A High Capacity and Imperceptible Text Steganography Using Binary Digit Mapping on ASCII Characters

Alfin Naharuddin[1,2], Adhi Dharma Wibawa[1], Surya Sumpeno[1]
[1]Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[2]Kementerian Agama Republik Indonesia, Banda Aceh, Indonesia
alfin.nahr@gmail.com, adhiosa@te.its.ac.id, surya@te.its.ac.id

***Abstract*— Due to its light and multiplatform feature, plain text is widely used to transmit the information in news and social media online. However, it is vulnerable to attacks (e.g. unauthorized access or misuse of the information where the text is modified for particular purposes). To deal with the problem, secret text (ST) is embedded in overall plain texts appearing as a cover text (CT) so that any changes on CT can be detected. The present study proposes ST embedding method in CT by mapping ST binary digit onto binary digit of CT using ASCII characters – involving spaces, punctuation, and symbols. Prior to embedding process, the ST text was firstly encrypted with a One Time Pad (OTP) into ciphertext and each character was converted into binary number representing 7-bits long. Unlike the ST text, the CT text was immediately converted into 7 bits of binary number. The embedding process was conducted by mapping one bit of ST onto the first bit of CT character containing the same number of bit (e.g. mapping bit 1 of ST character onto the first bit 1 of CT character). Such a process was repeatedly carried out to ensure all bit of ST was completely embedded in CT. The system recorded each bit position as a stego key for any bit of ST had its position on bit of CT. The stego key served as a key to extract ST embedded in CT. The result of embedding process did not change the appearance of CT and therefore the method worked well with hidden information serving as text steganography or as watermark. In short, all CT characters can be used as a medium to hide ST where 1 character of ST required 7 characters of CT. In addition, the stego texts produced appeared identical to CT, as measured in similarity distance with Jaro-Winkler Distance of 1.**

Keywords—embedded text, text steganography, plain text, bit mapping, ASCII characters.

I. INTRODUCTION

Nowadays, more and more people around the globe, especially those with smartphones or gadgets, easily get access to information either from online news, from social media, or from other online sources. The information on those online worlds is dominantly in plain text format that is prone to be modified and redistibuted by unauthorized user. This leads to unreliable information or hoax as the original one has been irresponsibly edited for particular purposes.

Since it is quite challenging to maintain validity and integrity of information in plain text, steganography or watermark method –by embedding secret text in the cover text– is applied to protect the information.This paper proposes embedding method of secret text in the plain text using ASCII characters by mapping out its binary number. The method ensures that the appearance of the plain text with secret text embedded in it is identical to the plain text with no secret text. Unauthorized user trying to modify (i.e. to reduce or to add) the plain text can be detected. Hence, the originality and the integrity of information in the plain text with secret text embedded can be guaranteed. This method is aplicable to online news, e-commerce, social media, and other digital contents using plain text. It is even applicable to short message services (SMS).

Data embedding is normally associated with three themes: information hiding, watermarking, and steganography. These three themes seem closely related and technically similar [1], yet there are obvious differences among them. While information hiding deals with concealing secret message in another media, watermark conceals secret message in another media where the main subject is the media its self, and steganography attempts to conceal secret information in another media where the main subject is the secret information. Steganography therefore is regarded as failed when the hidden message in the media is known by an authorized user [2].

Steganography derives from the ancient Greek, meaning hidden writing –an art of communicating secret messages. As narrated by Herodutos [3], steganography was initially used by a slave whose scalp was tattooed secretly by his master named Histiaeus. Soon after the slave's hair grew back, he was sent to Ionian city Miletus to meet the ruler, Aristagoras. Then, the slave's hair was cut bald to show the hidden message.

Contemporary steganography employs file cover in the form of text, image, audio, video, or other digital media to conceal secret messages to be sent to the intended recipient [4]. In steganography, the secret message concealed is called embedded data; the key used to encode the hiding process is called stego key, which later is used to extract secret messages; and file cover with embedded secret message is called stego file.

Steganography using text as a medium to conceal secret message is called text steganography. It has various methods (i.e. modify text format, modify words, order the jumbled characters, or use context-free grammar) to produce easily readable texts [5].

Based on its function, steganography can be distinguished into robust and fragile. Robust steganography is aimed at keeping the secret information embedded in the file cover

978-1-5386-7654-7/18/$31.00 ©2018 IEEE

Authorized licensed use limited to: National Institute of Technology - Jamshedpur. Downloaded on March 09,2023 at 12:15:45 UTC from IEEE Xplore. Restrictions apply.

from perceptibility or omission, whereas fragile steganography is aimed at keeping stego file from modification –is there a slight change to it, the integrity of the stego file can be confirmed [6].

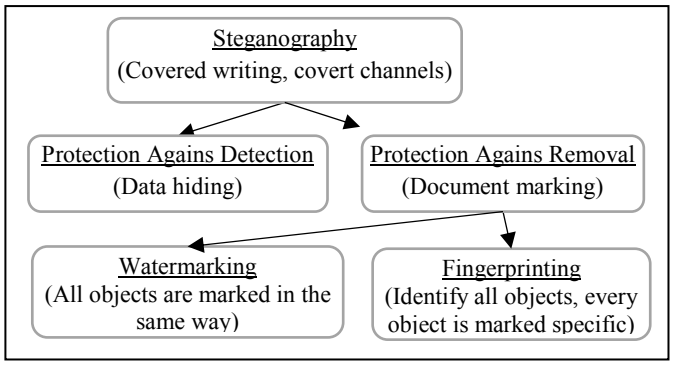

Fig. 1. Directions within steganography [7]

This paper is divided into several sections: introduction is presented in the first section; related works and theories are discussed in the second section; proposed method is explained in the third section; results and discussion are presented in the fourth section; and conclusions is provided in the last section.

II. Related Works and Theories

In the following section, we describe data embedding method as text steganography and theories related to proposed method.

A. Hiding Data in Wordlist

The Hiding Data in Wordlist method discusses how to hide ST in wordlist using dynamic CT. The initial letter and word with a particular length are used to hide secret characters, the first character is obtained by determining the decimal value of each secret character [8].

B. Using Paragraphs to Hiding Secret Text

The text steganography method [8] uses paragraphs in English text serving as CT to hide ST. This method produces a stego key that is used to embed and reextract secret messages. The stego key produced has the same appearance as CT.

The method uses binary code number of ASCII characters in which prior to ecncrypting the embedding process of secret text using One Time Pad (OTP), chiper text produced is converted to bit stream to ensure that ST is safe. Each bit of chiper text is inserted into each word of CT in sequence. Bit 0 or 1 is inserted into the first or the last letter of the chosen word –depending upon the word used. Nevertheless, it is not applicable to the words with the same first letter and last letter.

The Hiding Data in Paragraphs method [8] has been developed in the previous study [9], it works on the binary value of the characters and the value of the XOR between the first and the last words of a line in the CT paragraph, and in other previous studies [10], it works by hiding a bit of ST in each character in the words of a CT, each bit is hidden by picking a character from the CT and using the ASCII value of the character odd number or even number is generated depending on the bit to be concealed.

C. One Time Pad Encryption

One Time Pad (OTP) is a symmetric encryption algorithm, which means the same key was used for encryption and decryption process. Encryption using OTP is a common practice in cryptography technique. OTP is extremely difficult to solve since it is a single-use key and the number of keys equals the number of encoded messages [11]. As the key sequences do not show statistical trend, the attacker needs to use all possibilities to open it. OTP algorithm uses XOR or modulo (mod) with the following notation [12]:

$$C_i = (P_i + K_i) \bmod z \tag{1}$$

where C_i is ciphertext, P_i is plaintext, and K_i is the key, and z is the number of possible characters.

D. Plain Text

Plain text has no format and structure information such as font size and style, color, and layout. It is usually used in intercomputer network that has no agreement to exchange format information and text layout [13]. Plain text of ASCII characters normally has 7 to 8 bits. For instance, Notepad (Windows), edit (DOS), ed, emacs, vi, vim, Gedit or nano (Unix, GNU / Linux), SimpleText (Mac OS), or TextEdit (Mac OS X). There is a term difference between 'plain text' with spaces and 'plaintext' without spaces. The term 'plaintext' is commonly used in cryptography as a text before encryption process is conducted and as a cipher text after the encryption process has been conducted.

E. Jaro-Winkler Distance

Jaro-Winkler Distance is an algorithm to measure similarities between two strings. This algorithm is employed to detect duplicates –by comparing the similarities between the two strings. The score of 0 indicates no similary is found and the score of 1 indicates the two strings are perfectly similar. The basis of this algorithm has three parts: calculating the lenght of the string, finding the same number of characters in two strings, and finding the number of transpositions [14]. For example, to measure distance (d_j) between two strings s_1 dan s_2 [15] used the equation:

$$d_j = \frac{1}{3} \times \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) \tag{2}$$

where m is the same number of characters, $|s_1|$ is the string length 1 (cover-text), $|s_2|$ is the string length 2 (stego-text) and t is the number of transpositions. The characters on s_1 and s_2 are matched whether the two are the same and no more than:

$$\left\lfloor \frac{\max |s_1|, |s_2|}{2} \right\rfloor - 1 \tag{3}$$

To accurately calculate the similarity of the two strings, its Jaro-Winkler distance (d_w) is calculated with the following equation:

Authorized licensed use limited to: National Institute of Technology - Jamshedpur. Downloaded on March 09,2023 at 12:15:45 UTC from IEEE Xplore. Restrictions apply.

$$d_w = d_j + (lp(1 - d_j)) \quad (4)$$

where Jaro-Winkler uses prefix scale (p) with value of 0.1, and prefix length (l) is the same length of character of the string compared to reach maximum inequality of 4 characters and d_j is the result of string equality calculation s_1 and s_2.

III. Proposed Method: Bit Mapping on ASCII Characters

The proposed method provides techniques to embed secret text (ST) in the cover text (CT) by mapping binary digit (bit) of each ASCII characters of ST onto binary digit of each CT character. This bit mapping involves all binary digit of CT, including spaces, punctuation, and symbols.

A. Bit Mapping

Bit mapping is defined as mapping the binary (bit) digit value onto each ASCII character as an embedding method of text staganography to produce sequence number used as a stego key. The mapping uses 7 bits value of each character either as ST or as CT. For instance, binary value of letter Z is 1011010 used as ST and ABCDEFG as CT with their binary values of 1000001, 1000010, 1000011, 1000100, 1000101, 1000110, 1000111. Therefore, their mapping is by embedding each ST binary in sequence from left to right onto one of the same binary values in each CT character –by finding them in sequence from right to left and the sequence number is recorded as a stego key. The mapping stages are described in Fig. 2

	1	**1**	**1**	**3**	**2**	**2**	**4**	→	Stego-Key
	1	*2*	*3*	*4*	*5*	*6*	*7*		*Sequence number*
Z	1	0	1	1	0	1	0	→	Secret Text
	7	*6*	*5*	*4*	*3*	*2*	*1*		*Sequence number*
A	1	0	0	0	0	0	1		
B	1	0	0	0	0	1	0		
C	1	0	0	0	0	1	1		
D	1	0	0	0	1	0	0		Cover Text
E	1	0	0	0	1	0	1		
F	1	0	0	0	1	1	0		
G	1	0	0	0	1	1	1		
ASCII	Binary digits (bit) value								

Fig. 2. Bit mapping

B. Method

Text steganography with bit mapping method can be described as follow: As an example, a set of letters of the alphabet $A=\{A,B,C,\ldots,Y,Z\}$, a set of integers modulo 26, that is Z_{26}, secret-text $M=a_1,a_2,\ldots,a_m$ with $a_i \in A$, OTP key $K=k_1,k_2,\ldots,k_m$ with $k_i \in A$, then function of $Asc(x)$ is function to change $x \in A$ into the last 7 digits of ASCII code, and f is bijective function $f: A \rightarrow Z_{26}$ that change the letters of the alphabet into their integer representation. Encrypted text E can be obtained by:

$$E = f^{-1}([f(M) + f(K)] \bmod 26) \quad (5)$$

Subsequently, binary representation E can be obtained by:

$$E_b = Asc(e), \forall e \in E \quad (6)$$

with $|E_b| = 7m$. For example, cover-text used $C=c_1,c_2,\ldots,c_n$ with $n \geq 7m$. Binary form C can be obtained by:

$$C_b = Asc(c), \forall c \in C \quad (7)$$

Following this embedding process is conducted from binary digits E_b to binary digits C_b with rules E_b is read from left to right, C_b is read from the first character (on the first 7 digits) from right to left. As an example, $e_i \in E_b$ and $c_j \in C_b$, with $i=1,2,\ldots,7m$ and $j=1,2,\ldots,7n$. Later, the stego key is obtained from embedding process $W=w_1,w_2,\ldots,w_{7n}$ with rules $w_k=j$ if $e_i=c_j$, and as the result of the process obtained a stego-text S that appears identical to the cover-text.

The extraction process of stego text is as follow; for example, message recipient has received stego text S, stego key W and OTP key K. Thus, the first step; stego text S is converted into binary S_b with

$$S_b = Asc(s), \forall c \in S \quad (8)$$

secret text E_b can be extracted from S_b using stego key W. To illustrate, $w_k \in W$, then from each character (binary digit 7) in S_b, is extracted into E_b by taking index digit from w_k on S_b, with $k=1,2,\ldots,7n$. Afterward, E_b is converted into E with

$$E = Asc^{-1}(e), \forall e \in E_b \quad (9)$$

subsequently E is converted into M with

$$M = f^{-1}([f(E) - f(K)] \bmod 26) \quad (10)$$

Hence, the secret text that is previously embedded in the cover text is recovered.

Bit mapping method scheme on ASCII characters is shown in Fig. 5, and stages of embedding process can be seen in Fig. 3, and extracting algorithm in Fig. 4.

(1) Get the secret text (M) and encrypt the text (E);
(2) Get the encrypted text (E) and convert to its binary code equivalent (E_b) and make 7 last bits;
(3) Get the cover text (C) and convert to its binary code equivalent (C_b) and make 7 last bits;
(4) Read sequential and repetitive bits (E_b) left to right;
(5) Read the bits (C_b) sequentially per letter right to left and write in the embedded text (S);
(6) If bit (E_b) = 0 finds in bit (C_b) = 0;
(7) Else If bit (E_b) = 1 finds in bit (C_b) = 1;
(8) Then write the sequence number in the stego key (W);
(9) Iterate steps 4-8 till the last of bits (E_b);
(10) Send stego text (S) and stego key (W) to receiver.

Fig. 3. Pseudocode for embedding algorithm

Authorized licensed use limited to: National Institute of Technology - Jamshedpur. Downloaded on March 09,2023 at 12:15:45 UTC from IEEE Xplore. Restrictions apply.

(1) Get the stego text (S) and convert to its binary code equivalent (S_b);
(2) Get the stego key (W) and read sequentially;
(3) Read consecutive bits (S_b) per letter;
(4) If key (W) = 1 then take the first bit (S_b);
(5) If key (W) = 2 then take the second bit (S_b);
(6) If key (W) = 3 then take the third bit (S_b);
(7) If key (W) = 4 then take the fourth bit (S_b);
(8) If key (W) = 5 then take the fifth bit (S_b);
(9) If key (W) = 6 then take the sixth bit (S_b);
(10) Else If key (W) = 7 then take the seventh bit (S_b);
(11) Then write in the file (E_b);
(12) Repeat steps 3-11 till the last of bits (S_b);
(13) Convert every 7 bits the file (E_b) to its ASCII binary code equivalent and write to encrypted text (E);
(14) Obtained encrypted text (E);
(15) Decrypt (E) to secret message (M).

Fig. 4. Pseudocode for the extracting algorithm

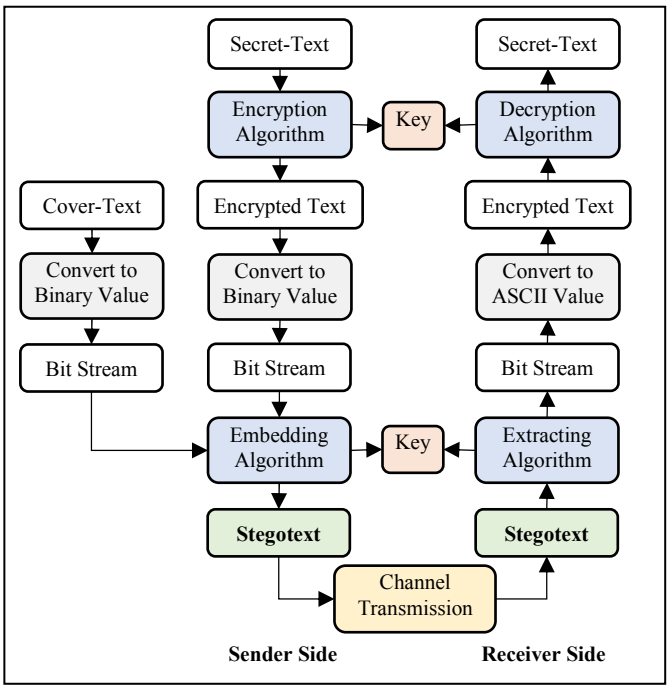

Fig. 5. Proposed method prosessing

C. Example

We applied the bit mapping method to the content of Presidential Decree of Republic of Indonesia No 7 of 2018 on the cost of Hajj operations in 1439H / 2018M (Presidential Decree of Republic of Indonesia No 7/2018 on Hajj pilgrimage cost 1439H/2018M) as CT, and used the vision of the Ministry of Religious Affairs as ST. To proceed, the ST text was firstly encrypted with an OPT scheme and then converted into bit stream. Unlike the ST text, the CT text was immediately converted into bit stream. Following this, the embedding process was conducted to generate a stego text along with its stego key. Below is the result of the process:

- Secret Text :VisiKementerianAgamaRepublikIndonesiaTerwujudnyaMasyarakatIndonesiaYangTaatBeragamaRukunCerdasDanSejahteraLahirBatinDalamRangkaMewujudkanIndonesiaYangBerdaulatMandiriDanBerkepribadianBerlandaskanGotongRoyong
- OTP Key : njvfuihaurfjannfvueheqfjnvifhvaerjfrfurufgrehfiaghufgjfnghpigjrezmnxbcvvhfggdfjdkslapoqiqwuieurytttgfghfdjdkzmxnxbcvvgfhdjskslapqoqiwiuewuerutyttqpplkjdhdiedutethvhvbcgcgpoiiuyuttwqwadsfdgfhjklklmnbvcxcvjnjd
- Cover text Fig. 7, and stego text Fig. 8, and Stego key Fig. 6

12111251121113122262211265111112241122121321324111221415142612112111141152426224212112111412261111122111223232441142111212126111143
12131111531151125113153421321211313261231231112112625321435234215133314111131111111213121111211532232223253554312311111113265232113
12211226151325112212151342142121132316513222124331211111251112212212111341242121231212131211641521516423113121231261113111222213514
41231212111316123123264111212141121321621331112112135533124112223111111353311122611311111114235143113222121112632112221331211221321
21121213211534123122231316111222121223111212262114121421131554432
41121131111223115111241222312621311121133215544314111223111111311
11331211111251211512411311341242221232326352213312211226113121112
52355442121111222111121212114632112613124525111621121311651111126
41111311111643513211111122621121132342141122232316132141212322625
21122422155341131122212111221124321131261521211122531544123121111
11121113123121112161525122322123154411421212123115132243111132262
15211112535544114111211211612111122133121321252111122111155432131
11222323162132231115123136423113121111162522223115111554413221112
13111111326421123113121231251222311531111341232221211213535412
12112121112115222531134124111121212631211222312121126212321141221
55332131122111111212211111224322211261521242321213153412322212111
12131352512331113121112222112225315443231112212121111133224112212
62124211322111554311412222321113511114121211115122111211515133

Fig. 6. Stegokey

Menetapkan besaran Biaya Penyelenggaraan lbadah Haji (BPIH) Tahun 1439H/2018M bagi Jemaah Haji sebagai berikut:

a.	Embarkasi Aceh	sebesar Rp31.090.010,00
b.	Embarkasi Medan	sebesar Rp31.840.375,00
c.	Embarkasi Batam	sebesar Rp32.456.450,00
d.	Embarkasi Padang	sebesar Rp33.068.245,00
e.	Embarkasi Palembang	sebesar Rp33.529.675,00
f.	Embarkasi Jakarta (Pondok Gede)	sebesar Rp34.532.190,00
g.	Embarkasi Jakarta (Bekasi)	sebesar Rp34.532.190,00
h.	Embarkasi Solo	sebesar Rp35.933.275,00
i.	Embarkasi Surabaya	sebesar Rp36.091.845,00
j.	Embarkasi Banjarmasin	sebesar Rp38.157.084,00
k.	Embarkasi Balikpapan	sebesar Rp38.525.445,00
l.	Embarkasi Makassar	sebesar Rp39.507.741,00
m.	Embarkasi Lombok	sebesar Rp38.798.305,00

Menetapkan besaran BPIH Tahun 1439H/2018M bagi Tim Pemandu Haji Daerah (TPHD) sebagai berikut:

a.	Embarkasi Aceh	sebesar Rp58.796.855,00
b.	Embarkasi Medan	sebesar Rp59.547.220,00
c.	Embarkasi Batam	sebesar Rp60.163.295,00
d.	Embarkasi Padang	sebesar Rp60.775.090,00
e.	Embarkasi Palembang	sebesar Rp61.236.520,00
f.	Embarkasi Jakarta (Pondok Gede)	sebesar Rp62.239.035,00
g.	Embarkasi Jakarta (Bekasi)	sebesar Rp62.239.035,00
h.	Embarkasi Solo	sebesar Rp63.640.120,00
i.	Embarkasi Surabaya	sebesar Rp63.798.690,00
j.	Embarkasi Banjarmasin	sebesar Rp65.863.929,00
k.	Embarkasi Balikpapan	sebesar Rp66.232.290,00
l.	Embarkasi Makassar	sebesar Rp67.214.586,00
m.	Embarkasi Lombok	sebesar Rp66.505.150,00

Fig. 7. Cover-text

Authorized licensed use limited to: National Institute of Technology - Jamshedpur. Downloaded on March 09,2023 at 12:15:45 UTC from IEEE Xplore. Restrictions apply.

Menetapkan besaran Biaya Penyelenggaraan Ibadah Haji (BPIH) Tahun 1439H/2018M bagi Jemaah Haji sebagai berikut:

n.	Embarkasi Aceh	sebesar Rp31.090.010,00
o.	Embarkasi Medan	sebesar Rp31.840.375,00
p.	Embarkasi Batam	sebesar Rp32.456.450,00
q.	Embarkasi Padang	sebesar Rp33.068.245,00
r.	Embarkasi Palembang	sebesar Rp33.529.675,00
s.	Embarkasi Jakarta (Pondok Gede)	sebesar Rp34.532.190,00
t.	Embarkasi Jakarta (Bekasi)	sebesar Rp34.532.190,00
u.	Embarkasi Solo	sebesar Rp35.933.275,00
v.	Embarkasi Surabaya	sebesar Rp36.091.845,00
w.	Embarkasi Banjarmasin	sebesar Rp38.157.084,00
x.	Embarkasi Balikpapan	sebesar Rp38.525.445,00
y.	Embarkasi Makassar	sebesar Rp39.507.741,00
z.	Embarkasi Lombok	sebesar Rp38.798.305,00

Menetapkan besaran BPIH Tahun 1439H/2018M bagi Tim Pemandu Haji Daerah (TPHD) sebagai berikut:

n.	Embarkasi Aceh	sebesar Rp58.796.855,00
o.	Embarkasi Medan	sebesar Rp59.547.220,00
p.	Embarkasi Batam	sebesar Rp60.163.295,00
q.	Embarkasi Padang	sebesar Rp60.775.090,00
r.	Embarkasi Palembang	sebesar Rp61.236.520,00
s.	Embarkasi Jakarta (Pondok Gede)	sebesar Rp62.239.035,00
t.	Embarkasi Jakarta (Bekasi)	sebesar Rp62.239.035,00
u.	Embarkasi Solo	sebesar Rp63.640.120,00
v.	Embarkasi Surabaya	sebesar Rp63.798.690,00
w.	Embarkasi Banjarmasin	sebesar Rp65.863.929,00
x.	Embarkasi Balikpapan	sebesar Rp66.232.290,00
y.	Embarkasi Makassar	sebesar Rp67.214.586,00
z.	Embarkasi Lombok	sebesar Rp66.505.150,00

Fig. 8. Stegotext

IV. Results and Discussion

The following is an example of the results of bit mapping method implementation and the results of the study along with the discussion.

A. Experimental

The experiment was conducted on plain text based information media to see if there was any change in the plain text after embedded ST and measured how much CT capacity can accommodate ST. Then, ST was extracted from stego text generated using stego key that had been made. Testing was conducted using article texts of several online mass media, e-commerce sites (used in the product description section) and some popular social media and SMS. The results of the experiment are presented in TABLE I.

TABLE I. Comparison Cover Text Capacity

A piece of plain text	Size as cover text		Secret text in bytes	Capacity %	Jaro score
	Bytes	Max / avg			
Detik.com	2241	Average	320	14.279	1
Kompas.com	2428	Average	346	14.250	1
Bukalapak	1094	Average	156	14.265	1
Tokopedia	1101	Average	157	14.260	1
SMS	160	Maximum	22	13.750	1
Twitter	280	Maximum	40	14.286	1
Facebook	63206	Maximum	9029	14.285	1
WhatsApp	65536	Maximum	9362	14.285	1

To generate average size of CT bytes of online news and e-commerce, we took 100 articles as samples of each field randomly using different ASCII characters and different paragraph lengths. The CT of social media and SMS have a maximum limit by default, SMS is used to indicate the methods can be done even on media with limited characters such as SMS.

B. Capacity

Capacity ratio calculation indicates that CT can accommodate ST, using equation (11) it is known that the capacity ratio of the proposed method is 1 to 7, where 7 bits of CT could store 1 bit of ST and it shows that 1 ASCII Character holds 1 bit of ST or the embedding capacity is 14.2%.

$$Capacity\ ratio = \frac{bits\ of\ secret\ text}{bits\ of\ cover\ text} \times 100\% \quad (11)$$

C. Similarity measure

After the embedding process, the stego-text generated was compared with CT using Jaro-Winkler Distance to measure the similarity of both, with the condition that score 0 indicates no similarity, whereas score 1 means exactly the same. The similarity of the two comparable strings used equation (2) and the result was 1, which means both are exactly the same.

D. Discussion

The main purpose of steganography is to hide confidential information into a certain medium to be conveyed to the intended recipient. The more disguised the concealment the better the method is. Therefore, the most ideal method is the method that produces the stego-file that is not much different from its cover file.

This bit mapping method generates a stego-text that is exactly the same as its CT. By utilizing the binary value of ASCII characters, an imperceptible stego-text can be generated so that it looks like a normal text and the irregular pattern produced cannot be detected by human vision. This method uses a double key to protect ST, the OTP Key for ST encryption before it is embedded and stego-key to embed ST in CT. Thus, this method is quite safe because the encryption used is symmetric with a statistically unpredictable key.

This method produces a larger embedding capacity with a 7-bit CT capability to hold 1 bit ST meaning that each CT character can hide ST and its capacity ratio is 14.2%. The bit mapping works on plain text that can be used in digital media in general so that this text steganography method is possibly applied to various digital needs, either to convey secret messages or to maintain the integrity of information text.

ST hidden in the CT may be defective and disappear if printed and scanned with OCR (*Optical Character Recognition*) due to character recognition limitations especially on *non printable* characters such as Tab (*Horizontal Tab*), CR (*Carriage Return*) and LF (*Line Feed*), but ST hidden in the CT will not be defective and disappear despite being copied-pasted on various digital platforms –as long as

Authorized licensed use limited to: National Institute of Technology - Jamshedpur. Downloaded on March 09,2023 at 12:15:45 UTC from IEEE Xplore. Restrictions apply.

the CT sequence is not changed and there is no reduction or addition to the CT characters. Hence, this method can be applied to maintain the integrity of a plain text because each CT character is inserted by the ST bit so that if there is a sequence change or addition or subtraction of CT characters can be detected. This method enables the plain text to be fragile so that the validity and the integrity of plain text can be guaranteed.

V. CONCLUSIONS

This bit mapping method successfully conceals secret-text (ST) into cover-text (CT) by converting characters into bit stream where 1 bit of ST can be hidden in 7 bits of CT, and its capacity ratio is 14.2 %. This means each ASCII character can be used to hide ST bits. Qualitatively, the stego-text generated from the bit mapping method has been computed by the similarity measure compared to CT and shows a score of 1 on the Jaro-Winkler Distance Measurement. This means that the stego-text is exactly the same as CT prior to embedded ST.

Regarding the test, in general the plain text we used comes from online news, social media and e-commerce, and we specifically used plain text that comes from government regulations, that is the Minister of Religious Affairs Decree. The test results show no difference, both can optimally be used as a cover-text by using bit mapping method. Furthermore, this study may be extended to some character encoding types other than ASCII.

ACKNOWLEDGEMENT

This study and publication were supported by the Ministry of Religious Affairs of Republic of Indonesia and the Ministry of Communication and Informatics of Republic of Indonesia.

REFERENCES

[1] T. Cox, I., Miller, M., Bloom, J., Fridrich, J., & Kalker, Digital Watermarking and Steganography. Amsterdam/Boston.: Morgan Kaufmann Publishers, 2008.

[2] J. Fridrich, Steganography in Digital Media. New York, NY: Cambridge University Press, 2009.

[3] A. De Sélincourt and J. Marincola, The histories. 1996.

[4] P. Johri, A. Mishra, and S. Das, "Survey on steganography methods (text, image, audio, video, protocol and network steganography)," 2016 3rd Int. Conf. Comput. Sustain. Glob. Dev., pp. 2906–2909, 2016.

[5] S. Sharma, A. Gupta, M. C. Trivedi, and V. K. Yadav, "Analysis of different text steganography techniques: A survey," Proc. - 2016 2nd Int. Conf. Comput. Intell. Commun. Technol. CICT 2016, pp. 130–133, 2016.

[6] M. Umamaheswari, "Analysis of Different Steganographic Algorithms for Secured Data Hiding," J. Comput. Sci., vol. 10, no. 8, pp. 154–160, 2010.

[7] R. Popa, "An analysis of steganographic techniques," Politeh. Univ. Timisoara, Fac. Autom. Comput. Dep. Comput. Sci. Softw. Eng., p. 65, 1998.

[8] M. Agarwal, "Text Steganographic Approaches: A Comparison," Int. J. Netw. Secur. Its Appl., vol. 5, no. 1, pp. 91–106, 2013.

[9] T. Acharjee, A. Konwar, R. K. Ram, R. Sharma, and D. Goswami, "XORSTEG: A New Model of Text Steganography," 2016.

[10] Ka. Kumar ProfSuresh Pabboju, "A Comparative Result Analysis of Text Based Steganographic Approaches," IOSR J. Comput. Eng. Ver. VII, vol. 17, no. 3, pp. 2278–661, 2015.

[11] R. Shukla, H. O. Prakash, R. P. Bhushan, S. Venkataraman, and G. Varadan, "Sampurna Suraksha : Unconditionally Secure And Authenticated One Time Pad Cryptosystem," 2013.

[12] C. A. S. Wellia Shinta Sari, Eko Hari Rachmawanto, De Rosal Ignatius Moses Setiadi, "A Good Performance OTP Encryption Image based on DCT-DWT Steganography A Good Performance OTP Encryption Image based on," Telkomnika, vol. 15, no. January 2018, pp. 1987–1995, 2017.

[13] P. Christensson, "Plain Text Definition," TechTerms., 2010. [Online]. Available: https://techterms.com/definition/plaintext. [Accessed: 14-May-2017].

[14] A. Kurniawati, "Implementasi Algoritma Jaro-Winkler Distance untuk Membandingkan Kesamaan Dokumen Berbahasa Indonesia," Proceeding, Semin. Ilm. Nas. Komput. dan Sist. Intelijen KOMMIT 2008, Depok, Indones., 2010.

[15] B. Leonardo and S. Hansun, "Text documents plagiarism detection using Rabin-Karp and Jaro-Winkler distance algorithms," Indones. J. Electr. Eng. Comput. Sci., vol. 5, no. 2, pp. 462–471, 2017.

Authorized licensed use limited to: National Institute of Technology - Jamshedpur. Downloaded on March 09,2023 at 12:15:45 UTC from IEEE Xplore. Restrictions apply.