International Journal of Scientific & Engineering Research, Volume 3, Issue 11, November-2012
ISSN 2229-5518

Level by Level Image Based Data Security Approach

Deepak Gupta, Shikha Gupta, Ranjeet Kumar Singh, Abhisek Gaur

ABSTRACT— Now-a-days, digital communication is part of every computing system. As the usage is increasing in every aspect of data transfer and sharing; so is the threat of information stealing and security. Data Security over the public network is the major concern of every communication expert. Cryptography and Steganography are the major techniques employed for hiding and securing the data from the channel intruders. This paper discusses a new approach that combines the concept of digital image processing, cryptography and steganography to provide a more secure data communication over the network

This paper describes a multilevel technique for hiding the plain text into random 24-bit bitmap images while also scrambling and encrypting the data through image keys (Image Files are used as Keys).The multi-level architecture provides more number of attacks to be made by the intruder to crack the encryption

Index Terms— Cryptography, Steganography, Scrambling

1. INTRODUCTION

Data Security is a much complex and challenging task for developers and designers over the local network. In this paper author uses two major techniques which are (1) Cryptography and (2) Steganography.

2. CRYPTOGRAPHY

In cryptography mainly the study of mathematical techniques are implemented and related to various aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication [3].

The process of conversion or encoding of the plain text into the non-readable code or cipher text is commonly known as ***Cryptography.***

A message which can be easily understood by the sender an, the recipient, and also by someone else who gets an access to that message is known as ***Plain Text****.*

Encrypted plain text which results from above processing is known as ***Cipher Text.***

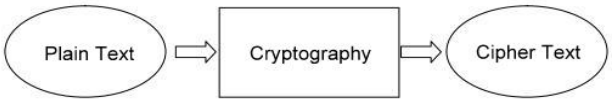


Figure 1: Cryptography Process

3. STEGANOGRAPHY

Steganography is an art of hiding information and an effort to conceal the existence of the embedded information so that no one can suspect the existence of messages apart from the sender and recipient [14].
Security through Obscurity is another form of ***Steganography.***

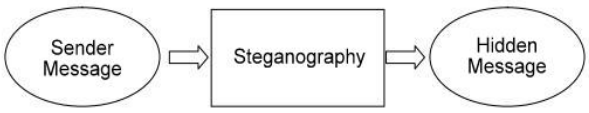


Figure 2: Steganography Process

Combination of Steganography and Cryptography provides much security of data which is to be transmitted.

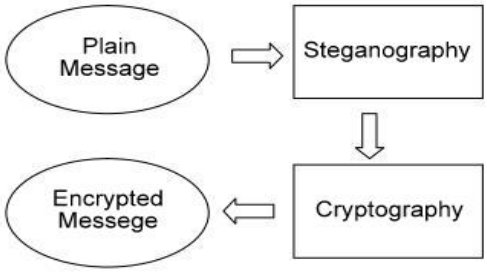


Figure 3: Combination of Cryptography and Steganography

This mechanism provides a multi-dimensional security to the user data and saves it from attacker. The attacker has to apply a decryption algorithm on the image data and then derives the data from image.

4. Encryption Process

The author is using an algorithm which is a 5-level encryption scheme. Each level introduces a new aspect of security. The schematic flow diagram of the encryption process is given below:

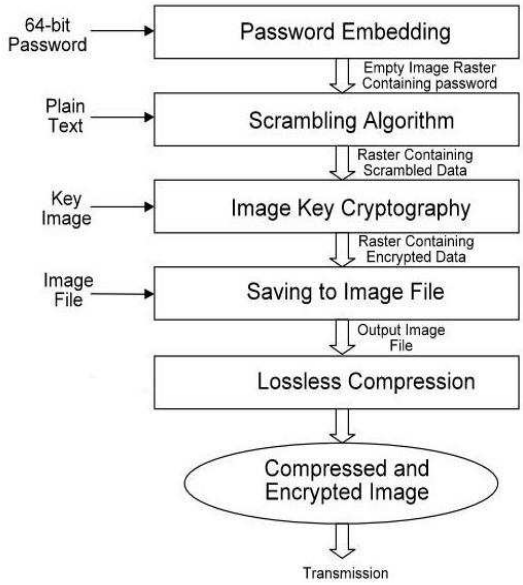


Figure4: Encryption Process

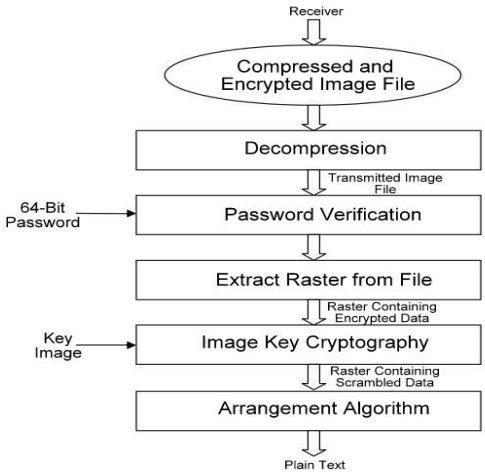
1) Password Embedding Process,
2) Scrambling Algorithm,
3) Image Key Cryptography,
4) Saving the Image File,
5) Lossless Compression.

As the data is encrypted the image file is sent for transmission.

5. Decryption Process

The reverse of the encryption procedure at the receiver end is known as Decryption. Retrieval of original image uses 5 steps procedure which is received from sender is as follow:

1) Decompression,
2) Password Verification,
3) Extract Raster from file,
4) Image Key Cryptography,
5) Rearrangement Algorithm.



Figurer5: Decryption Process

6. Operation Of Algorithm

This algorithm uses the 24-bit BMP image files and applies 5 steps of process for encryption and decryption. Every process magnifies the operations to be performed on various parts of BMP file structure. Lets' discuss the BMP image file format.

6.1 BMP File Structure

A bitmap file consists of four different parts [4]. The first consists of a 14-byte image header.

```
struct BITMAPFILEHEADER {
 WORD   bfType;
 DWORD  bfSize;
 WORD   bfReserved1;
 WORD   bfReserved2;
 DWORD  bfOffBits;
}
```

a) *bfType:* The type of this file usually should be two letters, 'B' and 'M'.
b) *bfSize:* The size of the file in bytes.
c) *bfReserved1 and bfReserved2:* Bits reserved for custom usage or future extensions.
d) *bfOffBits:* This variable indicates how many bytes are from the beginning of the file to the actual pixels.

The BITMAPINFOHEADER consists of all the attributes of the image file. Below given is a overview of BITMAPINFOHEADER structure:

```
struct BITMAPINFOHEADER {
    DWORD biSize;
    LONG  biWidth;
    LONG  biHeight;
```

```
    WORD  biPlanes;
    WORD  biBitCount;
    DWORD biCompression;
    DWORD biSizeImage;
    LONG  biXPelsPerMeter;
    LONG  biYPelsPerMeter;
    DWORD biClrUsed;
    DWORD biClrImportant;
}
```

After the BITMAPINFOHEADER structure comes a RGBQUAD variable; this is a color table. It is commonly either 16 colors or 256 colors, depending on how the file is specified in BITMAPINFOHEADER's biBitCount. In our case, it does not exist in the file because we are using a 24-bit bitmap.

After BITMAPINFO, the file is used to store pixel points, as a reminder. It is stored in form of linear array of bytes which occupies the rest of the file. According to the standards, the image is stored in such a way so that the top left corner is stored at the end of the file. Pixel at the lower-right corner of the bitmap is the starting pixel, and the order of all pixels RGB bytes are B-G-R. The BMP standard also has a mechanism to store some junk bytes "#000000" (in case of 3 junk bytes) after every pixel line or scan line inside the file. The count of junk bytes after every scan line can be determined as:

$$(Total\ size\ on\ disk = \frac{x * y * 3 + 14 + 40}{y})$$

where x and y represent the width and height of image respectively.

7. Algorithm Steps

7.1. Password Embedding

The BITMAPFILEHEADER are used to store the password bits. This password bit verifies an authenticated user at receiver side during encryption.

Input by the user is 64-bit (two words) password. The password is encrypted using an encryption algorithm or it can be put without any modification.

At receiver side, the decryption process uses that password are retrieved from the file. This mechanism is like an authentication system for the user of key. If a person doesn't have the password file one won't be able to crack the system.

7.2. Scrambling Algorithm

Key part of the encryption scheme is Scrambling Algorithm. This algorithm scrambles the data in 3-D space of RGB pixels in such a way, that it makes the data in non-readable form during retrieval.

This algorithm uses the 3-D space of pixel colors and divides the data which is in the form of image in three axes - R-Axis, G-Axis and B-Axis. The user data is taken in an orderly byte-by-byte manner and arranged along the three axes.

Firstly the user data is determined and estimates its size and then divided by 3. The values after division are known as Axis-coefficient. Now next step includes the filling of user data onto the R-Axis in a byte-by-byte order until the axis counter reaches the Axis-coefficient. Once the R-axis is filled with the coefficient, the remaining data is put upon G-Axis and in the same way for B-Axis.

The data is to be filled for the particular byte up to the nth pixel. For example, the first byte of user data is to be place in R-byte of first pixel, and then second byte is to place in R-byte of second pixel and so on until the threshold coefficient is achieved. After this, the first byte of user data is to be place in G-byte of first pixel and so on. In this similar manner, the process is completed for the B-bytes.

The image resolution can select from some predefined templates or can be defined by the user through input. When above process implemented in BMP File format it scrambles the complete arrangement of the data and makes the data more secure which leads to more unperceivable for the intruder. Internal structure of image changes with the first byte and the image header will be the last byte of the user data. In this way the data is multiples with Axis-coefficient.

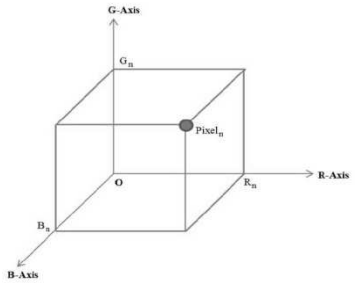


Figure6: Dimensional RGB representation of pixel

The value of Axis-Coefficient should be place after the EOF (End-of File) character. This value will be used at the receiver side for decryption and manages the possible rearrangements.

7.3. Image Key Cryptography:

Image key cryptography is a key concept of using an image like an encryption key [7]. This algorithm uses a key image which is to be used by the user as an input. This image is used to encryption of the scrambled data is followed by an image which was already obtained from the previous step.

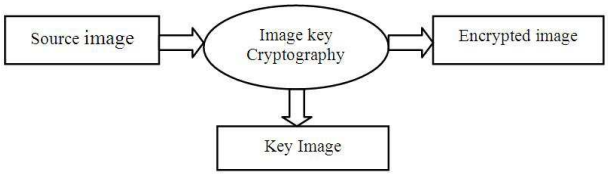


Figure7: Image Key Cryptography

The resolution of the key image and source image must be the same which was already derived in a previous step. Next step includes two operations on every pixel of image to perform- 1) Gray Encoding, and 2) XOR Operation.

In this, the first step is to select a pixel from source image and used as an input to a gray encoder, and then the output of encoder is eXORed with the corresponding pixel of Key Image. This makes the destination pixel is to place in the file.

This process in a similar way is carried out for every pixel from source and key image.
At the receivers' side, the reverse process takes place. Initially the input pixel is eXORed with the key pixel, and then the result is passed through a Gray Decoder to obtain the original pixel data.

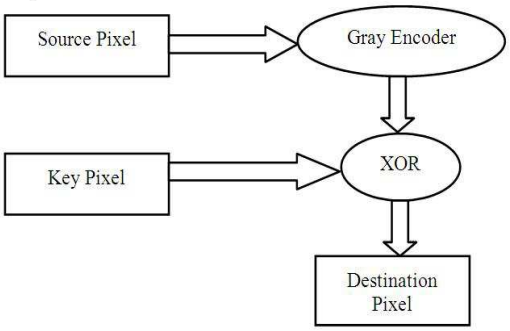


Figure8: Steps at Sender End

Now, that we have the encrypted data, in our hand, this data is put inside the BMP image raster and this raster is passed to the BMP codec. Once the raster is stored inside the image file, the reserved bits of header are replaced with the password bits determined in first step and Axis-Coefficient is placed at the End-of-File. After accomplishing these entire tasks image file is stored to the disk.

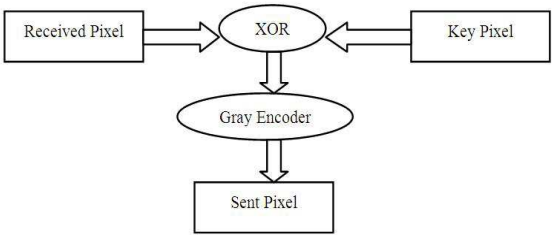


Figure9: Steps at Receiver End

Now the file can be transmitted with a lossless compression technique so that one can retrieve the original data and it also minimize the traffic during transmission. The important feature of this compression methodology is that, it improves both level of security of the data i.e., at transmission level and at decryption level.

8. Finished Tasks:

Now, that we have the encrypted data, in our hand, this data is put inside the BMP image raster and this raster is passed to the BMP codec. Once the raster is stored inside the image file, the reserved bits of header are replaced with the password bits determined in first step and Axis-Coefficient is placed at the End-of-File. After accomplishing these entire tasks image file is stored to the disk.

This file can be transmitted as-it-is or a lossless compression can be carried out to reduce the transmission traffic. The uniqueness of compression methodology can also improve the security of the data as only the corresponding decompression algorithm can retrieve the image from the compressed file.

9. Pros And Cons

The major advantage of this algorithm is the unmatchable data security. It provides an encrypting image cover to the user data. Although not a Steganography technique, this algorithm also provides the mechanism of data hiding, in sense that data is not directly perceivable by the intruder.

Apart from the efficiency of algorithm itself, it provides two security constraints to the users on two ends of communication channel- Password, and Key Image. Therefore reduces the chances of prediction as well as accidental attacks.

Every coin has its two faces and since nothing is perfect in this world, this algorithm is no exception. Behind the curtain of enhanced data security this algorithm provides a huge wall of increased complexities, implementation difficulties and file handling issues.

10. Conclusion

Finally, the security level of the output is very high, so the situations where the very high level data security is needed, this algorithm is the perfect choice for communication experts.
This level-by-level architecture makes the data very reliable in terms of security and make very difficult for the intruder to obtain the user data. To access the user data, attacker has to make various attempts and have to crack various walls of encryption.

11. Future Scope

This algorithm currently concentrates the use of the 24-bit Bitmap images. In future, instead of 24-bit Bitmap image this can be extended to modern image file formats like JPEG, TIFF and PNG. Although not required, the modification operation at the Image Key Cryptography step can be made more complex to enhance the data security.
In last words, extensibility has the limits of human brain. As the technology grows new methods and algorithms evolve. Only extensions can get anything to be with the flow of requirements.

References

[1] Musheer ahmed and M. shamsher Alam,"A new Algorithm of Encryption and decryption of Images Using Chaotic Mapping",Musheer Ahmed et al/international Journal on Computer Science and Engineering,Vol2(1),2009,46-50

[2] Menezes, P. van Oorschot, and S. Vanstonen, Handbook of Applied Cryptography, 5th ed. CRC Press, 2001, ch.1, pp 4.

[3] Dhananjay K. Theckedath, "Image Processing using MATLAB codes".

[4] Y.B. Mao,G. Chen. S.G. Lian,"A novel fast image Encryption scheme based on the 3D chaotic baker map,"Int. j. Bifurcate Chaos, vol. 14,pp.3613-3624,2004

[5] Yas A. Alsultanny,"Image encryption by cipher feedback mode", ICIC International Journal vol.3,589-596

[6] S.K. Panigrahy,B.Achraya,and D.jena,"Image Encryption using self-Invertible Key Matrix o Hill Cipher Algorithm,"1st International Conference on Advances in Computing ,Chikhli,India,21-22February 2008

[7] I.Qzturk and I.Sogupinar, "Analysis and Comparison of Image Encryption Algorithms", International Journal of Information Tec.

[8] Chin-Chen Chang, Min-Shian Hawang, tung-shou Chen, "A New Encryption Algorithm for Image Cryptosystems", The Journal of Systems and Software 58(2001),83-91

[9] Piyush Marwaha and Paresh Marwaha, "Visual Cryptographic Steganography in Images", Infosys Technologies Limited, India.

[10] A.Mitra,Y V.Subba Rao,and S.R.M. Prasnna, "A new Image encryption approach using combinational permutation techniques", Journal of Computer Science,Vol.1,no.1,2006,p.127

[11] A.Sinha, K.Singh, "A technique foe image encryption using digital signature", Optics Communications,2003,pp. 1-6

[12] S.S. Maniccam, N.G. Bourbakis, "Lossles Image Compression and Encryption using SCAN," Pattern Recognization, vol. 34,2001, pp.229-1245

[13] Arvind Kumar and Km. Pooja, "Steganography- A Data Hiding Technique", International Journal of Computer Applications, vol. 9, No. 7, November 2010

[14] Hanbo Sun, "Exploring the internal structure of a 24-bit Uncompressed Bitmap File". Available: http://www.codeguru.com.

[15] Rafael C.Gonzalez and Richard E.Woods, *Digital Image Processing*, 3th ed. Addison-Wesley, 1992.

———————————— ◆ ————————————

- *Deepak Gupta, Asst.Prof, Dept of CSE, GEC Ajmer*
  gupta_de@rediffmail.com
- *Shikha Gupta, Asst.Prof, Dept of IT, GEC Ajmer*
  shikhagupta17@rediffmail.com
- *Ranjeet Kumar Singh, M.Tech (IT), GEC Ajme*
  er.ranjeet1985@gmail.comr
- *Abhisek Gour, M.Tech (CSE), MBM Engg. College, Jodhpur*
  abhisek.gour@gmail.com