# Text Steganography with Multi level Shielding

**Sharon Rose Govada[1], Bonu Satish Kumar[2] , Manjula Devarakonda[3] and Meka James Stephen[4]**

**[1]Dept. of Computer Science and Engineering**
**KIET, Kakinada, INDIA**

**[2]Dept. of Computer Science And Engineering**
**Raghu Engineering College, Visakhapatnam, INDIA**

**[3]Dept. of Computer Science And Engineering**
**KIET, Kakinada, INDIA**

**[4]Dept. of Information and Technology**
**ANITS, Visakhapatnam, INDIA**

**Abstract**

Steganography it is a form of security through obscurity. It is the art and science of writing hidden messages in such a way that no one, except sender and intended recipient can understand the hidden message,. The purpose of steganography is covert communication-to hide the existence of a message from a third party. Compared with study on text-steganography, research on text-steganalysis is in its infancy.

In this paper, we present a method that is capable of performing text Steganography that is more reliable and secure when compared to the existing algorithms. Our method is a combination of Word shifting, Text Steganography and Synonym Text Steganography. So we called this as "Three Phase Shielding Text Steganography" This method overcomes various limitations faced by the existing Steganographic algorithms. The experimental results are very encouraging when compared to the already existing algorithms. Our method also helps in finding out the embedding rate of a secret message in a text document.

***Keywords:*** *steganography, encryption, decryption, message.*

## 1. Introduction

Text-steganography plays an increasingly significant role in covert-communication on Internet. The word steganography is derived from Greek language which literally means "concealed writing" This paper's focus is on a relatively new field of Study in Information Technology known as Steganography. Secure and secret communication methods are needed for transmitting messages over the Internet. So security plays a major role in internet to conceive it various methods including cryptography, steganography, coding, etc. are used for establishing hidden communication. The objective of steganography is to hide a secrete message within a cover-media in such a way that others cannot discern the presence of the hidden message. Technically in simple words "steganography means hiding one piece of data within another". This is the major distinction between steganography and other methods of hidden exchange of information. For example, in cryptography method, people become aware of the existence of information by observing coded information, although they are unable to comprehend the information. Most steganography jobs have been performed on images, video clips, text, music and sound [1]. But text steganography is the most difficult kind of

steganography; this is due to the lack of redundant Information in a text file, whereas lot of redundancy is present in a picture or a sound file, so these are used in steganography [2]. The structure of text documents is very identical compared to other types of documents such as in picture. Therefore, in such documents, we can hide information by making changes in the structure of the document without making a notable change in the concerned output. In this paper a new method for text steganography is presented. Where we have used US – UK Dictionary as our database in order to find the secret message. Here we have taken a UK based text document and compared the words with the database. With the presence of the 'familiar word' we find the encrypted text. In the final section the conclusion will be made after investigating and studying some advantages of this method.

## 2. Available Methods in the Literature

As stated in the previous section, that few works have been done on hiding information in texts. Here, we will come across the works done on the text steganography.

### 2.1. The line shifting

The line shifting method [4] is a method of altering a document by vertically shifting the locations of text files to some degree (for example, each line is shifted 1/300 inch up or down) to encode the document uniquely. This method is suitable for printed texts. However, in this method, the distances can be observed by using special instruments of distance assessment and necessary changes can be introduced to destroy the hidden information. Also if the text is retyped or if character recognition programs (OCR) are used, it might destroy the hidden information.

### 2.2. Word shifting

Word Shifting method [5] is a method of altering a document by horizontally shifting the locations of words within text lines to encode the document uniquely. This method is identified less, because change of distance between words to fill a line is quite common. But if somebody was aware of the algorithm of distances, they can compare the present text with the algorithm and extract the hidden information by using the difference. Although this method is very time consuming, there is a high probability of finding information hidden in the text. The same as in the method described under 2.1, retyping of the text or using OCR programs destroys the hidden information.

### 2.3. Syntactic method

Steganography is based on the fact that a given sentence may be represented into various syntactic structures without any or any essential change in meaning [6]. The typical example is, By placing some punctuation marks such as full stop (.) and comma (,) in proper places, one can hide information in a text file. This method requires identifying proper places for putting punctuation marks. The amount of information to hide in this method is trivial.

### 2.4. Semantic method

The Semantic method [3] uses the synonym of certain words thereby hiding information in the text. A major advantage of this method is the protection of information in case of retyping or using OCR programs (contrary to methods stated under 2.1 and (2.2)) may alter the meaning of the text.

### 2.5. Feature Coding:

This is a coding method that is applied either to a format file or to a bitmap image of a document. The image is examined for chosen text features, and those features are altered, or not altered, depending on the codeword. Decoding requires the original image, or more specifically, a specification of the change in pixels at a feature [7]. There are many possible choices of text features; here, we choose to alter upward, vertical end lines - that is the tops of letters, b, d, h, etc. These end lines are altered by extending or

shortening their lengths by one (or more) pixels, otherwise not changing the endline feature.

### 2.6. Text Abbreviations:

Abbreviations and spaces steganography can hide very little information in the text [8]. In this method, we take the help of abbreviations to encrypt the message. Using this method, very little information can be hidden in the text. For example, only a few bits of information can be hidden in a file of several kilo bytes. There is also other techniques which are similar to this technique such as open spaces.

## 3. Three Phase Shielding Text Steganography

All the existing methods so far, have their own limitations. None of the method proved efficient for hiding text in text document. To overcome various disadvantages of existing systems and to provide an efficient method of data hiding we hereby introduce our technique "Text Steganography with Multi level Shielding". Our method is a mixture of Synonym Text Steganography and word shifting method. Where it's capable to provide an efficiency of 95% when used for data encryption and delivery. Our method comprises of two stages. The first stage is the encryption phase, where we try to hide the secret message in the text document. The format can be in doc format, docx format and pdf format. The second stage is the decryption, where we extract the secret message from the encrypted text document.

Here we use two tables in the back-end. That is the US – UK English dictionary as database whose attributes are namely US and UK. The second table which we used is ASCII table which comprises of two attributes. Firstly the ASCII character and Second attribute is their ASCII value.

In the encryption mode, a UK based text document and secret message are given as the input. The algorithm scans each and every word in the text document. The encryption code has three modules. That is each input word will go through three levels during the process of encryption.

1. Database searching.
2. Decimal to Binary conversion.
3. Space insertion.

Initially, the encryption algorithm checks the US - UK database for the words in the text file. If the database has come across any entry, then it reads the first character of the secret message. Then the character is searched in the ASCII table for its value (decimal). The obtained decimal value is converted into binary form. Suppose if the first character of the message is A. Then its ASCII value is 97. Now this value 97 is converted into binary form as 1100001. From the obtained binary form the spaces are inserted as detailed below.

If the value of a bit in the binary value is 1, then an additional space is inserted at its position. Or if the value is 0, then no space is inserted. The additional spaces are inserted following the binary array pointer to the position of the next space. In this way, each and every character of the secret message is read and is inserted in the form of spaces in the text document. The output of encryption phase is a text document that contains a text document (normal) along with secret message. Figure 1 shown below shows the algorithm for hiding secret message in Text document

```
Start Encryption(d[],ref[],msg)
If(d[1]==ref[j])
1=i+1;
S=msg.charAt(i);
If(s==ASCii[z])
Dec=ASCiival[z];
While(dec!=0)
{
Rem[k++]=dec/2;
}
For(x=0 to k)
{
If(rem[x]==1)
Insert space at 1;
1++;
```

```
X++;
}
End encryption
```

Figure 1: Algorithm for Encryption

d[] array represents the string array that is obtained after reading all the words from the file. Ref [] is a string array that is used as a reference for encryption. Msg represents the actual message. The input for decryption is the output of encryption module i.e. the encrypted text document. The decryption module has three sub modules namely

1. Reading the file and searching the database.
2. Search for dual spaces in the text file.
3. Calculating binary array and converting into decimal.
4. Retrieving ASCII character from ASCII table.
5. Grouping the character and displaying as secret message.

Initially, the decryption algorithm reads the encrypted text file word by word and checks the US – UK database. If the database comes across any such value, then it looks after the next available seven spaces and are scanned. If there exists two spaces consecutively then it is read as one. If there exists single space, then it is read as zero. A binary array is constructed taking the occurrence of the spaces into consideration. Suppose if we have binary array as 1100001. Then it represents that there are two spaces occurring consecutively in first, second and seventh position. The constructed binary array is converted into decimal form. the binary array 1100001 after conversion into decimal form is 97. The value 97 is searched against the ASCII table and character "A" is retrieved. Similarly each and every sentence is searched and the secret message is retrieved and revealed, Figure 2 represents an algorithm for retrieving hidden data in the text document.

```
Start Decryption(d[])
While(end of file)
{
If(d[i]==ref[j])
1=i+1;
For(x=0 to 7)
if(d[i].charAt(1)==")
{
If(d[i].chart(i+1)==")
{
Bin[k]=1;
}
Else
{
Bin[k]=0;
}
X++;
K++;
}
For(x=0 to 7)
{
Dec=dec+rem[i]*pow(x,2);
}
If(dec==ASCIIval[z])
{
D=d+ASCII[z];
}
}
End decryption
```

Figure 2: Algorithm for Data Extraction.

d[] array represents the string array that is obtained after reading all the words from the file. Ref [] is a string array that is used as a reference for encryption. Bin [] array represents the binary form of ASCII character.

## 4. Experimentation Results

This section presents the results of the experiments conducted to study the performance of the proposed method. All experiments are performed on a 3.0 GHz Pentium IV PC machine with 512 MB main memory and 40 GB hard disk, running Windows XP Professional. This algorithm is implemented by using a simple java code, which is used for both data hiding and data extracting. Both process uses US - UK table from Oracle-XE, the data set consist of 2000 words (as long as the language grows the data set keeps on increasing so its not limited to just 2000 words) , and java swings is used

for user interface. Whenever embedding rate is greater than 5% then more than 98% stego-text document can be distinguished out. Our work will contribute to both text-steganalysis and text-steganography.

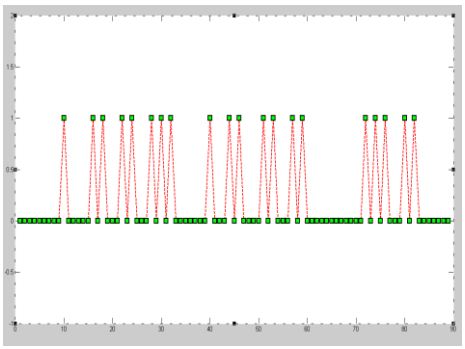

Figure 3. Attack on a stego-text file to find the revenant points and to know in which space the data is hidden

Figure 3 shows the postion of data in a text document. This figure shows spacing number along x-axis, and indicator if there is any variation in space length. This graph can be used to attack a stego-text files which used word- shifting as the data hiding technique.

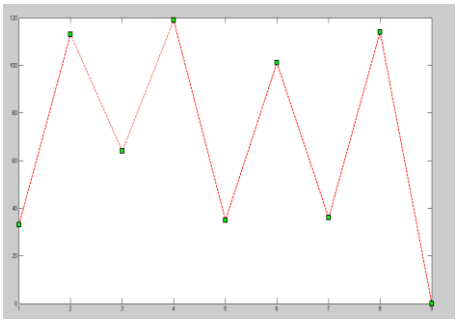

Figure 4: data hidden at revenant points

Figure 4 shows the data which is hidden at revenant points by converting the hidden data into ASCII value. In the data extraction algorithm the program searches for the revenant points which are the word which occur again after a few words. This point is found from the figure 3, in which a group of variations show that there is a point which occurs again and again in the passage or the text file. Using this revenant points, the data extraction algorithms extracts the hidden letters.

## 4. Conclusion

The intent of this paper was to cover the most common methods of Text-Steganogaphy which illustrates a bright prospect because of its wide transmission on today's internet. Therefore, corresponding steganalysis is also required to resist the illegal covert communication. Although our idea is a mixture of two existing algorithms with few changes, the results prove that it is more efficient when compared with the results of the existing algorithms.

## References

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

[2]. Lingjun Li, Liusheng Huang, Xinxin Zhao, Wei Yang, Zhili Chen," A Statistical Attack on a Kind of Word-Shift Text-Steganography", International Conference on Intelligent Information Hiding and Multimedia Signal Processing.

[3] PDFlib GmbH M¨ unchen, German, PDFlib Reference
Manual—Text Extraction Toolkit, 2.2 ed.

[4] M.Wu and B. Liu, "Data hiding in binary image for au-
Thentication and annotation," IEEE Transaction on Multimedia, vol. 4, pp. 528–538, August 2004.

[5] J. Brassil, S. H. Low, N. F. Maxemchuk, and L. O'Gorman, "Electronic marking and identification techniques to discourage document copying," IEEE Journal on Selected Areas in Communications, vol. 13, no. 8, pp. 1495–1504, 1995.

[6] J. T. Brassil, S. Low, and N. F. Maxemchuk,"Copy-right protection for electronic distribution of text docu-ments," Proceedings of the IEEE (USA), vol. 87, no. 7,pp. 1181–1196, 1999.

[7] J.T. Brassil, S. Low, N.F. Maxemchuk, and L. O'Gorman, "Electronic Marking and Identification Techniques to Discourage
Document Copying", IEEE Journal on Selected Areas in Communications, vol. 13, Issue. 8, October 1995, pp. 1495-1504

[8] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data
hiding", IBM Systems Journal, Vol. 35, No 4, pp. 313-336, 1996.