A PROPOSED MODIFIED TEXT STEGANOGRAPHY TECHNIQUE USING UNISPACH WITH XOR ENCRYPTION AND SHIFT CIPHER

Raka Adinugraha, Tito Waluyo Purboyo and Randy Erfa Saputra
Department of Computer Engineering, Faculty of Electrical Engineering, Telkom University, Bandung, Indonesia
E-mail: rakaadi264@gmail.com

ABSTRACT

Nowadays, security on communication was a necessity. However, sending an encrypted message can draw a suspicion from unintended parties. So, sometimes cryptography doesn't guarantee a full security because it could attract attempts to break and reveal the encrypted message. Steganography was introduced as a method to secure a secret message by hiding it inside an unsuspicious message. The message can be plain text or other data that can be represented as streams of bits. Many of steganography techniques are being proposed from time to time, means steganography is a promising method to secure a communication line beside cryptography. In this paper, an experiment is conducted by comparing two text steganography techniques, which is UniSpaCh and a steganography technique by altering the foreground color of an invisible character.

Keywords: steganography, bits, text, hiding, suspicion.

1. INTRODUCTION

The rapid development of communication technologies demands a technique to secure a communication line between parties. Especially when a sensitive data are involved, such as medical data, transaction detail, banking data etc., those data should only be obtained by the intended recipient. Cryptography is often used to secure those, but it still raises suspicion from third parties if spotted. Then, it could lead to attempts to crack and break the encrypted message to reveal the original one. In other ways, steganography was used to hide the presence of the secret message beneath a suspicious message. It really shows it advantages when it comes to digital text documents, it has a lot of features to be used a media to hide a secret message in steganography. Text steganography is a technique to a hide a text message within another cover text. The goal of steganography is to make the presence of the original message not realized by unintended parties by hiding it through innocuous cover when transmitted between communication lines. Moreover, steganography is such a good addition to cryptography to ensure even more security.

2. TEXT STEGANOGRAPHY

In Figure-1, a basic text steganography scheme was shown. First, by applying an embedding algorithm, a secret message will be hidden in a cover text. Then, it will produce a stego text. Next, the stego text will be sent through a communication channel, internet for example. In the recipient side, in order to recover the secret message hidden beneath the stego text they need to apply a decoding algorithm. The purpose of the key is to prevent easy detection and/or recovery of the secret message from unintended parties and assist in hiding (encoding) process.

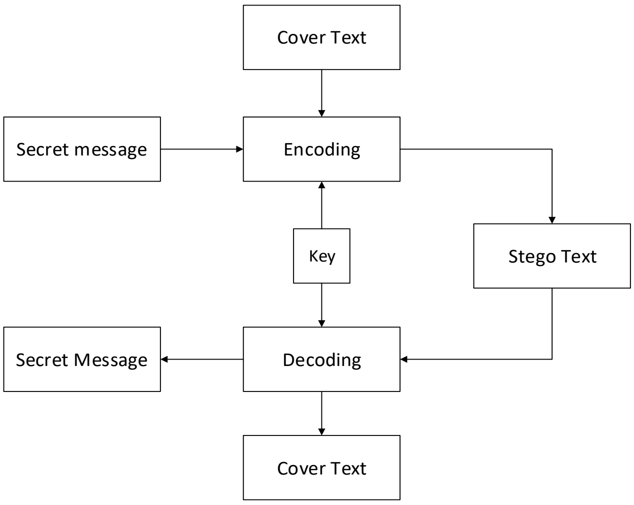

Figure-1. Simple scheme of text steganography.

In [2], text steganography was classified into three categories, random and statistical generation, a format based, and linguistic method.

Random and statistical generation is when a cover text generated according to statistical properties. This method based on a sequence of word and character. The information hidden in the character or word within the cover text must be to be random otherwise it could be raise unnecessary suspicion when the cover text got intercepted by unintended parties.

Format based apply a formatting on the physical aspect of the text document in order to hide the secret message [1]. Aspect such as font size, a space character, character color can be altered to hide a secret message on it. It is difficult to detect the change with human eyes, but it can be detected with a computer program.

Linguistic method uses the linguistic aspect that appears in the text document to hide the message. Abbreviation, and synonyms were an example of it, even an errors can be uses to hide the secret message [3].

3. RELATED WORK

WhiteSteg uses white spaces to hide the secret message [1]. The white spaces that are being use for data hiding is inter-word spaces, inter-sentence spaces, and end of line spaces. For example, bit 0 and 1 are represented by a space character in their respective place. In inter-word spaces, a single space could represent bit 0 while a couple of space could represent bit 1. For, inter-sentence spacing it could be the same on how the space characters are used but they are being placed in the last character of a sentence. In end of line spaces, a single spaces can be use to represent bit 0, two spaces to represent bit 1, and so on.

Khairullah [4] proposed a text steganography technique which is alter RGB value of invisible character, such as normal space and tab space. First, every character in the message needs to be converted into binary value, then convert those binary value into their respective decimal value. The decimal value will be place as a RGB value of invisible character (normal space or tab space). The color attribute of those two characters will not appear to the user like any other visible characters, so it could potential to hide a data in form of RGB value of those characters.

Mahato *et al.* [5] proposed a text steganography technique that generating Huffman code based on non-occurance probability of a synonym sets. The synonym sets is based on the words that appear on the cover text. The synonyms sets are used to matched with Huffman code of respective word synonyms to embed the bits of the secret message.

Line Shifting [6], this text steganography technique was done by moving a text line whether vertically up or down. The data that want to be hidden will determined whether a text line will be moved up or down, for example if it is bit 0 then a text line will have to moved up and if it is bit 1 the text line have to moved down or otherwise.

Word Shifting [6], this technique perform a horizontal shifting of a particular words to hide a secret message. A word need to be shifted to the left to hide bit 0 and shifted to the right to hide a bit 1 or otherwise. If a cover text have a very varying distance between words, this technique will be very suitable because it will look lees recognizable.

Feature coding [6], a feature that available in the cover text altered to hide a secret message. A feature example for a text is vertical lines in characters such as b, d, h can be altered either shortened or extended to hide a message.

Listega [7], this text steganography technique works by concealing a secret message in the list-based cover text. The list could be anything that is common to write it in a text list such as shopping list, book list, music list, etc. The textual list will be send through a communication line that have been acknowledge by every parties that involved, thus make this technique very robust because only the involved parties know the rule of this steganography technique.

Shahreza *et al.* [9] proposed another text steganography technique by using word synonyms to conceal a secret message. It use synonym based on US and UK terms in English, for example ‘Movie’ is an US term of ‘Film’ in UK. Then the synonyms are choose whether is an US term or an UK term to hide a bit 0 or a bit 1 respectively, at the end the extracted data will be saved in user computer.

4. UNISPACH

This particular text steganography technique utilizes Unicode space character to conceal a secret message in a text document with Microsoft Word [10]. The Unicode space characters were inserted in the white space available in the document. Three spaces are eligible for hiding data in the UniSpaCh, it was inter-word spacing, inter-sentence spacing, and end of line spacing.

Out of 18 Unicode space characters available, only 8 are eligible for data hiding purposes [10]. Because when ‘Show/Hide Formatting Marks’ are use in Microsoft Word that shows us any kind of formatting indication such as space characters and tab space, the rest of Unicode Space Character will appear as square and degree symbol. Those 8 Unicode Space Character that will be use to data hiding purposes is shown in table 1 below.

Table-1. Unicode space characters suitable for data hiding.

Code	Name
U+2000	En-Quad Space
U+2001	Em-Quad Space
U+2004	Three-Per-Em Space
U+2006	Six-Per-Em Space
U+2007	Figure Space
U+2008	Punctuation Space
U+2009	Thin Space
U+200A	Hair Space

Furthermore, for increasing the embedding efficiency two groups of a rule are formed to use for embedding the secret message. Group (A), for inter-word spacing and inter-sentence spacing. Group (B), for end of line and inter-paragraph spacing [10].

Table-2. Representation scheme for group.

Inter Word and Sentence Spacing	
Combination	**Sequence**
Normal Space	00
Thin + Normal	01
Six-Per-Em + Normal	10
Hair + Normal	11

In Table-2 for group A, if a simple replacement approach were used, that just replaces any ordinary space with selected 8 Unicode space characters, it could make the spacing between word and sentence looks weird either it too narrow or too wide. Therefore, the Unicode Space Characters that occupy the smallest width that was Thin, Six-Per-Em, and Hair, combined with ordinary space to encode the secret message in a sequence of 2 bits.

Table-3. Representation scheme for group.

Inter Word and Sentence Spacing	
Combination	**Sequence**
Hair	00
Six-Per-Em	01
Punctuation	10
Thin	11

In Table-3 for group B, the Unicode space character that used for data hiding is Hair, Six-Per-Em, Punctuation, and Thin. Same as group A, this 4 will encode a secret message in a sequence of 2 bits.

5. BASIC CONCEPTS

This section will be give a brief explanation about some theory and concepts that will be used in the experiment with the propose technique in this paper.

5.1 XOR operations

The Exclusive or (XOR) also known as exclusive disjunction is a logical operation that output are true when the same input were given, either way is false. Table-4 below is the truth table of XOR operations.

Table-4. XOR operations truth table.

Input		**Output**
p	**q**	
0	0	0
0	1	1
1	0	1
1	1	0

XOR operations can be use as an encryption method. Its commonly use as a part of more larger and complex encryption system. The idea is to combine the XOR encryption process with another encryption system such as shift cipher for example [8].

5.2 Substitution cipher

In cryptography substitution cipher also known as Caesar's Cipher. It is one of the most widely known encryption techniques. In this cipher, each letter in the plain text will be replaced by another letter in the alphabet. The replacement will be according to how many shift count that use in the encryption process. For example, with a right shift of 4, A would be replaced by E, B would be replaced by F, and so on. Nowadays, substitution cipher often use as a complementary to more complex encryption scheme. Below is example of susbtitution cipher with a right shift of 4.
depart on thursday

hativx sr xlyvwhec

Decryption is done by a left shift of 4.

6. PROPOSED TECHNIQUE

The experiment conduct using this proposed techniques, is done within Microsoft Word document. Embedding a secret message with length of 96 bit to some cover text that has been randomly pick regardless of the content. In the end, we try to compare the resulting stego text form this proposed technique, with the stego text resulting from the original UniSpaCh technique.

Table-5. Substitution cipher table.

Original	**Shifting**
a	e
b	F
c	g
d	h
e	i
f	j
g	k
h	l
i	m
j	n
k	o
l	p
m	q
n	r
o	s
p	t
q	u
r	V
s	w
t	x
u	Y
v	z
w	a
x	b
y	c
z	d

Table-5 shows the details of substitution cipher, done by doing a 4 right shift. Decryption is done by doing another four left shift.

6.1 XOR cipher

It is accomplished by performing double XOR with 2 different 16 bit keys. The first key was *x1,* and the second key was *x2*.
x1 = 0111001001100001;
x2 = 0110101101100001;

The can be expanded to 32 bit and so on. Also, to ensure more secure and more robust encryption the key can be generated by pseudo-random generator or using a one-time pad key is also recommended.

6.2 Work steps

This section shows step-by-step of the work flow of our proposed technique. Start form encoding process to the end of decoding process that will ultimately retrieve back the original message that have been encoded.

Encoding
Input : Secret message
Output : Stego text

a) Read the message that will be embedded into the cover text.
b) For each letter that present in the message, do a letter shifting process.
c) After letter shifting process complete, convert every letter to their binary value by looking up to ASCII code table.
d) Then, do double XOR to all binary value with two different key (x1 and x2).
e) The result of the double XOR process will be embedded to selected cover text using UniSpaCh technique.

Decoding
Input : Stego text
Output : Secret message

a) Use 'Find and Replace' feature to find and identified every Unicode space character that embedded using UniSpaCh technique.
b) Make a note about where and which Unicode space character has been place and use to make it easier to decode it and prevent unnecessary error.
c) Then, do double XOR process again but in reverted order using the previous key (x1 and x2).
d) Make a group consist of 8 bits from all the binary value that resulted from the previous process.
e) Then, convert those 8 bits group into letter by looking up ASCII code table.
f) Using result from the previous step, do a letter shifting to get their original letter form.
g) The secret message has been decoded.

7. EXPERIMENT

We encoded 96 bits message in length that consists of 13 characters in total. The message is 'steganography'. Then processed by shift cipher becomes *wxikerskvetlc.*

The Table-6 and Table-7 below showing 2 bytes sequence of character during the XOR process. We divided the word into two characters sequence to make it easier to distinguish to write it in table form because in this experiment we use 16 bit keys to done the XOR encryption. Thus, this kind of writing doesn't change the binary value we got from the XOR encryption process.

Table-6. Character binary value after XOR with Key x1.

wx	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**0**	**1**
ik	0	0	0	1	1	0	1	1	0	0	0	0	1	0	1	0
er	0	0	0	1	0	1	1	1	0	0	0	1	0	0	1	1
sk	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0
ve	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0
tl	0	0	0	0	0	1	1	0	0	0	0	0	1	1	0	1
c	0	0	0	1	0	0	0	1								

Table-6 shows result from the first XOR process with key x1. This will be use in the next XOR process with the x2 key. The result from the second XOR process is shown in the Table-7 below.

Table-7. Character binary value after XOR with Key x2.

wx	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**0**	**0**	**1**	**1**	**1**	**1**	**0**	**0**	**0**
ik	0	1	1	1	0	0	0	0	0	1	1	0	1	0	1	1
er	0	1	1	1	1	1	0	0	0	1	1	1	0	0	1	0
sk	0	1	1	0	1	0	1	0	0	1	1	0	1	0	1	1
ve	0	1	1	0	1	1	1	1	0	1	1	0	0	1	0	1
tl	0	1	1	0	1	1	0	1	0	1	1	0	1	1	0	0
c	0	1	1	1	1	0	1	0								

The last result as shown in Table-7 from the second XOR process will be the one that embedded to the cover text. In this experiment, we have 5 cover texts with different content and sizes. All of the cover text was in Microsoft Word document. All 5 cover text will be embedded with two techniques, the first will using our proposed technique, and using original UniSpaCh technique. Below is the table showing those representation schemes for each group.

Table-8. Modified representation for group (A).

Inter Word and Sentence Spacing	
Combination	**Sequence**
Normal Space	00
Six-Per-Em + Normal	01
Hair + Normal	10
Thin + Normal	11

In Table-8 shows that we slightly change the order of the combination base on visual look on how width the space occupied. Other than that, we can also change it for any other reason like because using pseudo-random number generator.

Table-9. Representation for group (B).

Inter Word and Sentence Spacing	
Combination	**Sequence**
Hair	00
Six-Per-Em	01
Punctuation	10
Thin	11

As in Table-9, we didn't change the representation scheme for Group (B), so it will remain as is. After we decided the representation scheme, then we embed the message into the cover text.

For our own propose technique we alter a little bit of the representation scheme of spacing group in Group (A), which is for inter-word spacing and inter-sentence spacing. Otherwise, the representation scheme for Group (B) which is for end of line and inter-paragraph spacing remain the same.

In Figure-2, we can the result of the stego text size compared to the cover text size. As we can see, there is no significant change in size, indicating that this technique has a very high embedding efficiency.

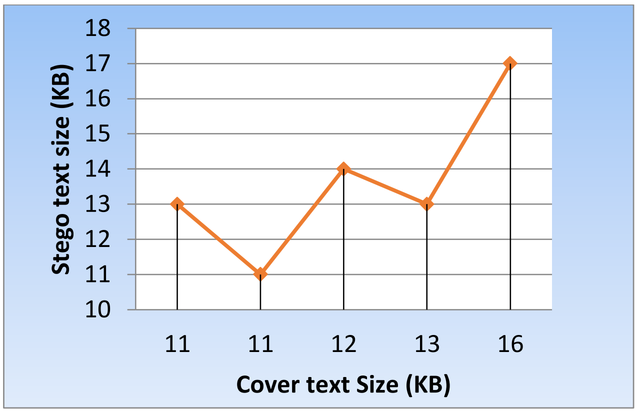

Figure-2. Simple scheme of text steganography.

Figure-3 shown our proposed technique doesn't give too much of a different result than original UniSpaCh technique. In other words our propose technique still give the high embedding capacity that the original UniSpaCh has. Even our propose technique don't give better efficiency, but we sure that it is ensure more security because of the encryption process done before embedding the actual message into the cover text occurs.

For future work, we want to make more robust encryption system than what are now are presented in this paper.

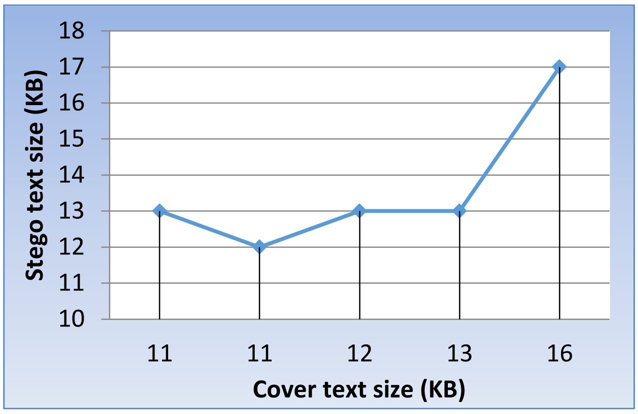

Figure-3. Result graph for encoding 96-bit message with our proposed technique.

Also, further study need to be conduct regarding the effect of cover text content to the resulting stego text. The embedding capacity still can be improved by doing some technique that can compress the secret message into smaller size.

8. CONCLUSIONS

Our proposed technique still gives the same embedding capacity that original UniSpaCh has, and also ensure more secure and robustness because the encryption process done prior on embedding message to the cover text. Both UniSpaCh and our proposed technique have a high embedding capacity, in other words it can embed a large enough message without significant size change in the stego-text file as a output. Even though there are no significant increase in term of embedding capacity, the proposed technique also doesn't decrease the embedding capacity, it can be indicated by the result that shown in the last graph. The embedding capacity could be done by compress either the secret message or the stego text. This experiment shows that steganography is a promising technique to hide a data and use a text document as a cover, because it have a low risk of exposure and a good combination with cryptography because it will increase it robustness against any attack and ensure more security

REFRENCES

[1] Por L.Y., T.F. Ang B. Delina. 2008. WhiteSteg: A New Scheme in Information Hiding Using Text Steganography. WSEAS Transaction on Computer. 7(6).

[2] Bennett K. 2004. Linguistic Steganography: Survey, Analysis, and Robustness Concerns for Hiding Information in Text. Purdue University, CERIAS Tech. Report.

[3] Topkara Mercan, Umut Topkara, Mikhail J. Attalah. Information Hiding Through Errors: A Confusing Approach. Departement of Computer Science, Purdue University.

[4] Khairullah Md. 2009. A Novel Text Steganography System Using Font Color of the Invisible Characters in Microsoft Word Documents. in Second International Conference on Computer and Electrical Engineering.

[5] Mahato Susmita., Danish Ali Khan, Dilip Kumar Yadav. 2017. A Modified approach to data hiding in Microsoft Word documents by change-tracking technique. Journal of King Saud University – Computer and Information Sciences.

[6] Brassil Jack T., Steven Low, Nicholas F. Maxemchuk, Lawrence O'Gorman. 1995. Electronic Marking and Identification Techniques to Discourage Document Copying. IEEE Journal on Selected Areas In Communication. 13(8).

[7] Desoky, Abdelrahman. 2009. Listega: list-based steganography methodology. International Journal of Information Security.(8):247-261.

[8] Han Lim Chong., Nor Muzlifah Mahyuddin. 2014. An Implementation of Caesar Cipher and XOR Encryption Technique in a Secure Wireless Communication. in 2nd International Conference on Electronic Design (ICED), August.

[9] M. Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza. 2008. A New Synonym Text Steganography. in International Conference on Intelligent Information Hiding and Multimedia Signal Processing.

[10] Por Lip Yee, KokSheik Wong, Kok Onn Chee, UniSpaCh. 2012. A text-based data hiding method using Unicode space characters. The Journal of Systems and Software. 85, pp. 1075-1082.

[11] Nitesh Rao M and Lavanya Pamulaparty. 2016. Text Steganography: Review. International Journal of Computer Science and Information Technology & Security. 6(4), July-August.