). Audio-covers have also been pursued. Example of audio steganography techniques include LSB (Cvejic & Seppanen, "Reduced distortion," 2004, "Increasing robustness, 2004), spread spectrum coding (Desoky, 2012; Kirovski & Malvar, 2001), phase coding (Desoky, 2012; Ansari, Malik, & Khokhar, 2004, and echo hiding (Ansari et al., 2004; Gruhl, Lu, & Desoky, 2012). In general, these techniques are too complex, and like their image-based counterparts, are still subject to distortion and are vulnerable to detection (Desoky, 2012, "Nostega," 2009; Cvejic & Seppanen, "Reduced distortion," 2004; Martin, Sapiro, & Seroussi, 2005). 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, n.d.). For example, the Windows 95 operating system has around 31 KB unused hidden space which can be used to hide data. As another example, unused space in file headers of image, audio, and so forth can also be used to hide data. This depends on the size of the hardrive being used. TCP/IP packets used to transport information across the Internet have unused space in the packet headers (Handell & Sandford, 1996). The TCP packet header has six unused (reserved) bits and the IP packet header has two reserved bits. The tremendous packets transmitted over the Internet can convey and transmit a secret data. However, again, these techniques are vulnerable to distortion attacks (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009).

Recently, a Graph Steganography (Graphstega) methodology has been developed (Desoky, "Graphstega," 2008, "Public steganography," 2006). Unlike all 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, for example, numerical values that can be plotted in a chart, and thus the message would not be 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 (Destoky & Younis, in press) exploits popular games such as chess, checkers, crosswords, and dominos for concealing messages in an unaltered authenticated data. Graphstega and Chestega represent a new paradigm in steganography research in which the message is hidden in the cover as data rather than noise. It is worth noting that Graphstega and Chestega follow this new paradigm, namely Nostega (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009, "Nostega: A novel," 2008).

3. INNOCIPHER PARADIGM

To illustrate Innocipher, consider the following scenario. Bob and Alice are on a spy mission, which involves a high level of security, in a country that allows transmitting ciphertexts over the Internet. The mission requires Bob and Alice to pretend that they are running an online-business, which requires them to encrypt their messages to protect themselves against credit card frauds and other online-business competitors. Before they went on their mission, which requires them to reside in two different countries, they plot a secure plan and set the rules for communicating covertly utilizing their online-business as a secure umbrella. They basically agree to secure data through two phases as follows. First, concealing the required data in a legible form other than ciphertext, for example, plaintext, plain-graph, plain-game, or plain-image, that looks innocent and legitimate. Second, it encrypts the output of the first phase, the encoded data in the legible form, into a ciphertext. Unfortunately, the adversary is capable of breaking the cryptosystem used by Bob and Alice. However, because Bob and Alice considered the worst case scenario which if an adversary succeeded in decrypting a cipher message then will get a legible form, for example, plaintext, other than the intended original message. Obviously, this decrypted ciphertext is in a form of legitimate plaintext that needs more decryption, reversing procedure, to reveal an intended original message. This makes an adversary to claim victory and to stop any further investigation while a secrete-message is "unrevealed." This fooling mechanism of Innocipher is the key feature that enables multilayer of high level security for protecting valuable information. The next section shows the overview of Innocipher architecture.

3.1. Innocipher Architecture

The scenario of Bob and Alice in section 3 demonstrates how Innocipher paradigm can be used. As mentioned earlier, the core idea of Innocipher is that it achieves its the security goal by camouflaging the required message in legible form then it encrypts the camouflaged message. The following is an overview of the Innocipher architecture, which consisted of two phases as shown in Figure 1:

1. Concealing data on a legible form (Phase 1): This phase is constructed based on the Noiseless Steganography Paradigm (Nostega), which is capable of concealing data in a noiseless legible form, for example, plaintext, plain-graph, plain-game, and so forth. The major reason of employing the Nostega-based system is its superior capabilities of convincing an adversary that the plaintexts are innocent and legitimate, which implies no hidden message is concealed.

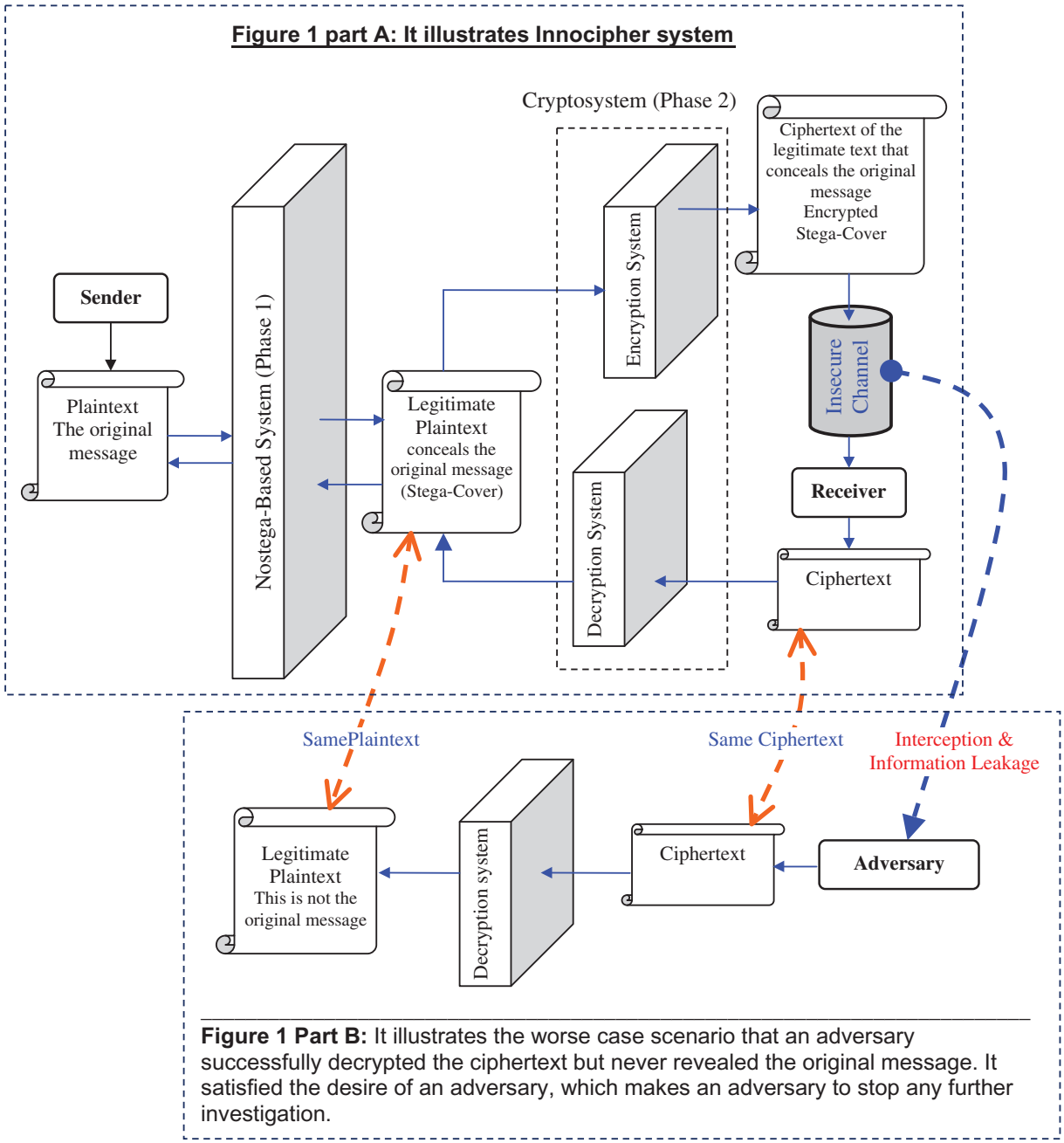


FIGURE 1 **Illustrates the architecture of Innocipher scheme and its rustiness in fooling the enemy (color figure available online).**

2. Converting the output of Phase 1 into ciphertext (Phase 2): The intended users determine an appropriate cryptography scheme to achieve their goal in order to convert the output of Phase 1 into ciphertext. Since the goal of intended users may differ from one group to another, selecting or constructing such scheme or using a contemporary one is flexible based on the user goal. For instance, level of security and which attacks are countered.

The above phases are discussed in the following subsections.

3.2. Concealing Data on a Legible Form (Phase 1)

This section discuses that the novel Noiseless Steganography Paradigm (Nostega) and its capabilities of playing a distinct role in securing the cryptographic communications. Nostega neither hides data in noise nor produces noise. Instead, it camouflages messages in a form of unquestionable data in the generated cover. Conversely, steganography approaches found in the literature have focused on how to conceal a message and not on how to camouflage its transmittal. Nostega

addresses such shortcomings not only by camouflaging a message but also by its transmission. In Nostega, the steganographical goal is achieved by determining a suitable domain that is capable of generating an unsuspicious steganographic cover in which a message intrinsically is embedded in a form of innocent data that is compatible with the chosen domain. In addition, Nostega establishes a covert channel by employing the selected domain to serve as a justification for the interaction and delivering the cover among the communicating parties. A number of Nostega-based methodologies that are published: Graphstega, Chestega, linguistic-based steganography, including Matlist, NORMALS, Listega, Notestega, and Sumstega. The following is an overview of the Nostega architecture, which consisted of five modules:

1. Steganographic Field Determination (Module 1): Determines the fields such as education, economics, graphs, games, etc. for achieving the steganographic goal. One of the major selection criteria is how the steganographic field facilitates the process of generating a noiseless cover in which the data is naturally embedded so that the cover looks innocent raising no suspicion and the hidden message is undetectable. Note that the process of Module 1 is only involved at the stage of constructing Nostega-based system.
2. Steganographic Parameters Determination (Module 2): Encodes a message in an appropriate form for the camouflaging process (Module 3). The form and the component of the output of Module 1 may have essential effect of how a message can be encoded. Therefore, studying and analyzing the output of Module 1 is necessary for determining the parameters that can be used by next module (Message Encoder). In other words, this module is responsible for determining what parameters can be employed in order to implement a steganographic code that can encode messages in an effective way. For instance, if the steganographic field is a graph then the Steganographic parameters may be numerical values to plot the graph cover (Wayner, 2002). On the other hand, if the steganographic field is chess games, then the Steganographic parameters may be chess moves (Desoky & Younis, in press).
3. Implementing Message Encoder (Module 3): Implements a message encoder that is capable of accommodating the requirements of Nostega paradigm as stated earlier.
4. Implementing Cover Generator (Module 4): Constructing a cover generator or using a contemporary tool that is capable of achieving the steganographical goal. For instance, if the cover is graphs such as charts then employing a tool that is used by a wide variety of people such as Microsoft Excel maybe a good option in order to generate a steganographic cover that looks an ordinary graph. On the other hand, if the cover is chess then chess software such as Chessmaster (Desoky & Younis, in press) may legitimize the steganographic cover.
5. Implementing Communications Protocol & Covert Channel (Module 5): Configures the basic protocol of how a sender and a recipient would communicate covertly. It includes the covert channel for delivering a Nostega-based cover between the communicating parties along with the decoder scheme to unravel a hidden message. A covert channel can be based on a justifiable reason as in the scenario of Bob and Alice discussed above.

The advantages of Nostega are several. If the adversary succeeded in decrypting the ciphertext will be fooled by the outcome of reaching a plaintext while the original message is still protected. This is because the plaintext is a virtual ciphertext that conceal data. Nostega promotes the camouflaging of both a message and its transmittal. Nostega neither hides data in a noise (errors) nor produces noise rendering the generated cover noiseless. Instead, it conceals messages in a form of noiseless dada in the generated cover using either unaltered authenticated data or untraceable data thus avoiding wide varieties of attacks. The concealment process of Nostega has no effect on the linguistics of the generated cover if text is used as a steganographic carrier rendering such text-cover legitimate. Unlike other approaches like translation-based it can be applied to all languages. For steganographic carriers, Nostega uses materials such as graphs, text, games, etc., which have plenty of room for concealing data. The implemented metrologies that are based on Nostega paradigm are keyless schemes. Yet Nostega is a public paradigm, which implies that it is resilient even when an adversary is well familiar with this new paradigm. It is observed that a steganographic system is based on Nostega is capable of fooling both machine and human examinations.

3.3. Converting the Output of Phase 1 into Ciphertext (Phase 2)

Since the focus of this paper is fooling an adversary who succeeded in decrypting the ciphertext, this section is balanced on showing how Innocipher system achieves such. In other words, the goal in this paper is not to

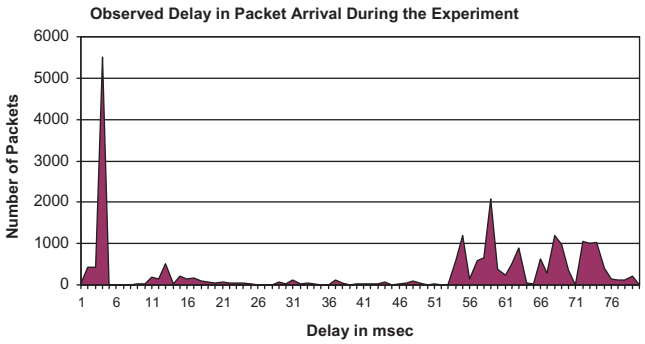
generate superior ciphertexts, but to secure valuable data even if a cryptosystem fails to do so. Employing a hard encoding system and cryptosystem to increase the protection of a message is obviously recommended and straightforward using any contemporary encoder or cryptosystem. Similarly, employing compression to boost the bitrate can easily be accomplished by using the contemporary techniques in the literature. Nonetheless, the intended users determents an appropriate cryptography scheme to achieve their goal. Generally, the goal of intended users may differ from one group of users to another. For instance, level of security and which attacks they are countering against. Moreover, intended users may intentionally decide to use a vulnerable cryptosystem to allow an adversary to decrypt a ciphertext to its plaintext in order to fool the enemy that the encrypted message contained innocent information. This mechanism is demonstrated in the next section.

3.4. Implementation and Validation Discussion

This section demonstrates the feasibility of Innocipher paradigm and its distinct capability of achieving the security goal. It is worth noting that the focus of this section is balanced on showing how Innocipher's capabilities are fooling an adversary rather than making it difficult for the adversary to decrypt a cipher message. The encrypted messages in this paper used an online tool (Online encryption, n.d.). Determining a sophisticated 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 the contemporary techniques in the literature.

3.5. Samples of Innocent Cipher by Employing a Nostega-based System

This section illustrates superiority of Innocipher paradigm to all other cryptosystems. Due to constrain of the size of this paper, this section will briefly demonstrate only two samples (Sample 1 and 2). Innocipher system in this paper employs Graph Steganography Methodology (Graphstega), which is one of many methodologies developed that are based on Nostega paradigm (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009, "Gaphstega," 2008; Desoky & Younis, 2006). Thus, Graphstega does not embed a message as a noise in a cover. Graphstega avoids the arousal of suspicion in covert communications by concealing a message as data point in a graph. This novel cover type is referred thereafter as a graph-cover. The popular usage of graphs in business, education, news, etc. and the availability of tremendous amount of graphs in electronic and non-electronic format make the investigation and detection of a hidden message extremely difficult. As known, Graphstega is resilient to contemporary attacks, such as traffic analysis, contrast, and comparison attack, even when launched by an adversary who is familiar with Graphstega (Desoky, 2012). Innocipher system inherits this robustness to achieve the security goal, which is protecting the concealed data even if a vulnerable cryptosystem used. Figure 2 (Sample 1) shows the sample of innocent graph that conceals data, which is then encrypted by a cryptosystem employed by Innocipher system. It is imperative to note that the presented graph in this section conceals worth of 16 pages of text. Similarly, Sample 2 shows both text-cover that conceals the original message and it cipher. If an adversary is capable of decrypting the ciphertext, then he will observe an ordinary and innocent graph, which is the plaintext, but obviously it is not the original message. The original message that is concealed in the presented graph in Figure 2 is the textual report of the Consumer Price Index (CPI) of July 2007. The size of this CPI report is 16 pages of text (Online encryption, n.d.). The bitrate of the presented example is roughly 21.51%. If an adversary succeeded to decrypt a ciphertext will get a plaintext that is not the original message, as shown by Sample 1 (Figure 2) and Sample 2. Nonetheless, this paper shows just an



Pages	16
Words	6,652
Characters with spaces	35,801
Lines	894
Paragraphs	701
Message size in bytes	46,923

FIGURE 2 **Illustrate Sample 1 in a form of graph-cover that conceals a long message (color figure available online).**

example of possible implementation following the steps outlined in the previous section while it is definitely expected that Innocipher can be implemented and applied differently.

Sample 2. In this sample, a message is concealed in a domain of Software Key License using Matlist (Desoky, 2012, "Matlist," in press), as shown below.

The original message is:

"Use my secret key"

The plaintext that conceals the message:

> Software Key License :
> 4393-109-83-4-54-115-33589045562022-105-33549048-2
> Please keep the tracking number. In case of calling customer support have the Software Key License ready.

Simply, the message is converted into numerical values and may include alphabet character as well (Desoky, 2012).

The final output of Innocipher system, the ciphertext, that conceals the above plaintext which is not the original message:

PIyu7MemqxhXlEp2acjcQfnbfj2c3yyYYXFhvi/s0h
Sf/EbnML3CshOSYm9lDaMXytyf0adFZE9+YG
CRHcUAydqSNIiTLupTaTT4PF1XEv5xqZJdj1ka
BOUfshNs7QFrWAZQxh7He0HQ45Nw4E88vL5
KfrgmqzmyjcMGvZKUFKaYk0lYEjjohRf1bMWR
YkCeHbgEdhhWU6T5XEuwe3NJrJgrSZ6TDbBD
nis2qk/Meo2WHrEU5EAkzw ==

4. SECURITY DISCUSSION

Since the focus of this paper is that it is assumed that an adversary succeeded in decrypting ciphertexts, then a cryptanalysis is not detailed here. Note that Innocipher system will inherent the strength of cryptosystem used. Thus, the cryptanalysis validation would be the same of that cryptosystem used. Therefore, the focus of attacks of this paper is beyond cryptanalysis. Nonetheless, Innocipher system will inherit the same strength of whatever cryptosystem used. Additionally, due to the size constrain of this paper, the aim of this section is simply to highlight the resilience key feature of Innocipher system against possible attacks. Nonetheless, the success of the Innocipher system is that if an adversary succeeded in decrypting a ciphertext, then they will get an innocent plaintext that does not reveal the original message. The Innocipher paradigm enables the fooling mechanism by engaging Nostega techniques with cryptography. This paradigm is qualified for securing data by its ability for avoiding an adversary's suspicion of the presence of a hidden message. When communicating parties in the stage of implementing an Innocipher system then they must assume that an adversary will perform all possible investigations. In addition, the adversary may also aware of Innocipher paradigm, as a public paradigm, but he does not know a particular Innocipher system configuration that the sender and recipient employ for their covert communication.

To illustrate Innocipher, consider the following scenarios. An adversary succeeded in decrypting a ciphertext to its plaintext form then there is no use of cryptanalysis techniques but the intuitive way of attacking such plaintext is that the only use of steganalysis techniques. Steganalysis techniques are the intuitive way of detecting hidden data. Simply, the presence of noise (flaws) may alert an adversary of concealing data. Additionally, the content of text-cover is often meaningless and semantically incoherent. These are just examples of detectable noise and it is not necessary to be linguistic flaws, for example, game-cover and graph-cover. The noise of such plaintext may be the presence of contradictions. For instance, finding contradictions in the CPI such as a CPI report detailing that a value of a particular product increasing when in fact it has actually decreased. In any case, such detectable noise by human or machine examinations can easily raise suspicion and defeat the security goal. However, the Innocipher paradigm is highly successful, in particular, due to its ability to disable self-actualization as it relates to the Humanistic Theory. This physiological lemma coined Roger and Maslow states that "humans are driven to achieve their maximum potential and will always do so unless obstacles are placed in their way" (Magee, Zachazewski, & Quillen, 2009). The plaintext induced by Innocipher acts as an obstacle, thus disabling the adversary's full potential, and fools the enemy into thinking that the given text is legitimate, when in fact it is not. The message then remains concealed and the adversary fails to reach his full potential.

4.1. Traffic Analysis

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 (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009, "Listega," 2009, "Graphstega," 2008; Desoky & Yoiunis, in press). For example, the intelligence community has a number of tools at their disposal for analyzing traffic on the internet, tracking access to web sites, monitoring checked out literature from public libraries, etc. The main goal of a traffic attack is to detect unusual or questionable associations 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 Innocipher, the profile of communicating parties and subjects of the revealed plaintext (steganographic cover) are checked rather than its validity and the consistency of its contents. If someone sends, receives, or accesses some materials without a legitimate reason for doing so, for example, a pretended deaf person receives song CDs from one of his friends, obviously suspicion can be raised and further investigation may be warranted. Similarly, it is suspicious if a medical doctor sends weather reports instead of medical reports. These types of communications can arouse suspicion after a ciphertext is successfully decrypted then an additional investigations will involve a thorough analysis of a steganographic cover, as detailed in the next subsections.

Traffic analysis is deemed ineffective with Innocipher. Innocipher camouflages the transmittal of a hidden message that is concealed in a plaintext to appear legitimate and thus suspicion is averted. Basically, Innocipher ensures that the involved parties establish a covert channel by having a well-plotted relationship with each other, rendering the communications traffic to appear innocent and look an ordinary communication. Analyzing the traffic between them will not reveal any questionable association and will not trigger any further investigation. In addition, Innocipher imposes on the communicating parties to use innocent domains, for example, contexts and martial, that retains high demand by a wide variety of people. Such domains 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 ciphertext without drawing attention, rendering Innocipher an attractive steganographical methodology to be used. Finally, it is noted that if further investigation on an Innocipher-text were triggered by traffic analysis, they would not be successful, as elaborated next. In Innocipher, differentiating between a plaintext (the decrypted message) that contains a hidden message and another peer plaintext without a hidden message is extremely difficult. This is because Innocipher is based on Nostega (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009, "Listega," 2009, "Graphstega," 2008; Desoky & Yoiunis, in press).

4.2. Contrast and Comparison Attacks

One of the intuitive sources of noise that may alert an adversary is the presence of contradictions in a plaintext. Examples of these contradictions may include finding suspicious repetition and flaws. Also, if a plaintext contains errors, it is not expected to be numerous or sever. Such contradictions may raise suspicion about the existence of a hidden message, especially when they are present in the same plaintext. As demonstrated in sections 3 and 4, the Innocipher system ensures that the generated plaintext, conceals data to appear legitimate, while the domain used is completely suitable for the communicating parties. Meanwhile, noise in the context of comparison attacks reflects alteration of authenticated (e.g., public image, audio, text) or previously used documents. The adversary's goal is to find any incorrect and inconsistent data that may imply the manipulation or alteration of contents of a plaintext that maybe used in order to embed a message. However, since Innocipher system based on Nostega paradigm, the comparison and contrast attacks are deemed ineffective (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009).

4.3. Linguistic Attacks

This is only used if the plaintext is contained text. Obviously, it cannot be used with content other than text, for example, graph, game, and image. Nonetheless, Linguistic examination distinguishes the text that is under attack from normal human language. This can be done through the examination of 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 generated plaintext retains the textual normality of legitimate text. As a result, the generated plaintext as demonstrated in the implementation section

is normal and legitimate text. Therefore, the plaintext of Innocipher system is capable of passing any linguistic attack by both human and machine examinations (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009).

4.4. Statistical Signature

As stated earlier, the Innocipher systems will inherent the strength of cryptosystem used and thus, the cryptanalysis validation would be the same of that cryptosystem used. In addition, since it is assumed that an adversary may succeed in decrypting the ciphertext, the focus of this attack in this paper is beyond cryptanalysis and on the plaintext, which is decrypted by an adversary. On the other hand, a statistical attack refers to tracking the profile of the used text. Nonetheless, 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 plaintext to detect any differences. An alteration in the statistical signature of a particular document may be a possible way of detecting a noise that an adversary would watch for. Unlike image steganography, tracking statistical signatures is an ineffective means for attacking linguistic steganography (Desoky, 2012). Nonetheless, Innocipher is resistant to statistical attacks because it is simply opt to naturally generate legitimate plaintext in which a message is embedded (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009). In addition, the generated plaintext by Innocipher system keeps the same profile of its other peer plaintext that contains no hidden message because it obey Nostega paradigm (Desoky, 2012, "Comprehensive," 2010, "Nostega," 2009, 2008).

5. CONCLUSION

This paper presents the novel Innocent-Cipher-based Cryptography Paradigm (Innocipher) that counters against adversaries who are capable of deciphering an encrypted text. An adversary may take advantage of recent advances in cryptanalysis techniques, a leakage of information about the cryptosystem or by other methods. Obviously, such an attack may easily defeat the cryptographic goal. Yet, an adversary may succeed in decrypting ciphertexts, while cryptographers unwarily continue using same vulnerable encryption techniques. These concerns motivate the development of Innocipher. Innocipher focuses on high level of security that protects private data through two phases as follows. First, the Innocipher system conceals the required data into a legible form of legitimate plaintext other than ciphertext, for example, legitimate text, graph, game, and image, that looks innocent and legitimate. Second, it converts the output of the first phase, the encoded data in the legible form, into a ciphertext. The main advantage of the Innocipher paradigm is that if a worst case scenario occurred, which is an adversary succeeding in decrypting a cipher message then he will be fooled. At this point, the adversary will stop any further investigation while a message is not revealed. This fooling mechanism of Innocipher is the key feature that enables a multilayer of security for protecting valuable information. In this paper, the presented implementation and validation of Innocipher demonstrated the robust capabilities of achieving the goal of securing information in a static stage and during data transmission to its legitimate recipient.

REFERENCES

Anderson, R. J., Needham, R., and Shamir, A. (1998). The steganographic file system. Proceedings of the Second International Workshop on Information Hiding. Lecture Notes in Computer Science, Vol. 1525, pp. 73–82.

Ansari, R., Malik, H., and 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), Vol. 5, pp. 17–21, 389–392.

Bolshakov, I. A. (2004). A method of linguistic steganography based on collocationally-verified synonymy. In J. J. Fridrich (Ed.), Information Hiding: 6th International Workshop. Lecture Notes in Computer Science, Vol. 3200, pp. 180–181. Berlin, Germany: Springer.

Bolshakov, I. A., and Gelbukh, A. (2004, June). Synonymous paraphrasing using wordnet and Internet. In F. Meziane and E. E. Metais (Eds.), Natural Language Processing and Information Systems: 9th International Conference on Applications of Natural Language to Information Systems, NLDB 2004. Lecture Notes in Computer Science, Vol. 3136, pp. 312–323. Berlin, Germany: Springer.

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

Chand, V., and 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., and Davida, G. (1997, November). Hiding the hidden: A software system for concealing ciphertext as innocuous text. Proceedings of the International Conference on Information and Communications Security. Lecture Notes in Computer Science, Vol. 1334, pp. 335–345. Berlin, Germany: Springer.

Chapman, M., and Davida, G. I. (2001). A practical and effective approach to large-scale automated linguistic steganography. Proceedings of the Information Security Conference (ISC '01). Lecture Notes in Computer Science, Vol. 2200, pp. 156–165. Malaga, Spain: Springer.

Chapman, M., and Davida, G. I. (2002). Plausible deniability using automated linguistic steganography. In G. Davida and Y. Frankel (Eds.), International Conference on Infrastructure Security (InfraSec '02).

Lecture Notes in Computer Science, Vol. 2437, pp. 276–287. Berlin, Germany: Springer.
Chapman, M., and Davida, G. I. (n.d.). Nicetext system official home page. Available from http://www.nicetext.com
CSRC. (2007, November). FIPS PUB 197: The official advanced encryption standard. Retrieved from http://csrc.nist.gov/groups/ST/hash/documents/FR_Notice_Nov07.pdf
Cvejic, N., and Seppanen, T. (2004). Reduced distortion bit-modification for LSB audio steganography. Proceedings of the 7th International Conference on Signal Processing (ICSP 04), Vol. 3 pp. 2318–2321.
Cvejic, N., and Seppanen, T. (2004, April). Increasing robustness of LSB audio steganography using a novel embedding method. Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04), pp. 533–537.
den Boer, B., and Bosselaers, A. (1993). Collisions for the compression function of MD5. Advances in Cryptology – EUROCRYPT '93. In T. Hesseseth (Ed.), Lecture Notes in Computer Science, Vol. 765. Berlin, Germany: Springer-Verlag.
Desoky, A. (2003). A novel hardware security methodology (HSM) for computers and networks. STOS 2003 Symposium Event (Secure Trusted Operating System Consortium), The George Washington University, Washington, DC.
Desoky A. (2008, January). Graphstega: Graph steganography methodology. Journal of Digital Forensic Practice, 2(1), 27–36.
Desoky, A. (2008, March). Nostega: A novel noiseless steganography paradigm. Journal of Digital Forensic Practice, 2(3), 132–139.
Desoky, A. (2008, July). Matlist: Mature linguistic steganography methodology. Technical Report TR-CS-08-02, Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore, MD.
Desoky, A. (2009, January). Notestega: Notes-based steganography methodology. Information Security Journal: A Global Perspective, 18(4), 178–193.
Desoky, A. (2009, April). Listega: List-based steganography methodology. International Journal of Information Security.
Desoky, A. (2009, May). Nostega: A novel noiseless steganography paradigm. (Doctoral Dissertation). University of Maryland, Baltimore, MD.
Desoky, A. (2010, July). NORMALS: Normal linguistic steganography methodology. Journal of Information Hiding and Multimedia Signal Processing, 1(3), 145–171.
Desoky, A. (2010, Spring). Comprehensive linguistic steganography survey. International Journal of Information and Computer Security, 4(2).
Desoky, A. (2012). Noiseless steganography: The key to covert communications. Boca Raton, FL: CRC Press.
Desoky A. (in press). Auto-summarization-based steganography. In Proceedings of the 5th IEEE International Conference on Innovations in Information Technology, Al-Ain, UAE, December 2008.
Desoky, A. (in press). Matlist: Mature linguistic steganography methodology. Journal of Security and Communication Networks.
Desoky, A. (in press). Jokestega: Automatic joke generation-based steganography. International Journal of Security and Networks.
Desoky, A., and 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., and Younis, M. (in press). Chestega: Chess steganography methodology. Journal of Security and Communication Networks. Online Encryption. (n.d.). Retrieved from http://infoencrypt.com
Diffie, W., and Hellman, M. (1976, June). Multi-user cryptographic techniques. AFIPS Proceedings 45, 109–112.
Diffie, W., and Hellman, M. (1976, November). New directions in cryptography. IEEE Transactions on Information Theory, IT-22, 644–654.
Dobbertin, H. (1996). Cryptanalysis of MD4. Fast software encryption. Cambridge Workshop. In D. Gollman (Ed.), Lecture Notes in Computer Science, Vol. 1039. Berlin, Germany: Springer-Verlag.
Dobbertin, H. (1996, May). Cryptanalysis of MD5. Rump Session of Eurocrypt 96. Retrieved from http://www.iacr.org/conferences/ec96/rump/index.html
El Gamal, T. (1983). A public key cryptosystem and a signature scheme based on discrete logarithms. CRYPTO 1984, 10–18.
El Gamal, T. (1983). A subexponential-time algorithm for computing discrete logarithms over GF(p^2). CRYPTO 1983, 275–292.
El Gamal, T. (1984). A subexponential-time algorithm for computing discrete logarithms over GF(p2). IEEE Transactions on Information Theory, 31(4), 473–481.
El Gamal, T. (1985). On computing logarithms over finite fields. CRYPTO, 396–402.
El Gamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory, 31(4), 469–472.
Gruhl, D., Lu, A., and Bender, W. (1996, May). Echo hiding. Proceedings of First International Workshop on Information Hiding. Lecture Notes in Computer Science, Vol. 1174, pp. 295–316, Cambridge, UK.
Handel, T. G., and Sandford, M. T. (1996). Data hiding in the OSI network model. Information Hiding: First International Workshop, Proceedings. Lecture Notes in Computer Science, Vol. 1174, pp. 23–38.
Ibrahim, A. Al-Kadi. (1992, April). The origins of cryptology: The Arab contributions. Cryptologia, 16(2), 97–126.
IETF. (n.d.). RFC 2440 – Open PGP Message Format. Retrieved from http://tools.ietf.org/html/rfc2440
Kahn, D. (1996). The code breakers: The story of secret writing (Rev. ed.). New York: Scribner.
Kirovski, D., and Malvar, H. (2001). Spread-spectrum audio watermarking: requirements, applications, and limitations. Proceedings of the 4th IEEE Workshop on Multimedia Signal Processing, pp. 219–224, Cannes, France, October.
Koblitz, N. (1994). A course in number theory and cryptography (2nd ed., pp. 54–76). Berlin, Germany: Springer.
Knudsen, L. R., and Mathiassen, J. E. (2005). Preimage and collision attacks on MD2. FSE, 255–267.
Luciano, D., and Prichett, G. (1987, January). Cryptology: From Caesar ciphers to public-key cryptosystems. The College Mathematics Journal, 18(1), 3.
Magee, D. J., Zachazewski, J. E., and Quillen, W. S. (2009). Pathology and intervention in musculoskeletal rehabilitation. Musculoskeletal Rehabilitation Series (MRS). St. Louis, MO: Elsevier.
Martin, A., Sapiro, G., and Seroussi, G. (2005, December). Is image steganography natural? IEEE Transactions on Image Processing, 14(12), 2040–2050.
Menezes, A. J., van Oorschot, P. C., and Vanstone, S. A. (1996). Handbook of applied cryptography. Boca Raton, FL: CRC Press.
Merkle, R. (1989). One way hash functions and DES. Advances in Cryptology – CRYPTO '89, In G. Brassard (Ed.), Lecture Notes in Computer Science, Vol. 435. Berlin, Germany: Springer-Verlag.
Muller, F. (2004). The MD2 hash function is not one-way. ASIACRYPT 2004, 214–229.
Murphy, B., and Vogel, C. (2007). 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, January.
Nakagawa, H., Sampei, K., Matsumoto, T., Kawaguchi, S., Makino, K., and Murase, I. (2001). Text information hiding with preserved meaning – a case for Japanese documents. IPSJ Transaction, 42(9), 2339–2350. (Originally published in Japanese)
NCUA. (2004, July). Letter to credit unions. Retrieved from http://www.ncua.gov/letters/2004/04-CU-09.pdf
Niimi, M., Minewaki, S., Noda, H., and Kawaguchi, E. (2003, August). A framework of text-based steganography using sd-form semantics model. IPSJ Journal, 44(8). Retrieved from http://www.know.comp.kyutech.ac.jp/STEG03/STEG03-PAPERS/papers/12-Niimi.pdf
NIST. (n.d.). Recommendation for block cipher modes of operation. Retrieved from http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

Rivest, R. (1990). The MD4 message-digest algorithm. Advances in Cryptology – CRYPTO '90. In A. J. Menezes and S. Vanstone (Eds.), Lecture Notes in Computer Science, Vol. 537, 303–311. Berlin, Germany: Springer-Verlag.
Rivest, R. (1992, April). The MD5 message-digest algorithm. IETF RFC 1321.
Rivest, R., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM, 21(2), 120–126.
Rogier, N., and Chauvaud, P. (1997). MD2 is not secure without the Checksum Byte. Designs, Codes and Cryptography, 12(3), 245–251.
Schneier, B. (1996). Applied cryptography (2nd ed.). New York, NY: Wiley.
ScramDisk: Free Hard Drive Encryption for Windows 95 & 98. (n.d.). Retrieved from http://www.scramdisk.clara.net
Stallings, W., and Brown, L. (2008). Computer security: Principles and practice. Upper Saddle River, NJ: Pearson Education.
Topkara, U., Topkara, M., and 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, NY: ACM Press.
Topkara, M., Topkara, U., and Atallah, M. J. (2007). Information hiding through errors: A confusing approach. Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents, January.
U.S. Bureau of Labor Statistics. (2007, July). Consumer Price Index. Retrieved from http://www.bls.gov/news.release/cpi.nr0.htm
Washington, L. (2003). Elliptic curves: Number theory and cryptography. Boca Raton, FL: Chapman & Hall/CRC.
Wayner, P. (1992). Mimic functions. Cryptologia, XVI/3, 193–214.
Wayner, P. (2002). Disappearing cryptography (2nd ed., pp. 81–128). Burlington, MA: 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 from http://alumni.imsa.edu/~keithw/tlex/lsteg.ps
Winstein, K. (n.d.). Lexical steganography. Retrieved from http://alumni.imsa.edu/~keithw/tlex

BIOGRAPHIES

Dr. Abdelrahman Desoky is a scientist and an ambitious Computer Engineering Doctorate with over twenty years of experience in the computer field. He is an experienced educator at both the graduate and undergraduate levels. Furthermore, he has industrial expertise in developing full life cycle systems such as software, hardware, security, and telecommunications/networks. Dr. Desoky received a Doctoral Degree (Ph.D.) from The University of Maryland, Baltimore County (UMBC) and a Master of Science (M.Sc.) from the George Washington University; both degrees are in Computer Engineering. His Doctoral Dissertation is entitled "Nostega: A Novel Noiseless Steganography Paradigm." The paradigm explores the topic of noiseless steganography, which refers to the science and art of covert communications. Nostega provides a way to secure information in static stage and during data transmission to a legitimate recipient. His M.Sc. degree concentrated on Computer Architecture and Networks. His research is entitled "Security Architecture for Computers and Networks." He is the CEO of The Academia Planet.