ECR(Encryption with Cover Text and Reordering) based Text Steganography

Sahil Kataria
Department of Computer Engineering
Govt. Engineering College Bikaner
Bikaner, India
sahilisforyou@gmail.com

Kavita Singh
Department of Computer Engineering
Govt. Engineering College Bikaner
Bikaner, India
singh.kavi855@gmail.com

Tarun Kumar
Department of Computer Engineering
Govt. Engineering College Bikaner
Bikaner, India
ertarunkumar@yahoo.co.in

Maninder Singh Nehra
Department of Computer Engineering
Govt. Engineering College Bikaner
Bikaner, India
maninder4unehra@yahoo.com

***Abstract*—This paper presents ECR(Encryption with Cover Text and Reordering) based text steganography approach which works on simple encryption technique using ExOR Operation of two characters and reorder them which would be more secured and hard to fetch original message from enciphered text. Our encrypted text is reordered using Eight bit random key for hiding our data in a more secure way. Our Eight bit random key will contain four number of zero and four number of one where one bit describes our encrypted text and zero bit describes cover text. We are also merging our random key with our enciphered text at the last. We are also presenting comparison of our proposed approach with some of the previous popular text steganographic approaches with load time and also the data which will be required to be enciphered using n bit cover text. At the last we are showing that how our approaches are performing best in the existing approaches. As our approach generates 2n bytes where n are no. of bytes in our plain text and n bytes for cover text and reordering using random key.**

***Keywords*—Information,Hiding,Reordering,key-based steganography,Steganography,Cryptography,Encryption,Text Steganography**

I. INTRODUCTION

Steganography means conceal communication. The main goal of steganography is to transmit a message in such a way i.e. text, image, audio or video over a communication channel in order to effectively conceal the existence of the message. Steganography is a technique of hiding our original text in such a way that no one else except sender and intended recipient can understand it just by simply looking to our enciphered text. Basically there are two ways of encryption for our plain text for secure data transfer over the Internet which is Cryptography and Steganography. Cryptography technique is used to protect the contents of message using public/private key and Steganography can be used to hide the existence of our original message using cover text. Because of some limitation in cryptography as the third party is always aware of the communication, steganography is used more for concealing the original message. Steganography gained importance after the advent of 9/11 because the US and the British government banned the use of cryptography as third party is still aware of the communication at present.

Steganography overcomes this limitation by hiding our secret message in an innocent looking object called cover media which can be any text,image,video. Most steganographic algorithm methods use images, audios, videos and text files as the cover medium and rely on the changes in the features and structure of the target medium in such a manner that is not identifiable by human eyes. However, text medium is relatively difficult as compared to other target media because of lack of available redundant information in text data. In this paper, we are presenting an overview of Text steganography and various existing text-based steganography methods. The main problem with many existing methods of Text Steganography is that it uses large cover text for hiding our secret message and also it takes too much time in encryption and decryption. We have introduced new approaches for text Steganography which would be easy for encryption and decryption but hard to fetch the original message from encrypted text.

This paper presents ECR(Encryption with Cover Text and Reordering) based text steganography approach which works on simple encryption technique using Ex-OR Operation of two characters and reorder them which would be more secured and hard to fetch original message from enciphered text. Our encrypted text is reordered using 8-bit random key for hiding our data in a more secure way. Our 8-bit random key will contains four 1's and four 0's where 1's bit describes our encrypted text and 0's bit describes cover text. We are also merging our random key with our enciphered text at the last. We are also presenting comparison of our proposed approach with some of the previous popular text steganographic approaches with load time and also the data which will be required to be enciphered using n-bit cover text. At the last we are showing that how our approaches are performing best in the existing approaches .As our approach generates 2n+1 bytes where n are no. of bytes in our plain text and n bytes for cover text and 1 byte holding random key for reordering.

We are presenting presenting here some existing text steganography technique with their result of time execution for encryption and decryption and how many bytes of data we can send with cover text.Next we will describe our proposed approaches with implementation algorithm for encryption and decryption. We are also describing our technique using an example so that it would be easy to understand our proposed approach. In the next section we are showing the results of our proposed methods and comparing with the existing methods. As every coin has two facets so this method also has some merits and demerits which we will try to remove in future.

II. RELATED WORK

The aim of Steganography technique is to hide some significant data in such a manner so that it can hide in some cover media . It is necessary to use some other redundant data as a cover for the existing valid data. The probable media that can be used as a cover can be text, image or a movie clip. Out of these different media files, a text file normally contains lesser storage and use less memory as compared to other media. In addition, text data can communicate more information and needs less cost for printing, too. On the other hand, the structure of image file or video file is changed as per original file after encryption, because more attributes about the data is required to be stored. This causes complication in our techniques as compared to handle a text file. Hence,text cover media is always preferable to be used in Steganography – such an approach is known as Text Steganography.

In Text steganography there are many methods are available. Some methods change format of text while some other methods change actual word to hide secret data as per cover text. In an Open Space method white space is used to hide our secret data. Some other methods are available which are using white spaces for hiding secret message are Inter-Sentence space method and End-of-line space method and Inter-word space method. These methods are not used as much as data would be lost if somehow format of file is changed. Then other method is Semantic method which is used to hide secret message by changing actual words. In this method Synonyms are used to hide secret data such as 0 and 1.Where as Syntactic method uses punctuations to hide secret data like commas(,) and full stop(.). This method is better than previous one but this would change meaning of our original text when we have more punctuations in our text file. In an acronyms and semantic method, meaning of information can be changed because these methods uses actual word replacement or punctuation method to hide secret data. After this technique a new terminology comes which uses characteristics of that particular language to hide data such as in Persian/Arabic Text steganography and Hindi Text steganography. So we need a method which will encipher our secret data without changing meaning of text and also data would not be lost if format of file is changed. So, considering some problems like format changing,changing meaning of secret data, etc. in existing text steganography methods next CASE approaches comes for text steganography. This method uses grouping of letters as per alphabet shapes. Letters which have only one vertical line are grouped into one and letters which contains circle or curve are grouped into others. This method is better as compared to previous methods as data would not be lost in case of format changing or meaning of data is also not changed but all these existing methods and CASE approaches hide less bytes of data and take too much time to encrypt and decrypt the data. So this would be time overhead for the encryption of our text.

So we have Proposed a new approach which is far better than other existing approaches. In the next section we will describe our approach with implementation details that how our method works. We will also show that our proposed approaches take very less time overhead and memory overhead as compared to existing approaches .Also we can hide more number of bytes using proposed approach. Required cover text size is also very small in proposed approach which will be equal to our original text.

III. THE PROPOSED APPROACH

This model(ECR) is used to encrypt our plain text using simple encryption techniques by Ex-or operation of each byte of plain text with cover text.As Ex-Or Operation is very fast and easy process for encryption and decryptoin so this approach taking less time overhead. This Ex-or Operation between plain text and cover text would generate an enciphered text which would be merge with same cover text.The merging of these two strings of enciphered text and cover text using 8-bit random key reordering method. This steganography approach is easy to encrypt our plain text as only one Ex-or operation is performed so it's very fast as compared to other previous techniques making our techniques more secure and hide our original text in cover text in a better way .We are reordering whole text using 8-bit random key. This would make more harder technique so it would be hard to fetch original text or also hard to judge that the text contain some secret data. Reordering is performed as per 8-bit random key which would have four 1's and four 0's on random position. So we will arrange our cipher text as per this random key. We will place the enciphered text when we will find 1's in our 8-bit key corresponding to our final enciphered text and place same cover text when 0 is found in the 8-bit key. In addition ,we are also going to send our 8-bit key in different way just by placing our random key at the last character of our cipher text which would be easy to get during decryption. At the time of decryption first we will fetch 8-bit random key from last character of enciphered text and then we will create two array strings in which one will contain cover text and the other will contain encipher text so next we will generate our original text using same Ex-Or operation b/w cover text string and encipher text string. So if we have n bytes of cover text then we can hide n bytes of plain text. Finally we need 2n+1 bytes to send which will have 8-bit random key,cover text and enciphered text.

IV. IMPLEMENTATION

For implementation of our proposed approach we have developed two algorithms for hiding and retrieving our secret message. We have implemented this techniques in PHP.

Pseudo code for message hiding

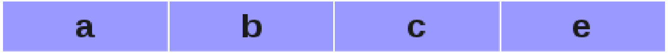
```
Procedure ECR_stegano_hide (String msg, String covertext)
begin;

For i=0 to 8
    // 8-bit random key four 1's and four 0's
        key[i]=random(0,1);
End For
For i=0 to msg.length()
        encipher[i] = msg[i]^covertext[i];
End For

SET cov to 0
SET enc to 0
For i=0 to 2*msg.length()
  IF key[i%8]==0 THEN
    IF cov < msg.length() THEN
      hidden_message[i]=covertext[cov];
      ADD 1 in cov
    ELSE
      hidden_message[i]=encipher[enc];
      ADD 1 in enc
    End IF
  ELSE
    IF enc < msg.length() THEN
      hidden_message[i]=encipher[enc];
      ADD 1 in enc
    ELSE
      hidden_message[i]=encipher[cov];
      ADD 1 in cov
    End IF
  END IF
END For
STORE key in hidden_message[i];
return hidden_message;
End Procedure
```

Pseudo Code for Message Retrieval

```
Procedure ECR_steno_unhide(string hidden_message)
begin;

SET len to hidden_message.length();
STORE hidden_message[len] to key;
SET cov to 0
SET enc to 0
For i=0 to len-1
  IF key[i%8]==0 THEN
    IF cov < len/2 THEN
      covertext[cov]= hidden_message[i];
      ADD 1 in cov
    ELSE
      encipher[enc]=hidden_message[i];
      ADD 1 in enc
    End IF
  ELSE
    IF enc < len/2 THEN
      encipher[enc]=hidden_message[i];
      ADD 1 in enc
    ELSE
      encipher[cov]=hidden_message[i];
      ADD 1 in cov
    End IF
  END IF
END For

For i=0 to len/2
        original_msg[i] = covertext[i]^encipher[i];
End For

return original_msg;
End Procedure
```

Example description for Encryption

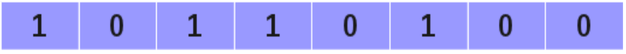
1) Let us assume that our plain text is "abce" which is shown in figure-1.



a	b	c	e

Fig. 1

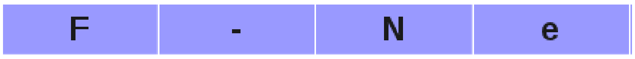
2) Generated random key is 10110100 which is shown in figure-2



1	0	1	1	0	1	0	0

Fig. 2

3) Generated Random cover text is



F	-	N	e

Fig. 3

4) Performing Ex-Or Operation b/w Cover text and original message which is shown in figure-4

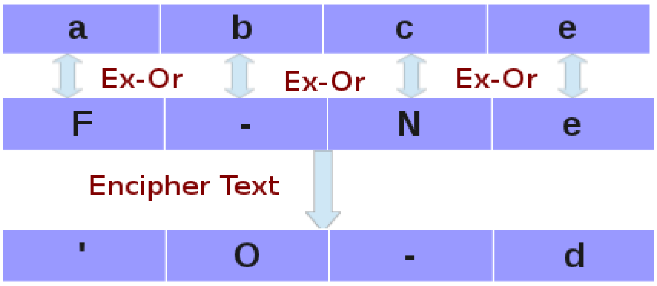
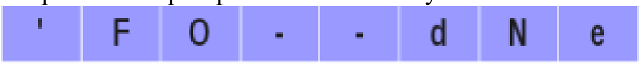


Fig. 4

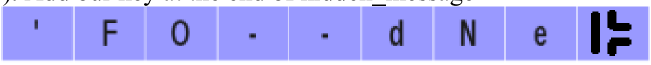
5) Merge Cover text and encipher text which was generated from previous step as per 8-bit random key.



'	F	O	-	-	d	N	e

Fig. 5

6). Add our key at the end of hidden_message



'	F	O	-	-	d	N	e	[illegible]

Fig. 6

Example description for Decryption

1) Get the hidden text message in a string which is shown in Figure-7

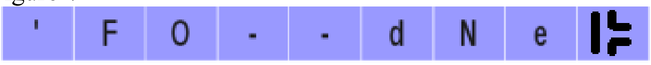


Fig. 7

2) Get the key value from last character of our hidden message which is shown in figure-8

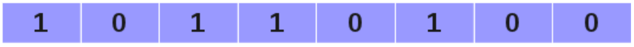


Fig. 8

3) Read the character one by one and put in two string variable where str1 contains cover text and str2 contains encipher text which is shown in figure-9.

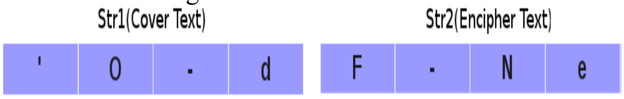


Fig. 9

4) Perform Ex-Or operation b/w cover text and encipher text which we have fetched in previous step.

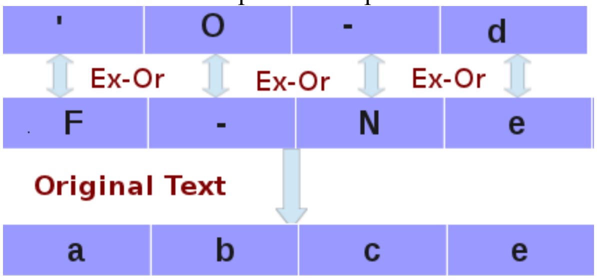


Fig. 10

V. EXPERIMENTAL RESULTS

This section shows the performance of proposed approaches and compares the results in forms of length of hiding message and time overhead with existing text steganographic approaches.

Length Capacity defines how many bytes of data will be required to be sent from sender to receiver while secret message is of n bytes.

Capacity ratio = (amount of hidden bytes) / (size of the cover text in bytes)

Assuming, one character occupies one byte in memory, we have calculated the percentage capacity which is capacity ratio multiplied by 100.If we want to hide n bytes of secret message we will require total 2n+1 bytes of cover text. So for this, our capacity ratio is defined as:-

Capacity ratio = $n/(2n+1)*100$ = 50% approx.

Below we are showing comparison of some existing approaches results with our proposed approach in form of time overhead and no. of bytes hide with fix amount of cover text for all text steganography approaches.We can see that our proposed approach is hiding all message with fix cover text and also taking too less time overhead for encryption and decryption process.

TABLE 1 *EXECUTION TIME AND COVER TEXXT SIZE REQUIRES FOR HIDING*

Text Steganography Techniques	**Message Text Size(Bytes)**	**CoverText Size(Bytes)**	**No. Of Bytes Can hide(Bytes)**	**Time Overhead(ms)**
Feature Coding	600	1980	52	20,289
Inter Word space	600	1980	45	22,604
CASE	600	1980	200	1,666
CEBTS	600	1980	424	71.012
ECR Technique	600	1980	600	15-20

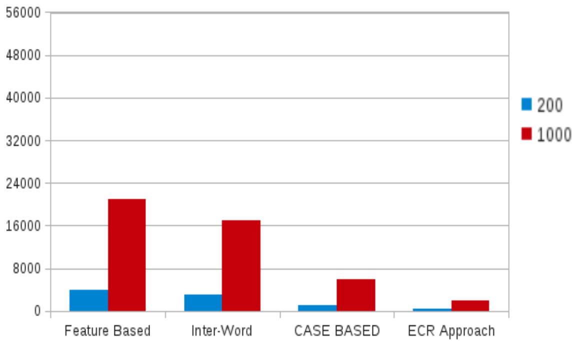


Fig. 11. Maximum Cover Text required to hide 200 bytes and 1000 bytes

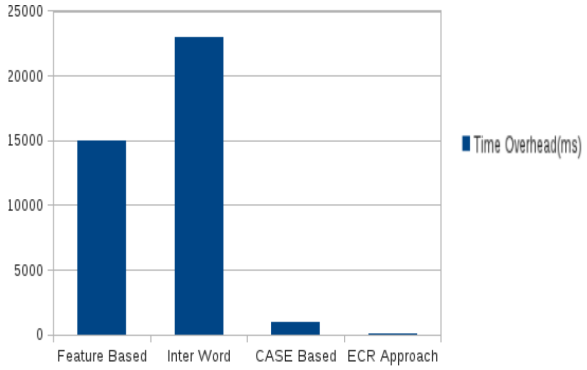


Fig. 12. Time Overhead for all approaches

VI. *CONCLUSION*

In this paper, we have proposed new approach for text-based steganography for English language text. In this approach, we are using simple Ex-Or Operation for enciphering our secret message and reordering as per 8-bit random key for more secure way. Based on our survey of the existing Text Steganography approaches, we have shown that our proposed approach can hide more number of bytes and it has also very small cover text and required very less time overhead as compared to other techniques. In addition, our proposed approach is also immune to retyping and reformatting of text.This Approach generating an non-sense message which can be used in cloud computing where we need only to store the data in large so this techniques will take less time to encrypt the data and also take too much less memory to store the data on the servers.

REFERENCES

[1] s. Chaudhary,P. Mathur, T. Kumar "A Capital Shape Alphabet Encoding(CASE) Based Text Steganography" , R. Sharma , Conference on Advances in Communication and Control Systems 2013 (CAC2S 2013) , Iindia.

[2] M.H..S.Shahreza and M. S. Shahreza, "A new approach to Persian/Arabic language text steganography".

[3] I. Banerjee, S. Bhattacharyya, and G. Sanyal, "Novel text steganography approach through special code generation,".

[4] S. Bhattacharyya, G. Sanyal and I. Banerjee, "A novel approach of secure text based steganography model using word mapping method," International Journal of Computer and Information Engineering, pp. 96-103, 2010.

[5] William Stallings, Cryptography and Network Security: Principles and Practice 5/e., India,

[6] B. Chen and G.W. Wornell, "Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding," IEEE Trans. Information Theory, Vol. 47, No. 4, pp. 1423-1443, 2001.

[7] N. Morimoto W. Bender, D. Gruhl and A. Lu.," Techniques for data hiding", IBM Systems
Journal, 35:313–316, 1996.