Uniform Resource Locator Based Steganography Methodology

Abdelrahman Desoky
(Corresponding author: Abdelrahman Desoky)
Department of Computer Science, Claflin University
400 Magnolia St, Orangeburg, SC 29115, USA
(Email: desoky@desoky.com)
(Received Dec. 30, 2017; revised and accepted Apr. 22, 2018)

Abstract

The Internet is widely popular and has become a culture of most, if not all, mankind worldwide. The use of the Internet is solely through accessing web-addresses like social-media, online-news, emails, *etc.* Obviously, this generates enormous traffic allowing communicating parties to establish a covert channel without a suspicious pattern. This renders the Uniform Resource Locator (URL) a highly attractive steganographic carrier to securely conceal and transmit messages. As a result, a novel URL-Based Steganography Methodology (URL-Stega) is presented in this paper. URL-Stega generates a steganographic cover in a form of a web-link. Simply, URL-Stega encodes a message then assigns it to steganographic carriers such as words, alphabet, numeric, alphanumeric, or other legible URL characters in order to camouflage data in the generated URL (Web Address). In addition, generated text-cover (URL-Cover) can be embedded among other legitimate non-coded text to make it harder for adversary to suspect and/or analyze such text. Unlike contemporary text-steganographic techniques, URL-Stega doesn't hide messages in the actual text body. Instead, URL-Stega conceals both the message and its transmission in innocent web-links rather than webpage contents. Yet, shortening the generated URL-Cover as any non-steganographic URL makes it more impressive to conceal data. URL-Stega neither hides data in a noise (errors) nor produces noise while a message is concealed in a legitimate URL. The presented implementation, validation, and steganalysis of URL-Stega demonstrate: robustness of achieving the steganographic goal, adequate room for concealing data, and superior bitrate than any other contemporary text steganography approaches from roughly about 39.47% up to 75.0%.

Keywords: Indicators; Social-media; Steganography; URL-Stega

1 Introduction

Steganography is the scientific art of concealing the presence of covert communications. The steganographic goal is to prevent an adversary from suspecting the existence of covert communications [7, 8]. Unlike cryptography, the aim of steganography is not to impede an attacker from deciphering a hidden message like ciphertext. To emphasize, if suspicion is raised when using any steganographic technique, the goal of steganography is defeated regardless of whether or not a plaintext is revealed [7]. Contemporary approaches in the literature are often classified based on the steganographic cover type like image, audio, graph [21], and text. When text is employed for hiding data and generating the steganographic cover, an approach is usually categorized as textual steganography to distinguish it from non-textual techniques like image or audio. Textual steganography has become more favorable in recent years because the size of non-textual cover is relatively large and burdens the traffic of covert communications [28].

Contemporary steganography, other than Nostega-based techniques, hides a message as noise in a cover that is assumed to be unnoticeable. For instance, an encoded message can be embedded into an image by altering it without noticeable degradation to human eyes [28]. Similarly, a message can be hidden in a text by modifying the format and style of an existing text [7, 8]. The alteration of authenticated covers may raise suspicion, and the message may be detectable regardless of whether or not a plaintext is revealed. The same applies for hiding data in unused or reserved space for systems software including designated storage area of an operating system, the file headers on a hardrive [30], or in the packet headers of communication protocols such as TCP/IP packets [26]. These techniques are vulnerable to distortion attacks [7].

On the other hand, a similar argument is made in the literature about textual steganography approaches such as: null cipher [7], mimic functions [36], NICETEXT and SCRAMBLE [5], translation-based [25], confusing ap-

proach [33], and abbreviation-based [31]. The vulnerability and concerns of these textual approaches are explained minutely in [7] and can be summarized as follows. First, the textual-cover either introduce detectable flaws (noise) such as: incorrect syntax, lexicon, rhetoric, or grammar when generating a text-cover. Such flaws can raise suspicion about the presence of covert communications. Second, the content of the cover may be meaningless and semantically incoherent, and therefore draw suspicion. Third, the bitrate is very small. Since there is a limit on how many flaws a document can typically have, a very large document is needed to hide few bytes of data. In fact, this applies to non-textual approaches as well. Fourth, the bulk of the efforts have been focused on how to conceal a message and not on how to conceal its transmittal. 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 inspections. A successful textual steganographic system (stegasystem) must be capable of passing both computer and human examinations. These concerns have motivated the development of the

URL-Based Steganography Methodology (URL-Stega), as introduced in this paper. URL-Stega overcomes the issues mentioned above by manipulating only the textual part of a web-link (Web Address/URL) to camouflage both a message and its transmittal. Fundamentally, URL-Stega exploits textual elements of URL such as words, alphabet, numeric, alphanumeric, and other legible URL characters in order to camouflage data in the generated Web Address. In addition, the generated text-cover (URL-Cover) can be embedded among other legitimate non-coded text to make it harder for an adversary to suspect and/or analyze such text. Unlike contemporary text-steganographic techniques, URL-Stega doesn't hide messages in the actual text body. To emphasize, URL-Stega does not conceal data in the actual content of a webpage. Instead, URL-Stega conceal both the message and it its transmission in innocent web-links. Shortening the generated URL-Cover as any non-steganographic URL makes it more impressive to conceal data. Such elements can be fabricated in a legitimate way in order to embed data without generating any type of suspicious pattern. Basically, URL-Stega encodes a message and then assigns it to legitimate elements (*e.g.* words, alphabet, numeric, alphanumeric, other legible URL's characters, *etc.*) in order to generate a text-cover in a form of a web address.

The main advantages of URL-Stega are as follows. First, the high demand for using Internet by a wide variety of people worldwide creates a high volume of traffic which averts suspicion in the presence of covert communication channels. Second, URL-Stega does not imply a particular pattern (noise) that an adversary may look for. Third, the concealment process of URL-Stega has no effect on the linguistics of the generated cover (URL-cover) because no linguistic structures are required in URL to be obeyed. Therefore, a URL-cover is linguistically legible, and as such is capable of passing both computer and human examinations. Fourth, URL-Stega can be applied to all languages. Fifth, the textual of URL-Cover has plenty of room for concealing data, as demonstrated later in the paper. The observed average bitrate of the current implementation experiments is superior to all contemporary textual steganography approaches found in the literature to be roughly around 3.38% up to 7.67%. Sixth, URL-Stega is resilient to all known attacks, and the hidden message is anti-distortion. It is worth noting that the presented methodology in this paper follows this new Nostega paradigm [7, 9, 10] by exploiting URL to camouflage data without generating any suspicious pattern. Examples of other Nostega-based system are Sumstega [11, 12], Listega [13], Notestega [14], Matlist [15], NORMALS [16], Edustega [17], Headstega [18], Jokestega [19], and Chestega [22]. The implementation and steganalysis validation demonstrate that URL-Stega methodology is capable of achieving the steganographical goal.

The remainder of this paper is organized as follows. Section 2 explains the URL-Stega methodology and its implementation in detail. Section 3 presents the steganalysis validation and its bitrate versus others. Finally, Section 4 concludes the paper.

2 URL-Stega Methodology

To illustrate URL-Stega, consider the following scenario. Bob and Alice are on a spy mission. Like any ordinary people, Bob and Alice access, send, and receive URLs from each other via chat, email, or by any other electronic means. Before they go on their mission, which requires them to reside in two different countries, they strategically plan and set the rules for communicating covertly using their friendship as a steganographic umbrella to justify sending and receiving messages. They agree on concealing messages only in URLs in such a way that does not look suspicious while the content of a webpage is legitimate and nothing is concealed in it. To make this work, Bob and Alice can legitimately send, receive, and forward emails, chats, posts, and texts to each other or to other individuals without suspicion. Covert messages transmitted in this manner will not look suspicious because the content of the webpage contains no hidden message except its web-link. Furthermore, Alice is not always the sole user or recipient of Bob's URL and vice versa. In other words, other non-spy people may also receive messages from Bob or Alice. As a result, suspicion is further warded off, thereby fooling an adversary. However, only Bob and Alice will be able to unravel the hidden message because they know the rules of the game.

2.1 An Overview of URL-Stega Architecture

The core idea of URL-Stega methodology is basically camouflaging data in the natural and legitimate URL.

Therefore, URL-Cover will look like any ordinary web address.

As shown in Figure 1, a legitimate URL shows the use of Google search engine when searching the word "test". Note that a web link looks like an illegible text. However, due to the use of URL, an adversary will be fooled in URL-Cover because URL will legitimize the text-cover. URL is an excellent means for camouflaging data due to the common use of illegible format of text that can contain a combination of alphabet, numbers, or special characters, without obeying any linguistic syntax. In addition, URL looks as if it is in a random format which makes it easy to embed encoded messages for covert communications.

Linguistically and logically, the URL format like the combination of characters used in Figure 1, qualifies URL as a legitimate steganographic cover. Additionally, URL-Stega methodology takes advantage of the heavy traffic of the Internet via accessing web-addresses, social-media, online-news, emails, and more to conceal both a message and its transmittal in a web link format.

URL-Stega Architecture:
The following is an overview of the URL-Stega architecture, which consists of three modules, as shown in Figure 1:

1) Determining the Set of Characters (Module 1) to be used for encoding messages.

2) Building URL-Stega Encoder (Module 2) that is capable of encoding and camouflaging messages using the determined characters format from Module 1.

3) Establishing a Covert Channel (Module 3) to embed an encoded message in order to camouflage the message and its transmittal in a sub domain name such as a web link.

The following subsections explain these modules in detail.

2.2 Determining the Set of Characters (Module 1)

Determining the set of characters to be used by the encoder (Module 2) for encoding messages, as discussed in Subsection 3.3. Simply, the maximum number of characters in the character set may be equal to all allowed characters in Uniform Resource Identifier (URI) and should not exceed its maximum unless there is a legitimate reason to generate an illegitimate web-link. Generating illicit web-link is not recommended without legitimacy because it can easily be detected. Therefore, this paper will only use a legitimate character set that is allowed in URI/URL while other characters are forbidden. Table 1 shows the allowed characters in URI [1–3]. URL-Stega may use the entire set of characters in Table 1 or a subset.

Table 1: Allowable characters by URI

RFC 3986 section 2.2 Reserved Characters (January 2005)		RFC 3986 section 2.3 Unreserved Characters (January 2005)				Other characters in a URI must be percent encoded.
	!		A	a	0	
	*		B	b	1	
	'		C	c	2	
	(		D	d	3	
	)		E	e	4	
	;		F	f	5	
	:		G	g	6	
	@		H	h	7	
	&		I	i	8	
	=		J	j	9	
	+		K	k	-	
	$		L	l	_	
	,		M	m	.	
	/		N	n	~	
	?		O	o		
	#		P	p		
	[		Q	q		
	]		R	r		
			S	s		
			T	t		
			U	u		
			V	v		
			W	w		
			X	x		
			Y	y		
			Z	z		

2.3 Building URL-Stega Encoder (Module 2)

Coding is a very well researched technical field, and there are numerous published techniques that can be employed to generate steganographic code [7,20]. Therefore, this subsection only focuses on key issues that affect the implementation of URL-Stega Encoder and building a URL-Stega Encoder that is capable of encoding messages using the determined character type and the format from Module 1. Table 1 shows a list of most of the characters that can be used in URI/URL. However, when building such encoder, a subset of Table 1 may suffice to achieve the steganographic encoder goal. Note that the character set must cover the entire length of binary code. In other words, if the length of a binary code equal n digits then the steganographic parameters must be capable to cover n digits from all of 0's and up to all of 1's (e.g. 00000-11111, length of 5 digits). To emphasize, 2 digits 00-11, 3 digits 000-111, 4 digits 0000-1111, 5 digits 00000-11111, 6 digits 000000-111111, and so on. This is to cover all possible binary values. Therefore, if 2 digits are selected, then 4 different symbols/characters are needed in order to cover 4 different binary values, and if 3 digits are chosen, then 8 different characters are needed in order to cover 8 different binary values. Thus, a message may be encoded by slicing its binary string into a particular length of bits such as four bits, seven bits, or any required bit length.

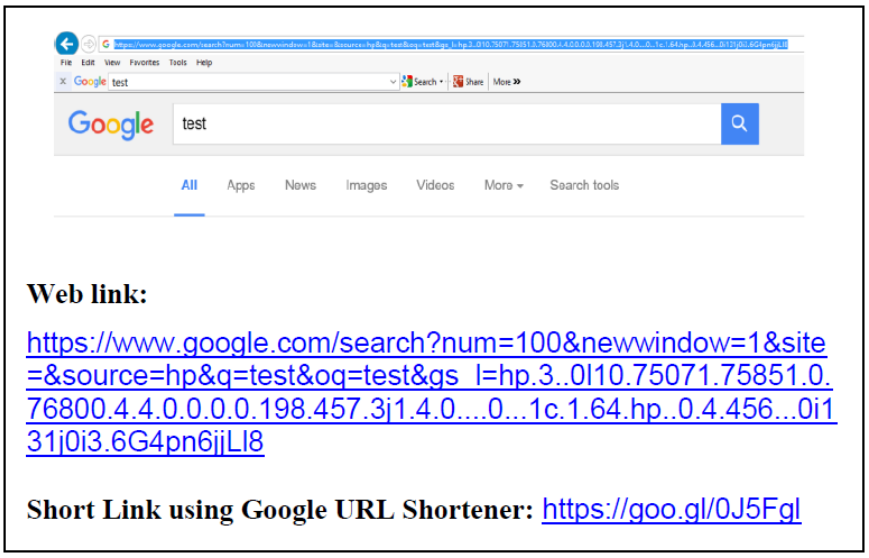

Figure 1: Shows a common web search result for the word "test" using the Google Search Engine. The search result is presented via web-link. Obviously, this web-link does not contain a hidden message and it is just an innocent and common practice of web search engine. The second web-link appears after shortening the first web-link using Google URL Shortener.

Example:

- A message in plaintext: "our meeting 8pm";
- The concatenated binary string of the ASCII representation of the above message is:

  01101111011101010111001000100000011101101
  01100101011001010111010001101001011011110
  01100111001000000011100001110000001101101

- Slicing this string (from the previous step) into 6 bits each will result in:

  011011 110111 010101 110010 001000 000110
  110101 100101 011001 010111 010001 101001
  011011 100110 011100 100000 001110 000111
  000001 101101

- URL-Cover of the binary code above, generated using Table 2 and Table 3, shows the mapping process for each character based on Table 2. The following is a pre-final URL-Cover before embedding it in domain name: "b3VyIg1lZXRpbmcgOHBt";
- Final URL-Cover, as shown in Figure 3, after embedding it in domain name (e.g. www.desoky.com) and it is ready to be delivered by accessing or sending it: "http://desoky.com/b3VyIg1lZXRpbmcgOHBt". Obviously, other existing domain names or generating new domain names can be used. In addition, URL-Cover can be shortened using any URL shortening tools, as shown in Table 4.

Table 2: 6 bit-based steganographic code table

URL-Stega Code	Binary Code	URL-Stega Code	Binary Code	URL-Stega Code	Binary Code
A	000000	a	011010	0	110100
B	000001	b	011011	1	110101
C	000010	c	011100	2	110110
D	000011	d	011101	3	110111
E	000100	e	011110	4	111000
F	000101	f	011111	5	111001
G	000110	g	100000	6	111010
H	000111	h	100001	7	111011
I	001000	i	100010	8	111100
J	001001	j	100011	9	111101
K	001010	k	100100	-	111110
L	001011	l	100101	_	111111
M	001100	m	100110		
N	001101	n	100111		
O	001110	o	101000		
P	001111	p	101001		
Q	010000	q	101010		
R	010001	r	101011		
S	010010	s	101100		
T	010011	t	101101		
U	010100	u	101110		
V	010101	v	101111		
W	010110	w	110000		
X	010111	x	110001		
Y	011000	y	110010		
Z	011001	z	110011		

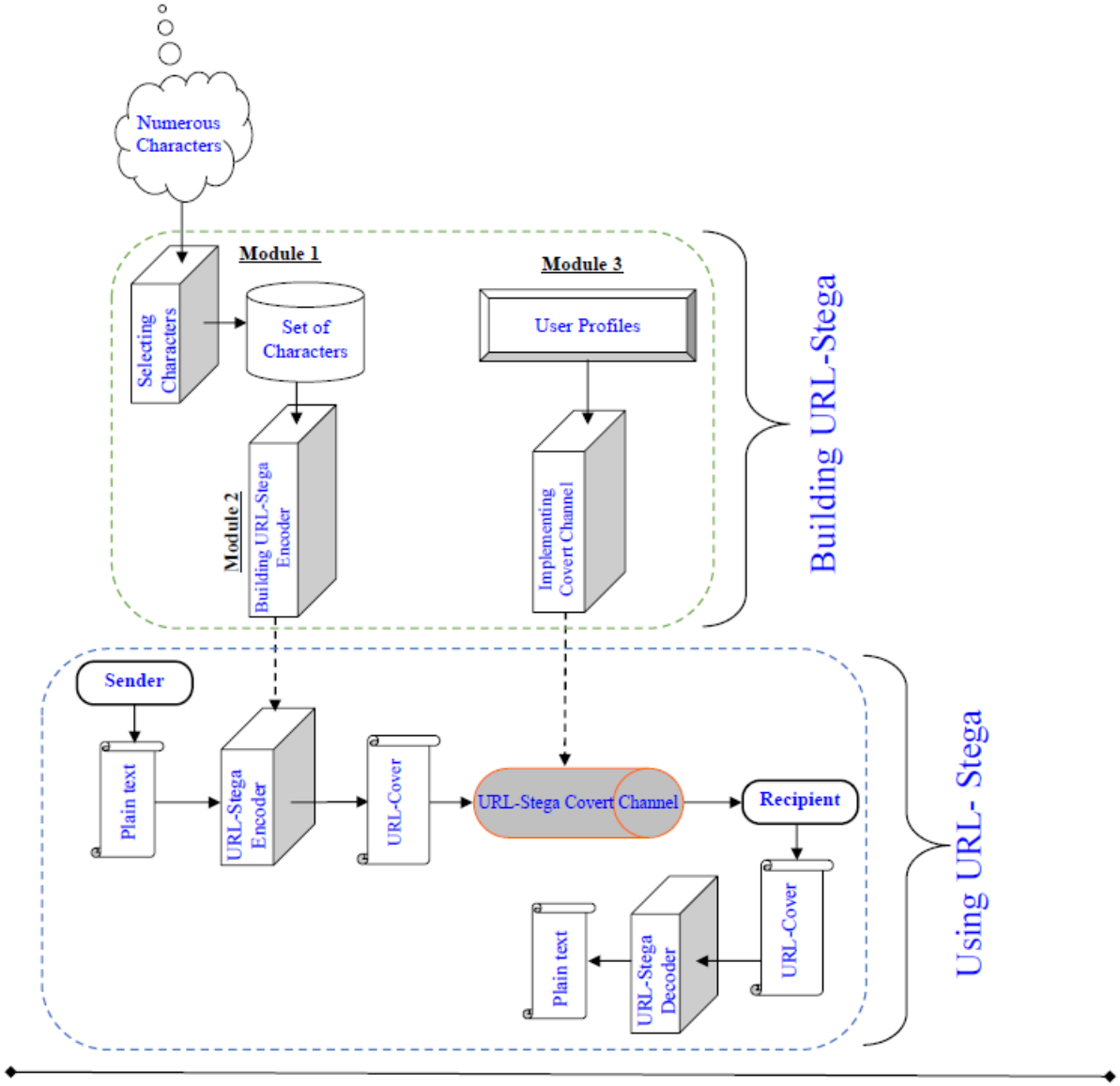

Figure 2: Illustrates the architecture and the use of URL-Stega. It shows the interaction of various modules to build URL Stega. Then, it shows the use of URL-Stega scheme by the communicating parties.

Table 3: Encoded message by encoding each 6 bits from Table 2

Index	Binary Message	URL-Cover
1.	011011	b
2.	110111	3
3.	010101	V
4.	110010	y
5.	001000	I
6.	000110	g
7.	110101	1
8.	100101	l
9.	011001	Z
10.	010111	X
11.	010001	R
12.	101001	p
13.	011011	b
14.	100110	m
15.	011100	c
16.	100000	g
17.	001110	O
18.	000111	H
19.	000001	B
20.	101101	t

- URL Checker can be used to avoid false link. There are some online tools that verify short URL in order to show the full URL, e.g. unfurlr.com [27], getlinkinfo.com [23], checkshorturl.com [6], unshorten.it [34], and urlxray.com [35].

In this paper, the steganographic code example is configured by determining the number of bits (m bits) used for binary values e.g. 4, 5, 7, *etc.* This is based on the available number of characters in the defined steganographic character set which is 64 characters according to Table 2. It is worth noting that Table 2 is derived from Table 1 as discussed earlier. The 64 characters in a steganographic character set are suitable to slice a binary message based on a length of 6 bits. The grouping in lengths of 6 digits will result in a value of 0 up to 63 in decimal and changing the value from 000000 up to 111111 in binary. Each character in URL-Cover conceals a particular m bits according to the steganographic code defined in Table 2. The current steganographic code is just a simple example to ease the understanding of the presented approach. The steganographic code may differ from one implementation to another and many alternatives with more sophisticated encoding techniques can be employed.

2.4 Establishing a Covert Channel (Module 3)

The frequent use of URL is widely popular and generates a high volume of traffic that allows communicating parties to establish a covert channel without a suspicious pattern. This makes web-link an attractive steganographic carrier for transmitting hidden messages. In this paper, Module 3 is responsible for embedding an encoded message in a domain name like a sub-link in order to generate a URL-Cover for concealing data. Unlike other steganographic approaches (e.g. image, audio, text, *etc.*) where a message is hidden in URL-Cover, it also addresses how a message is delivered. Specifically, a message is concealed in an image or audio file and then the file is delivered. Conversely, when concealing a message in URL, the message is delivered via accessing or sending the same URL. Thus, the steganographic cover is the same steganographic transmittal method, which is the covert communication channel. A sender will hide a message in a web-link, then a recipient will access or receive it via email, posting, chat, or by any other way. A sender and recipient may pre-agree on a particular URL domain name to use in order to hide and retrieve messages. Other ordinary people who are not part of the steganographic game can also access, send, and/or receive the same URL from each other for non steganographic purposes. Therefore, suspicion is warded off. Using URL makes it more legitimate and very difficult, if not impossible, to investigate. It is essential that legitimate users plot a convincing strategic plan and set the rules for communicating covertly using justifiable reasons as a steganographic umbrella to avert suspicious from covert communications. Basically, legitimate users have the right to use URL via accessing, sending, receiving, forwarding emails, chatting, posting, or texting each other. Covert messages transmitted in this manner will not look suspicious because such URL is sent via email, chat, posting, or texting while its content does not contain any hidden message except in the web-link. Therefore, such communication will be fully legitimate and justifies the discernable communications. The example presented in this paper conceals up to 120 bits in the URL-Cover. It is worth noting that the use of words and numerical values can be employed to conceal data. Due to the size constrain of this paper, the presented URL is just an example, and URL-Stega can conceal longer messages. After concealing data in URLs, web-links can also be combined with other web addresses (non-coded) that are not used to camouflage data for further protection and legitimacy. In this case, a predetermined-based protocol can be employed among communicating parties such as read every other URL, every fifth URL, or any other way in order to ease the process of unraveling a hidden message while making it harder on an adversary.

3 Steganalysis Validation

The aim of this section is to show the resilience of URL-Stega to possible attacks. The success of a steganographic approach is its ability of preventing an adversary from suspecting the presence of a hidden message. It is assumed that an adversary will perform all possible investigations,

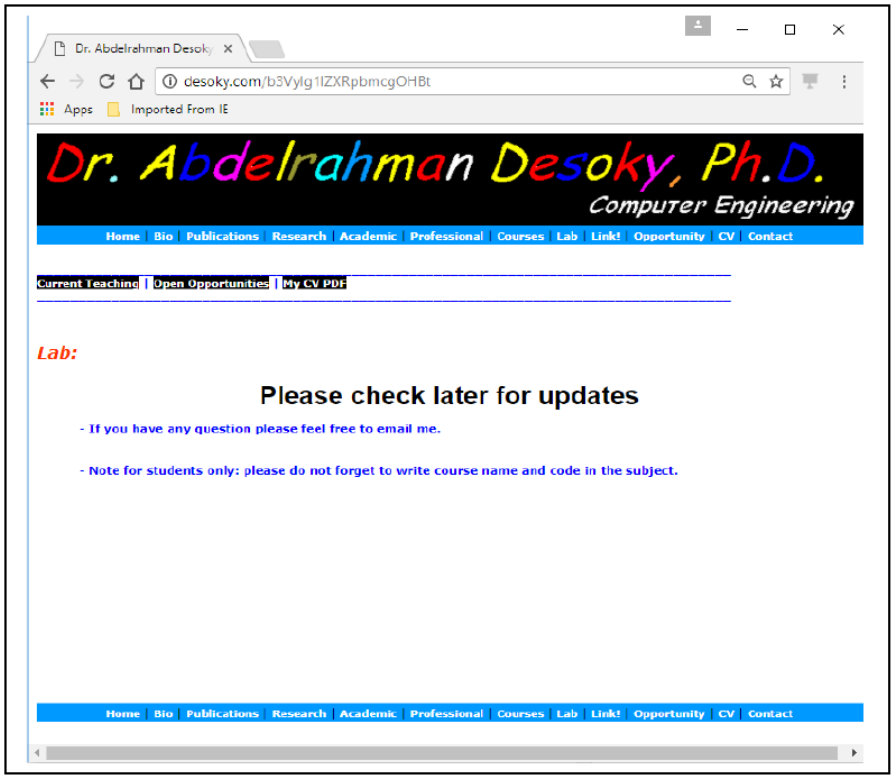

Figure 3: An actual example of URL-Cover. The message is not concealed in the web page content, but it is embedded in the URL.

and he is aware of URL-Stega as a public methodology. However, he does not know the actual URL-Stega configuration that the sender and recipient employed for their covert communication.

3.1 Traffic Attack

One possible attack an adversary may pursue is to inspect the communication traffic of images, graphs, audio files, and text files in order to detect the existence of covert communications. 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, and so on. 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 and their profiles (*e.g.* occupation, interest, hobbies, *etc.*) will play an essential role in either legitimizing or suspecting the presence of covert communications. Traffic attacks can be a threat for most contemporary steganographic techniques regardless of the steganographic cover types used. In regards to URL-Stega, the profile of users and webpage contents of particular URL are checked instead of its validity and its consistency to the URL's text body. In other words, the URL is most likely overlooked because no one will read or pay too much attention to it since it contains no meaningful information. For this reason, it is a common behavior that all Internet users accessing a web-address pay attention to the contents of the webpage rather than its web-address. On the other hand, if someone unrelated to the medical field such as a taxi driver, baker, or carpenter accesses, sends, and receives web-address for medical research without a justifiable reason, suspicion can be raised and further investigation may be prompted. Additional investigations may involve a thorough analysis of a steganographic cover as detailed in the next subsections.

Traffic analysis is deemed ineffective with URL-Stega. URL-Stega camouflages the transmittal of a hidden message (URL-Cover) making it appear legitimate, averting suspicion. URL-Stega is based on Nostega paradigm [9, 10]. URL-Stega by default ensures that the involved parties establish a covert channel. This is achieved by securing that the users have a legitimate relationship with each other and the used URL is justifiable. As such, the traffic of communication is innocent and appears like normal communication. Analyzing the traffic between them will not reveal any questionable association and will not trigger any further investigation. In addition, URL-Stega requires the communicating parties to use innocent URL domains like news, blog, or others that are commonly used by a wide variety of people. The common use of such domains generate a high volume of traffic which makes it impractical for an adversary to investigate all traffics.

Table 4: List of different short link of the same URL-Cover

Tool Name for Shortening the URL	Short URL-Cover
Google URL Shortener [24]	https:/goo.gl/3AcmRm
Bitly [4]	http://bit.ly/2rwqLic
Ow.ly via Hootsuite [29]	http://ow.ly/XZjk30cIKC4
TinyURL by Gilby [32]	https://tinyurl.com/y9zvza9w

The voluminous traffic plays a core role that allows the communicating parties to establish a covert channel in order to transmit a URL-Cover without drawing attention. As a result, URL-Stega is an attractive steganographic methodology. Finally, note that if further investigations on a URL-Cover are triggered by traffic analysis, they would not be successful as will be explained later. To sum, differentiating between a URL-Cover containing a hidden message with that of other URL peers without a hidden message is infeasible.

3.2 Contrast and Comparison Attacks

In text steganography, there are two ways of contrast attack [7]. First, contradictions between a profile of users and a URL/webpage subject which an adversary may look for. Such, contradictions reveal the fact of an unmatched subject of URL and its user profiles as discussed in traffic attack. This can be a sign of using steganographic tools. URL-Stega scheme is resilient against such attack by establishing a covert channel that guarantees to match the user profiles to a webpage subject. Second, another type of contradictions may exist in the text, and an adversary may try to look for them only inside a text-cover. This is unlike the first type of contradictions where the text is compared to its user profiles. When a piece of text contains this type of contradictions, most likely the text is incoherent. Whether or not a URL-Cover contains contradictions in its textual URL, suspicion will not be triggered because a URL is not intended to be read. In other words, a URL is not a text that obeys linguistics rules such as syntax and grammar as such it is not intended to be read or contain information. This is a strong natural immunity for URL-Cover against contrast attack.

Unlike contrast attack, comparison-attack attempts to detect alterations in an authenticated text. To emphasize, an adversary's goal is to employ comparison-attack to find any modifications between the original text and the target-text that may reveal the manipulation of content to embed a message. For example, if an adversary compares an article to its original and detects alteration, it implies a steganographic tool was used. However, comparison attack cannot be used against the presented approach because URL-Cover is not a textual document like a news article that can be subject to alteration. Therefore, URL-Cover is naturally resistant against comparison attack too.

3.3 Linguistics Attacks

Linguistic examination distinguishes the text that is under attack from normal human language which can be done via inspecting the meaning, syntax, lexicon, rhetoric, semantic, coherence, and any other features that can help in detecting or suspecting the existence of a hidden message. These examinations are used to determine whether or not the text under investigation is normal. Obviously, the URL is a type of text that link users to a webpage, and it is not an informative text to be read. No one pays attention to such text (URL). Conversely, everyone will pay attention to the contents of a webpage rather than its URL. This is a common behavior of all Internet users due to the fact that there is nothing to be read in a web-address itself. A web-address (URL) may contain weird text, as shown in Figure 1, which will ease the generating process of URL-Cover and help legitimize it. This is very noticeable in looking to a number of web-links. For example, when searching the web, the URL of the search result will contain abnormal text. In this paper, a text abnormality means that a text neither obeys linguistic syntax nor correct spelling of any legitimate languages. Generally, in text steganography when detecting noise (text abnormality) the goal of steganography is defeated regardless of whether or not a plaintext is revealed. However, this is not the case in the URL-Stega because it is very common and legitimate that the URL contains such abnormality which makes web-link an attractive steganographic carrier for concealing data.

The text used in URL is a different type of text that follows only the rules of URI rather than following the rules of normal language like syntax, grammar, and so on. Investigating the textual URL-Cover should be based on the rules of URI such as the permissible characters, as shown in Table 1. URL-Stega methodology requires the implementation process to obey all the rules of URI. One may say a wrong web-link that violates the rules of URI can also be used because there are so many users that send, receive, and attempt to access incorrect web-addresses. However, this may trigger suspicion because it is not a common practice to frequently use a wrong web-link. Additionally, when using incorrect web-link, the detection of violating URI's rules can easily be achieved. URL-Cover does not use sophisticated text, and it is easy for such scheme to retain the textual normality according to URI rules. Yet, there is no linguistic structure to be obeyed in URL and thus it does not generate any noise

(linguistic flaws). As a result, the generated cover is normal text, as demonstrated in the implementation section. Therefore, URL-Stega is capable of passing any linguistic attack by both human and machine examinations.

However, a statistical attack tracks a profile of the text used. A statistical signature (profile) of a text may refer to the frequency of words and characters used. An adversary may use the statistical profile of a particular topic for documents that contains no hidden message and compares it to a statistical profile of the suspected URL-Cover 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 textual steganography [7, 10, 25]. Nonetheless, URL-Stega is resistant to statistical attacks because it uses legitimate textual URL that is generated based on URI rules. In addition, the generated textual cover (URL-Cover) retains the same profile of its peers' text that contains no hidden message. Basically, most alterations introduced by URL-Stega are nonlinguistic and do not produce any flaws (noise), as demonstrated in the implementation section. As a result, statistical attacks on URL-Cover is ineffective.

3.4 Bitrates

The aim of this section is to evaluate the presented URL-Stega bitrate to contemporary textual steganography approaches. The bitrate is defined as the size of the hidden message relative to the size of the cover. The average bitrate of the presented URL-Stega system used in this paper is roughly between 39.47% and 75.0%. It is worth noting that the bitrate may differ from one element to another and from one implementation to another, as observed. To put this bitrate figure in perspective, the bitrate of contemporary textual steganography approaches has been investigated and for more information refer to [7]. Tables 5 and 6 summarize the findings of the bitrate and categorize them based on the pursued approaches.

Table 5: The bitrate of URL-Cover with and without shortening it

Tool Name for Shorting the URL	Bitrate
Without shorting the URL	39.47%
Google URL Shortener [24]	75.0%
Bitly [4]	71.42%
Ow.ly via Hootsuite [29]	62.5%
TinyURL by Gilby [32]	53.57%

Table 6: The bitrate of contemporary textual steganography approaches other than Nostega-based approach as discussed in [7]

Approach	Bitrate
Mimic functions	0.90%
NICETEXT	0.29%
Winstein	0.5%
Murphy et al.	0.30%
Nakagawa et al.	0.034%
Translation-based	0.33%
Confusing	0.35%

4 Conclusion

The high demand of using the Internet by a wide variety of people makes it feasible for communicating parties to establish a covert channel for transmitting hidden messages (URL-Cover). Thus, URL is an attractive steganographic carrier. Such features motivated the development of the URL-Based Steganography Methodology (URL-Stega). URL-Stega conceals data only in legitimate textual URL/web-address. URL-Stega neither hides data in a noise (errors) nor produces noise. Instead, it camouflages data by exploiting elements that are allowed by URI rules, such as alphabet, numeric, alphanumeric, abbreviation, words, and other legible URL characters in order to construct a URL-Cover that looks innocent. The bitrate of the presented implementation in this paper is roughly about 39.47% and up to 75.0%. This bitrate is superior to all other contemporary text steganography approaches found in the literature and it confirms the effectiveness of URL-Stega. The steganalysis validation shows that URL-Stega methodology is capable of achieving the steganographic goal.

References

[1] T. Berners-Lee, L. Masinter, M. McCahill, *Uniform Resource Locators (URL)*, RFC 1738, Dec. 1994. (`https://tools.ietf.org/html/rfc1738`)

[2] T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifiers (URI): Generic Syntax*, RFC 2396, Aug. 1998. (`https://tools.ietf.org/html/rfc2396`)

[3] T. Berners-Lee, R. Fielding, L. Masinter, *Uniform Resource Identifier (URI): Generic Syntax*, RFC 3986, Jan. 2005. (`https://tools.ietf.org/html/rfc3986`)

[4] Bitly, *Harness Every Click, Tap And Swipe*, June 23, 2018. (`https://bitly.com`)

[5] M. Chapman, G. I. Davida, "Plausible deniability using automated linguistic steganography," in *International Conference on Infrastructure Security (In-*

fraSec'02), Lecture Notes in Computer Science, vol. 2437, pp. 276-287, 2002.

[6] CheckShortURL, *URL Checker*, June 23, 2018. (`http://www.checkshorturl.com`)

[7] A. Desoky, *Noiseless Steganography: The Key to Covert Communications*, Information Security Publisher/CRC Press/Taylor & Francis Group, 2016.

[8] A. Desoky, "Comprehensive linguistic steganography survey," *International Journal of Information and Computer Security*, vol. 4, no. 2, pp. 164-197, 2010.

[9] A. Desoky, "Nostega: A novel noiseless steganography paradigm," *Journal of Digital Forensic Practice*, vol. 2, no. 3, pp. 132-139, Mar. 2008.

[10] A. Desoky, *Nostega: A Novel Noiseless Steganography Paradigm*, Ph.D. Dissertation, University of Maryland, Baltimore County, May 2009.

[11] A. Desoky, "Sumstega: Summarization-based steganography methodology," *International Journal of Information and Computer Security*, vol. 4, no. 3, pp. 234-263, 2011.

[12] A. Desoky, *et al.*, "Auto-summarization-based steganography," in *Proceedings of the 5th IEEE International Conference on Innovations in Information Technology*, Dec. 2008.

[13] A. Desoky, "Listega: List-based steganography methodology," *International Journal of Information Security*, vol. 8, no. 4, pp. 247–261, 2009.

[14] A. Desoky, " Notestega: Notes-based steganography methodology," *Information Security Journal: A Global Perspective*, vol. 18, no. 4, pp. 178-193, Jan. 2009.

[15] A. Desoky, "Matlist: Mature linguistic steganography methodology," *Journal of Security and Communication Networks*, vol. 4, no. 6, pp. 697-718, 2011.

[16] A. Desoky, "NORMALS: Normal linguistic steganography methodology," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 3, pp. 145-171, 2010.

[17] A. Desoky, "Edustega: An education-centric steganography methodology," *International Journal of Security and Networks*, vol. 6, no. 2/3, pp. 153-173, 2011.

[18] A. Desoky, "Headstega: E-mail-headers-based steganography methodology," *International Journal of Electronic Security and Digital Forensics*, vol. 3, no. 4, pp. 289-310, 2010.

[19] A. Desoky, "Jokestega: Automatic joke generation based steganography," *International Journal of Security and Networks*, vol. 7, no. 3/4, pp. 148-160, 2012.

[20] A. Desoky, "Innocipher: A novel innocent-cipher-based cryptography paradigm - High level of security for fooling the enemy," *Information Security Journal*, vol. 22 , no. 2, 2013.

[21] A. Desoky, M. Younis, "Graphstega: Graph steganography methodology," *Journal of Digital Forensic Practice*, vol. 2, no. 1, pp. 27-36, Jan. 2008.

[22] A. Desoky and M. Younis, "Chestega: Chess steganography methodology," *Journal of Security and Communication Networks*, vol. 2, no. 6, pp. 555-566, Mar. 2009.

[23] GetLinkinfo, *URL Checker*, June 23, 2018. (`http://getlinkinfo.com`)

[24] Google, *Google URL Shortener*, June 23, 2018. (`https://goo.gl`)

[25] C. Grothoff, *et al.*, "Translation-based steganography," in *Proceedings of Information Hiding Workshop (IH'05)*, pp. 213-233, June 2005.

[26] T. G. Handel, M. T. Sandford, "Data hiding in the OSI network model," in *First International Workshop, Proceedings on Information Hiding*, Lecture Notes in Computer Science, vol. 1174, Springer, pp. 23-38, 1996.

[27] MailChimp, *What's behind that short link?*, June 23, 2018. (`http://unfurlr.com`)

[28] A. Martin, G. Sapiro, G. Seroussi, "Is image steganography natural?" *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2040-2050, Dec. 2005.

[29] Ow.ly via Hootsuite, *Ow.ly: Now Inside the Hootsuite Dashboard*, June 23, 2018. (`http://ow.ly`)

[30] SecuriTeam, *ScramDisk - Disk Encryption Tool*, Jan. 4, 2000. (`http://www.securiteam.com/tools/5VP011F0BY.html`)

[31] M. Shirali-Shahreza, *et al.*, "Text Steganography in SMS," in *International Conference on Convergence Information Technology*, pp. 2260-2265, Nov. 2007.

[32] TinyURL by Gilby, June 23, 2018. (`http://tinyurl.com` and `http://www.gilby.com`)

[33] M. Topkara, U. Topkara, and M. J. Atallah, "Information hiding through errors: A confusing approach," in *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents*, Jan. 2007.

[34] Unshorten.It, *URL Checker*, June 23, 2018. (`https://unshorten.it`)

[35] URL X-ray, *URL Checker*, June 23, 2018. (`http://urlxray.com`)

[36] P. Wayner, "Mimic functions," *Cryptologia*, vol. 16, no. 3, pp. 193-214, 1992.

Biography

Dr. Abdelrahman Desoky is an associate professor at Claflin University. He is an experienced scientist researcher and educator at both the graduate and undergraduate level with over twenty years of IT experience in the academic and industrial sectors. Dr. Desoky received a Doctoral Degree (Ph.D.) from the University of Maryland, Baltimore County, 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, with research focusing on Security Architecture for Computers and Networks. He is the author of security book entitled "Noiseless Steganography: The key of Covert Communications".