An Image Steganography Technique for Files and Messages

Rutvij H. Jhaveri
SVM Institute of Technology
Old N.H.-8
Bharuch, Gujrat

Kruti J. Dangarwala
SVM Institute of Technology
Old N.H.-8
Bharuch, Gujrat

Ashish D. Patel
SVM Institute of Technology
Old N.H.-8
Bharuch, Gujrat

ABSTRACT
Digital Steganography is a process of hiding confidential digital information into a file of any kind. Digital Information is a resource which holds a lot of value in the modern world. In this paper we propose Steganography with SMS alert which is an approach to send secure digital information over a network in a secure way. The proposed algorithm secures files and messages and targets enterprise level security to hide and deploy digital information without the risk of network attacks or data theft.

General Terms
Security, Steganography, Files, Messages.

Keywords
Information security, Digital steganography, Proposed algorithm.

1. INTRODUCTION
Steganography is a technique to hide messages or files into another file so that one does not recognize major changes in the contents of the original file. In this technique messages and files can be embedded and made password protected in the first phase and being made as steganos file. Embedded file will be uploaded to a web server and link of the file and password will be sent as an SMS to the specified recipient through gateway. Embedded file can be downloaded and then extracted with our software application by providing the specified password.

2. PROPOSED ALGORITHM
Our Algorithmic concept in our application is as follows as shown in Fig. 1:

1. Input source file 1, target File 2 and password.
2. Then both the files are converted into binary files.
3. Both binary files are stored in byte array according to the input sequence of the file (First file 1 and then file 2).
4. The password is encrypted and it is stored in the byte array at the end after file 2.
5. The merged file will be stored as the extension of first file, File 1.

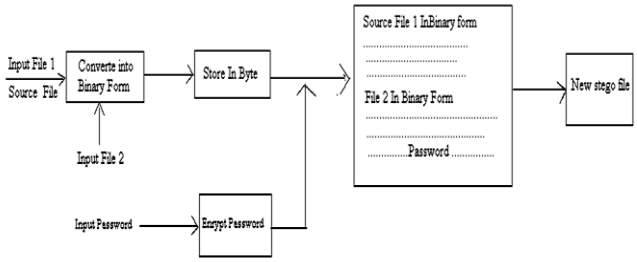

Fig. 1: Embedding a File

Sending and receiving the steganos file is shown in Fig. 2

6. The merged new steganos file is uploaded on Internet.
7. File link is uploaded and password is sent to receiver's mobile via SMS.
8. Receiver downloads and opens file with the given password.

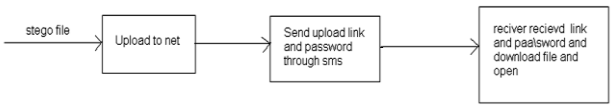

Fig. 2: Sending and Receiving the Steganos File

3. APPLICATION DESCRIPTION
We carry out our implementation by using Microsoft DOT NET Framework 2.0 as front-end and MS-SQL 2005 as back-end.

3.1 Embedding and Retrieving File
The complete flow of the application is as follows:

9. First choose any options as shown in Fig. 3.
10. On selecting option embed file, as shown in Fig. 4, input source file, target file, and target location where embedded file is to be stored. After successfully embedding files as shown in Fig. 5, enter secret password and mobile number of recipient as shown in Fig. 6.
11. Click OK to embed file for specific path.
12. If the choice is to retrieve file then input embedded file and target location where to store the file, using a password.

Fig. 3: Options in Application Software

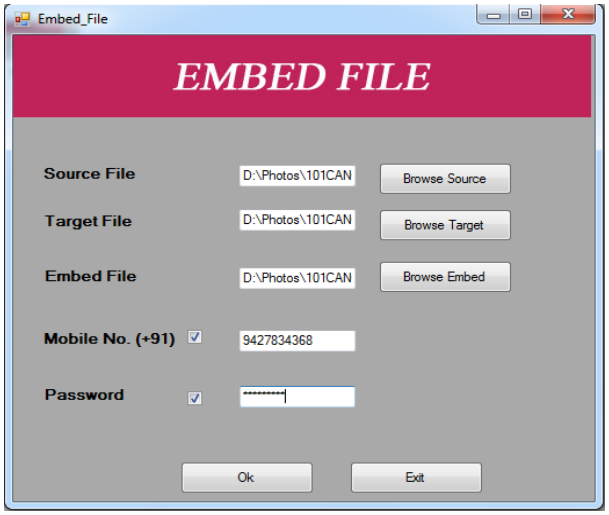

Fig. 4: Embedding File

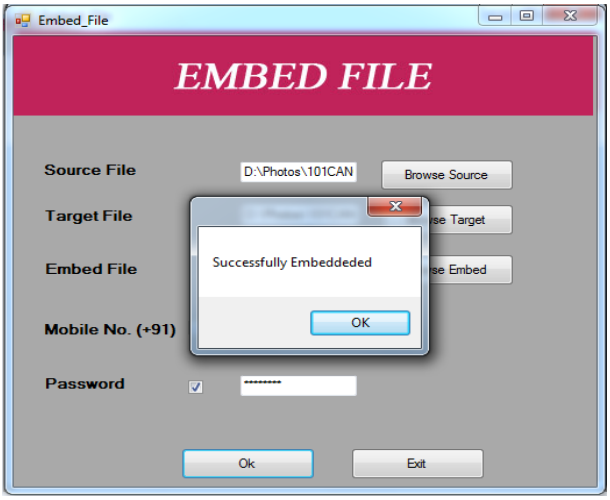

Fig. 5: File Embedded Successfully

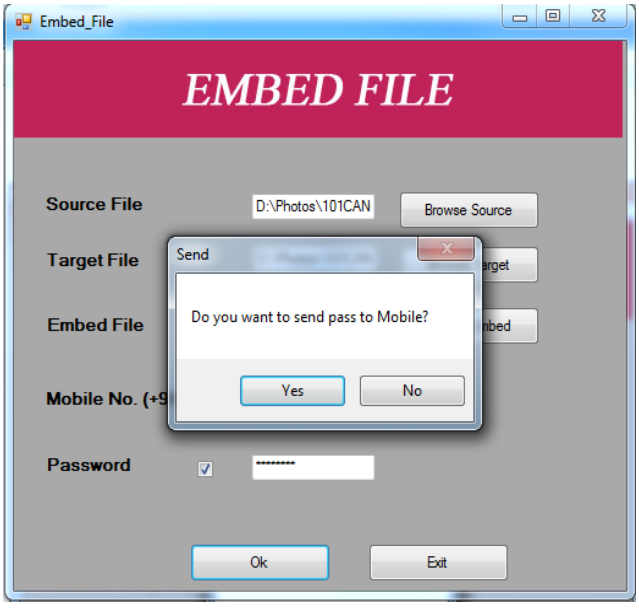

Fig. 6: Send Password via SMS

3.2 Embedding and Retrieving Message

13. First select the source file, target location, message and optional passowrd as shown in Fig. 7.
14. Mobile number of user is to be entered by the user for sending password as shown in Fig. 8.
15. If the choice is to retrieve message then input embedded file and target location where to store the file, using the password as shown in Fig. 9
16. If message is retrieved successfully, a message is shown as shown in Fig. 10.

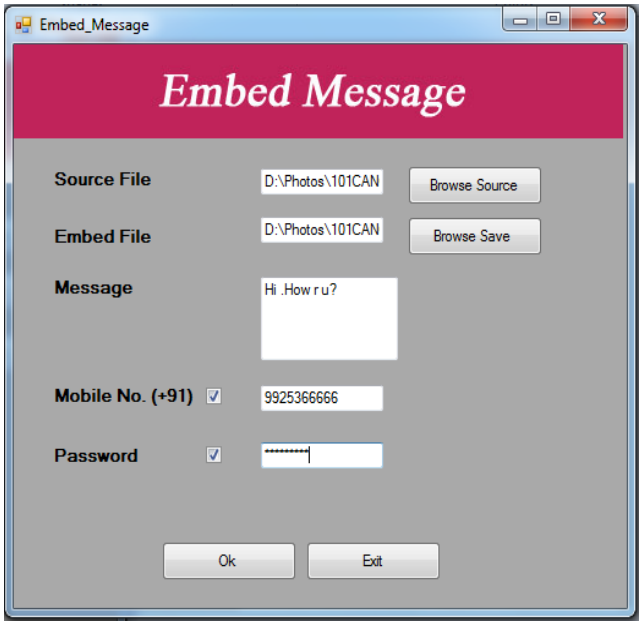

Fig. 7: Embedding Message

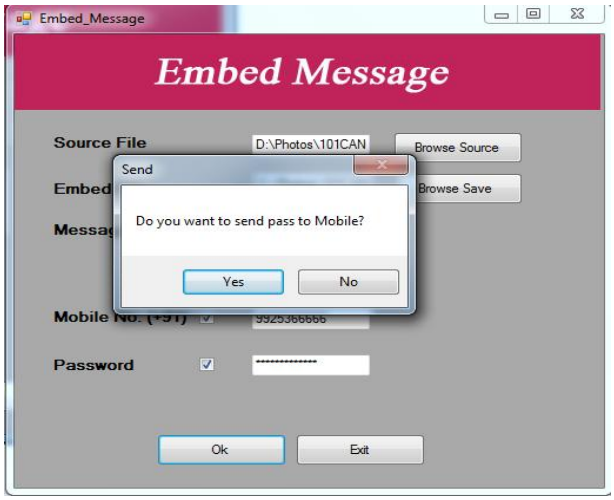

Fig. 8: Send Password via SMS

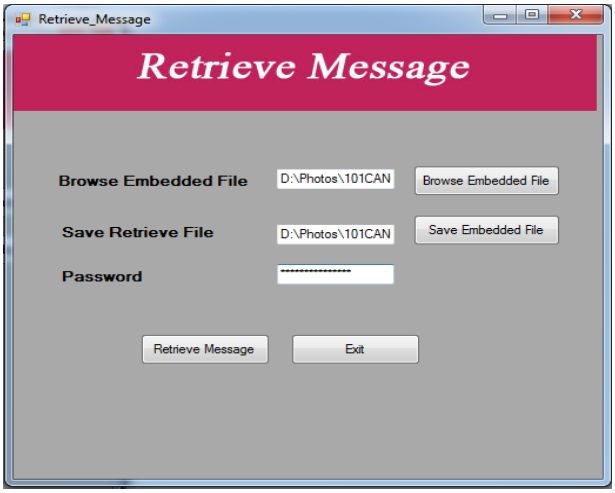

Fig. 9: Retrieve Message

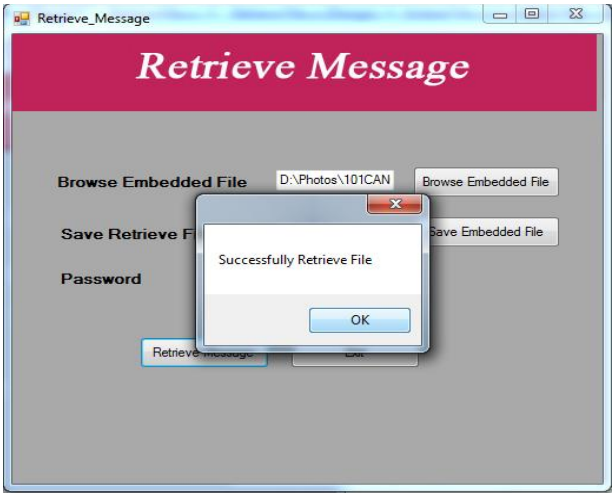

Fig. 10: Message Retrieved Successfully

4. EMBEDDING FILE/MESSAGE AND RETRIEVING FILE/MESSAGE

4.1 Image to Image Embedding and Retrieving

We carry out practical implementation on the described software. Fig. 11 shows a source image file and Fig. 12 shows the target image file. The resulting embedded file is shown in Fig. 13 which looks almost similar to Fig. 11, though it contains hidden file in it. The resulting file size is nearly the addition of size of source file and size of target file. Fig. 14 shows the retrieved image file at the destination.

Fig. 11: Source File with Size 92.7 kb

Fig. 12: Target File with Size 78.6 kb

Fig. 13: Embedded File with Size 171.3 kb

Fig.14: Retrieved Target File with size 78.6kb

4.2 Image to Doc/Text File Embedding and Retrieving

We also carry out implementation for hiding a Microsoft Word document in the same image as Fig. 11. Fig. 15 shows the target document file. The resulting embedded file is shown in Fig. 16 which looks almost similar to Fig. 11, though it contains hidden document file in it. The resulting file size is nearly the addition of size of the source file and size of the target file. Fig. 17 shows the retrieved document file at the destination.

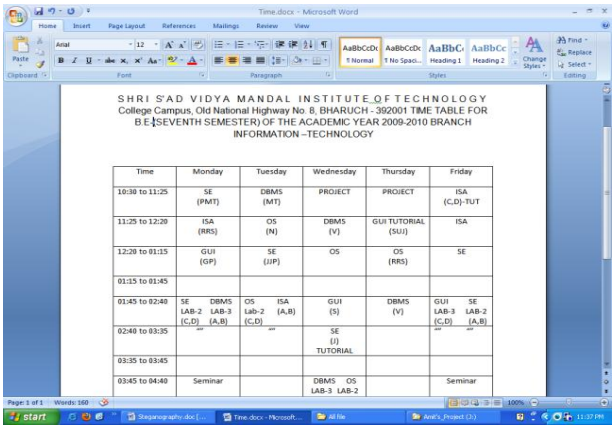

Fig. 15: Target File with size 50 kb

Fig. 16: Embedded File with size 142.7 kb

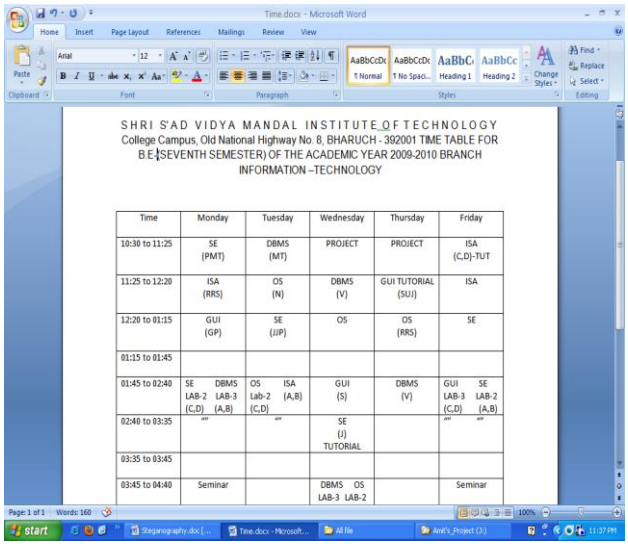

Fig. 17: Retrieved .doc File with Size 50 kb

5. APPLICATIONS

17. It provides one of the most secured File/Data transfer over Internet/Intranet.
18. AFiles can be embedded with the algorithm.
19. The password and other details are being encrypted by AES (Advanced Encryption Standard) which is considered as a standard in the encryption in the current digital world.
20. Location and password of the project will be sent only to the recipient through SMS which shortlists the possibility of any third party indulging in the file.
21. Even if any third party downloads the file directly from FTP, still he requires the password to open it which is only delivered to the recipient by SMS.
22. There is no database which keeps the record of the file details and password. Therefore no one can see the hidden information.
23. There is no information stored for password on the server so user has to manually request for the password if he has lost or misplaced.
24. If the recipient has lost the embedded file there is no way to generate it back rather than requesting the owner to make the steganos file again and upload it again.

6. CONCLUSION

As the world is going digital and information is transferred in the digital form, it is imperative to increase security to avoid any kind of loss or modification in the digital information by a adversaries. We developed algorithm that can prove its usefulness in enterprise level security. We successfully implemented the new steganographic algorithm that can hide a file or a message into a file and can be securely sent to the recipient.

7. REFERENCES

[1] József Lenti,"Steganographic Methods", Periodica Polytechnica Ser. El. Eng. Vol. 44, No. 3-4, pp. 249–258, 2000.

[2] M. Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza, "Steganography in Persian and Arabic Unicode Texts using Pseudo-Space and Pseudo Connection Characters", Journal of Theoretical and Applied Information Technology, pp. 682-687.

[3] Bryan Clair, "Steganography: How to Send a Secret Message", 2001. http://www.strangehorizons.com /2001/20011008/steganography.shtml

[4] Shen Wang, Bian Yang and Xiamu Niu, "A Secure Steganography Method based on Genetic Algorithm", Journal of Information Hiding and Multimedia Signal Processing, Vol. 1, No. 1, pp. 28-35, 2010.

[5] Adnan Abdul-Aziz Gutub, and Manal Mohammad Fattani, "A Novel Arabic Text Steganography Method Using Letter Points and Extensions", World Academy of Science, Engineering and Technology, pp.28-31, 2007.

[6] Faird, Hany. "Detecting Steganographic Messages in Digital Images", 2000.

[7] Shawn D. Dickman, "An Overview of Steganography", 2007.

[8] Amanpreet Kaur, Renu Dhir and Geeta Sikka, "A New Image Steganography Based On First Component Alteration Technique", International Journal of Computer Science and Information Security, Vol.6, No.3, pp. 53-56, 2009.