ISSN: 2229-6948 (ONLINE)
DOI: 10.21917/ijct.2013.0111

STEGANODB – A SECURE DATABASE USING STEGANOGRAPHY

R. Rejani[1], D. Murugan[2] and Deepu V. Krishnan[3]
[1]Department of Computer Science and Engineering, Manonmanium Sundarnar University, India
E-mail: [1]rejani@gmail.com, [2]dhanushkodim@yahoo.com
[3]Infosys Limited, Technopark Campus, India
E-mail: deepu_krishnan@infosys.com

Abstract

The safety of data and communications for both organizations and personal purposes which at the same time is easy to maintain and having less overhead on system resources is a very important issue. This paper presents a new/alternate secure Database system based on steganography for data hiding. The system provides integrity more confidentiality, and authentication during access or editing of confidential data. The proposed DB system uses steganography technique to store a database of records. The system allows a user to create basic tables and records which are hidden from others inside an image. In this paper propose an architecture which can be used by application developers to retrieve data from the created database in an easy manner. The proposed method is highly useful for use as an embedded DB also in mobile computing as it can store small amount of data easily.

Keywords:
Data Protection, Secure Database, Steganography, StegoDB

1. INTRODUCTION

Confidentiality of data is a crucial issue in today's diverse environments involving computers, tablets and smart phones. In mobile environment for example there will be need to store financial data, personal information and passwords which should not reach hackers or other miscreants. Many of the applications today use methods of storing critical information like encryption and sometimes even as plaintext. Cryptography implies that there should be a secret cipher present. Even if a hacker might not be able to solve the cipher, he might be still be able to corrupt or delete the message thereby making the information useless and also making the application associated also useless.

A different approach is to the security question is to hide the secret information in a way that users are not aware of its existence, i.e., as users do not know about the information, the secrecy is kept. Hiding information in media like image, audio and video is what the digital steganography aims for. Most of the current Steganography techniques tend to affect the quality of the image as well if there is more data inserted into the image. In this paper we have tried to avoid this issue by using the best possible steganography algorithms at the same time finding the maximum amount of data that can be inserted into an image before starting the insert process.

The problem with traditional steganography technique is that the application developer always faces the overhead of creating specialized procedures and programming logic for handling the encryption or data storing logic. Hence to avoid this and to make the life of an application developer much easier we are proposing a new steganography database architecture which is very easy to setup and provides the application developer easy to use methods which can be called easily while developing applications. The proposed steganography based database will use digital images for storing related data but at the same time will allow data to be fetched using the standard queries instead of using complex algorithms. We also propose to provide application libraries which will allow the developers to use simple queries to fetch the data and also will act as a layer between the application and the stegano images while performing the create, read, delete operations.

This paper is organized as follows: In Section 2, we will be describing about the related work in this area. Section 3 presents background concepts and important technologies used. Section 4 will be talking about the steganography that we have developed and the section 5 describes the proposed architecture for security.

2. RELATED WORK

Statistically Invisible Steganography (SIS) to hide data in JPEG images is described in [1], which are invulnerable to current methods of stegano analysis. Specifically, this method does this by embedding data in the selective coefficients of the selective Discrete Cosine Transform (DCT) blocks that have high complexity and maintain original statistics. In [4] there is a set of criteria to analyze and evaluate the strengths and weaknesses of the presented techniques. It allows for the change of the plans intensity of (24 bit) colored image to embed secret message in a specific distance between them. This method is based on changing the distance of two random selected pixel channels in a specific range that represent hidden data. It is intended to be robust against distortion and takes into account other important criteria such as security and capacity allowing for the development of a successful steganographic system. The security is accounted for by adapting a robust encryption algorithm. A high capacity is achieved by compressing encrypted message, and using multiple carriers to fit secret message. Each distance between two random pixel channels will embed one message bit by adjusting the distance to the closest value in the distance difference sequence whose binary value is identical to the message bit. In the de-embedding phase, the stego-key is used to generate the same distance differences and binary sequences as those used in the embedding process. For each random selected pixel channels, the closest distance between them is computed to find the hidden message from the corresponding binary value.

The hidden information is considered as additive noise to the image in [3]. The alpha-trimmed method estimates steganographic messages within images in the spatial domain and provide flexibility for classifying various steganography methods in the JPEG compression domain. Three JPEG steganography methods along with three embedded message files applied to an image data set, this method results in better separability between clean and steganographic classes. These results are based on comparisons between the presented method and two existing methods in which classification accuracies are increased by as much as 32%.

The R.S. Gutte, Y.D. Chincholkar and P.U. Lahane's research proposal [9] the secret communication system has two layered security. In the first level encryption of the text using extended substitution algorithm and in the second level through embedding the encrypted text into LSBs variably. From the experimentation that inserting the data at three LSB positions does not change any image parameters like PSNR, Mean or Standard deviation. Therefore, it retains the image quality similar to two LSB scheme. Using this can be able to conceal almost all types of alphabets (upper & lower case), special characters and mathematical symbols. The variable x takes values as 0, 1, 2, and 3. Embedding the cipher at LSBs is decided by variable x. As the LSB in each pixel are not same but decided according to variable value. It helps to minimizing the error.

Herve Chabanne Morpho, Mehdi Tibouchi Ecole normale superieure [11] implemented a security system for protecting PIN numbers and ID statements using a class of cryptographic protocols called Password Authenticated Key Exchange (PAKE).

By performing an analysis on related work, two important features arise when developing steganography for secret writing in compressed images: First, the stego image must be viewable by regular software. This ensures the stego image to be perceived as a common image, without embedded messages. Second, it must be possible to retrieve an exact copy of the secret information embedded into the stego image.

3. CONCEPTS AND TECHNOLOGIES

3.1 STEGANOGRAPHY

Steganography is the technology of writing hidden messages in such a way that no one apart from the recipient knows of the existence of the message. This is different from cryptography because it is a technique used to embed secret information into non-secret data, preventing the message from being detected by non-authorized people. With the advantage of digital media, we can hide digital information within files. For example, the color value of every 100th pixel within a digital image could represent a letter in the alphabet. Digital steganography aims to provide the means to hide Information into digital media like text, image, audio and video. The media object used to receive the secret information is named Stego Object.

Different techniques of steganography are used for data hiding. For audio steganography Discrete Wavelet Transforms (DWT) or Direct Sequence Spread Spectrum (DSSS) is used. Since we are using Digital image steganography we make use of Least Bit Significant Steganography (LSB) based steganography. Least significant bit (LSB) encoding decoding is the most popular method of the coding encoding techniques used for digital images. The LSB of each byte (8 bits) in an image for a secret message can store 3 bits of data in each pixel for 24-bit images and 1 bit in each pixel for 8-bit images. More information can be stored in a 24-bit image. Depending on the color scheme used for the cover image (i.e., all gray), it is possible to take 2 LSB's from one byte without the human visual system (HVS) being able to tell the difference.

3.2 JPEG

JPEG is the most commonly used method of lossy compression for digital photography /image. The percentage of compression can be adjusted, by allowing a selectable tradeoff between image quality and storage size. JPEG typically achieves 10:1 compression with little perceptible loss in image quality. It greatly reduces the size of a digital image by removing excess image data and calculating a close approximation of the original image. JPEG compression is a usually used image file formats. JPEG/Exif is the usual image format used by digital cameras and other photographic image capture devices. Along with JPEG/JFIF, it is the widely used format for storing and transmitting photographic images on the World Wide Web. JPEG supports a maximum image size of 65535×65535. The file name extension of a JPEG image can be .jpg, .jpeg or .jpe.

3.3 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy to use. Also it is easy for machines to compile. It is based on the Java Script Language. JSON is a text format that is completely language independent but it uses conventions that are familiar to programmers of the C-family of languages, including C#, C++, C, Java, JavaScript, Python, Perl, and many others. These are the properties make JSON an ideal as a data-interchange language. Mainly JSON is built on two structures:

- A collection of name or value pairs. In other languages, it is realized as record, object, structure, dictionary, keyed list, hash table or associative array.
- Values as ordered list. In many languages, this is realized as vector, list, array, or sequence.

One of the biggest advantages of JSON format is that it maps to most of the existing data structures in programming languages and it has a layout that is quite simple enough to keep coding and db design simple. It is also simple and flexible enough to express most data in a fairly natural way. Example of a JSON structure is given as,

Table.1. Sample JSON record structure

```
{"firstName": "D",
   "lastName": "Krishnan",
   "age": 25,
   "address": {
      "streetAddress": "15E, LIC Lane",
      "city": "TVM",
      "state": "Kerala",
      "postalCode": 695004
},   "phoneNumber": [
      {"type": "home", "number": "471 3277-341"},
      {"type": "fax","number": "471 555-456"} ]}
```

4. PROPOSED TECHNIQUE

The technique proposed in this paper creates a parallel data structure based on JSON to store information within a JPEG image. We will be using LSB based steganography for this purpose. While at the same time it is possible to view the image in a regular image viewer or any other software.

Using JSON will allow us to store related data rather easily and also will allow us to retrieve the data in an easy manner. Therefore this work overcomes some of the drawbacks that are there in a traditional steganography from an application developer's perspective.

However one another important criteria of the system is to make sure that the optimum amount of data is stored in the image without affecting the integrity of the image as such. To make sure this happens we first calculate the optimum amount of data that can be stored on an image. This will make sure that no noise is added to the particular image.

The STEGDB package will accept the following functions along with an input image.

1. Stegdb.insert
2. Stegdb.upsert
3. Stegdb.delete
4. Stegdb.remove
5. Stegdb.find

The Fig.1 shows the architecture of the steganoDB proposed including the various operations that are handled by the StegoDB package.

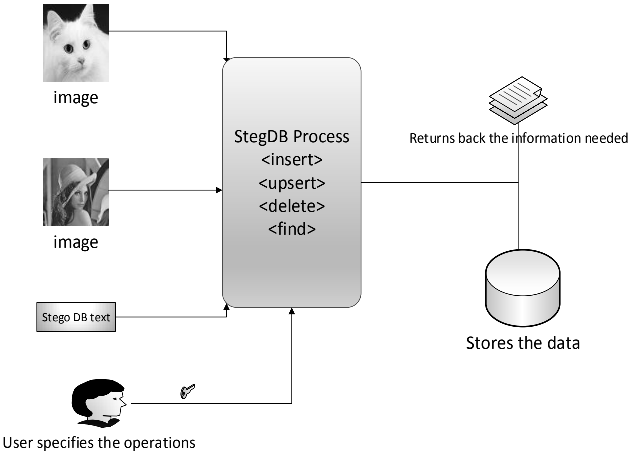


Fig.1. SteganoDB architecture

Along with storing the data within the image we have to find out the optimum possible storage that is possible within an image. For this we can find the PSNR value of the encoded image (Peak Signal to Noise Ratio).

For calculating the PSNR we have to find the MSE first (Mean Squared Error) *MSE* is defined as,

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I(i,j) - K(i,j)\right]^2 \tag{1}$$

The *PSNR* is defined as,

$$\begin{aligned} PSNR &= 10.\log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20.\log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20.\log_{10}(MAX_I) - 10.\log_{10}(MSE) \end{aligned} \tag{2}$$

Maximum possible pixel value of the image is represented by MAX_I.

In our SteganoDB package, we allow the PSNR value of a base image to reach till the value of 30 from Eq.(2).

For every request the package will check the function requested by the user.

- Inserts – The stegoDB package checks for the available space in the image and if the size request exceeds, the package will return an error.
- Upsert – The stegoDB package checks if there is an existing record, if the record exists it will be updated, else the record will be inserted as a new record.
- Delete – The stegoDB package will check and delete the record specified.
- Remove – The stegoDB package will remove all the records from the image.
- Find – The stegoDB package will list the records that match the search criteria.

In order to encode the data in JSON structure, we will be using the LSB based Steganography algorithm. The entire encoding of data will be taken care by the package itself and the application developer needs not worry about the complexity of algorithms used internally within the system.

The system can also be used as a client server model for programs which need to authenticate a user. For example let us consider a scenario where the user wants to authenticate his password against a centralized app server. Instead of sending the password as plain text over an unsafe public network, the password and other critical information can be encrypted and encoded as a StegoDB inside an image and passed over the network. This will allow for an added protection against hackers. At the server side an automated mechanism can be implemented to read the StegoDB and decrypt the password and other details that are encoded inside the stego image. The system is detailed in Fig.2 below,

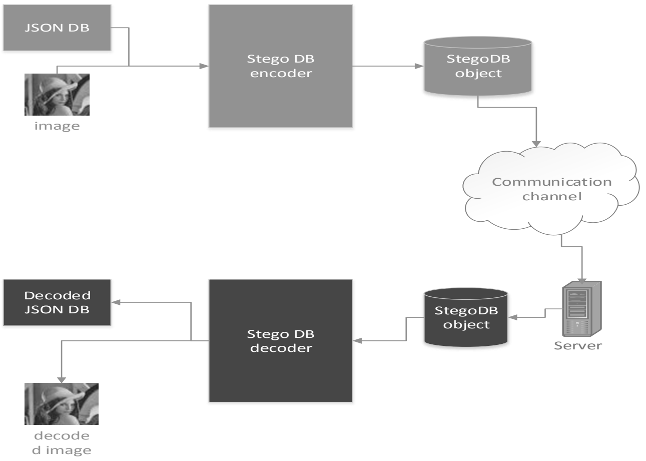


Fig.2. SteganoDB System

Furthermore the system can also have the option of providing encryption over the standard steganoDB to provide additional security to the user. The architecture of such a secure SteganoDB is provided below. The encryption system will provide much more security and make it extremely hard for the hacker to extract the actual data from the SteganoDB.

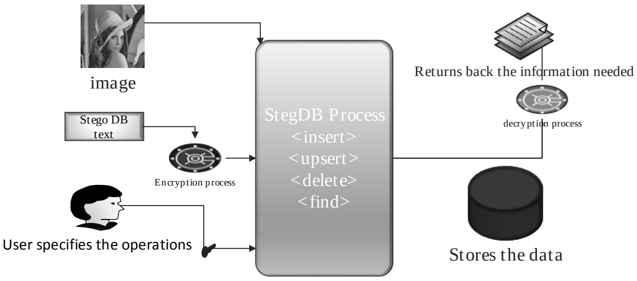


Fig.3. SteganoDB Processes

5. EXPERIMENTAL ANALYSIS

The StegoDB comes as a package which can be included in .net based windows development tools. The initial package will also come with tools to create the database on a new image. Afterwards the DB functions can be called from the program created using the various .net technologies (C#, VB, C++) to perform the DB operations like insert, delete, upsert etc.

The processing of the system will involve the below steps:

First of all it gives an option to the user to select the cover image. It can be any of the image format file. Since this system prefers to reduce the file size and its advantage we use jpeg format for output.

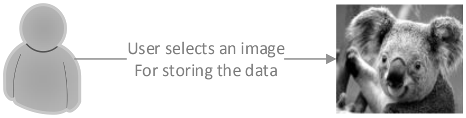


Fig.4. Asking user for the image to be used for storing data

According to the data structure specified in this system the user can provide the information/data. This data is the stegan. This data is encoded and stored inside the cover image. This preserves the data more secure. This data is considered same as the relational database data. This will be stored inside the image until the user asked to resolve the data. Then only this data is visible for the human visual system.

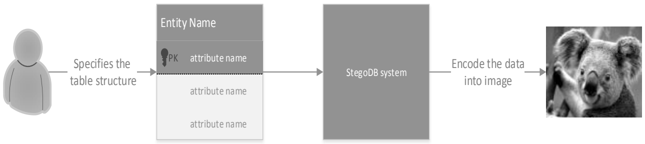


Fig.5. Data insertion step

At the retrieval side the data is first decoded from the cover image and then pass this data to the user.

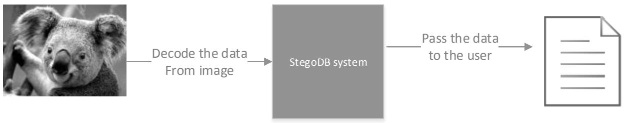
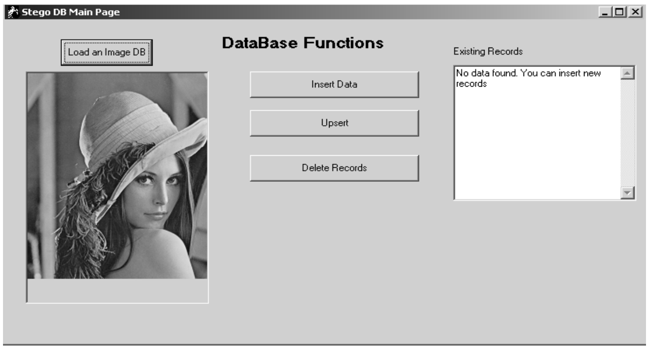


Fig.6. Data retrieval step

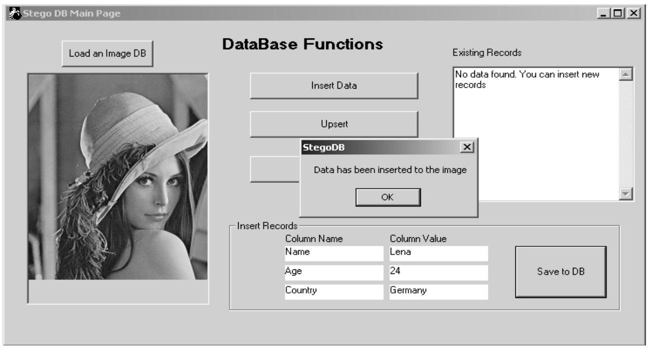
The experimentation results were done on a PC by developing a Graphical User Interface in Windows to store the data into Stego DB and then retrieve and display it. The below figure shows the main interface which allows to load an image and load data into it in JSON format.



No data found. You can insert new records

Fig.7. Main screen

There are separate options for inserting, upserting and deleting data. If there are no records existing it will display an error saying that no records could be found.



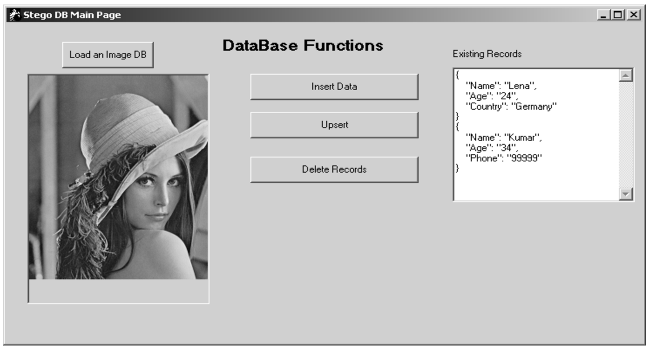
No data found. You can insert new records

Data has been inserted to the image

Column Name	Column Value
Name	Lena
Age	24
Country	Germany

Fig.8. Data insertion step

The records as per the input will be inserted into the image in JSON format using Steganography.



```
{
    "Name": "Lena",
    "Age": "24",
    "Country": "Germany"
}
{
    "Name": "Kumar",
    "Age": "34",
    "Phone": "99999"
}
```

Fig.9. Data insertion step

Main screen of application after inserting couple of records in the previous step.

6. CONCLUSION

This paper presents a new novel and non-conventional architecture to create a secure Stegano DB which will be storing related data in JSON format. This helps the application developers to include a security mechanism within their applications to store the data securely. Unlike other database security methods, the overhead of providing security is not there as the method used is very simple. Due to the simplicity of the method this method can be also suited for modern mobile and embedded systems. Since the user can change the cover image as per his/her wish nobody suspects about the image and stegan inside of it. Furthermore the system can also be extended to be used as a client server system.

Another important thing is that only users who have the StegoDB package will be able to insert and edit the data on the images. One drawback of the system is that because we are storing data on to images, there will certainly be limitation in storing data before the image quality starts degrading.

To overcome this as a future expansion we are working to adapt the system to store the data into video and audio files which will allow us to store much more data.

REFERENCES

[1] Qingzhong Liu, Andrew H. Sung, Zhongxue Chen and Xudong Huang, "A JPEG-Based Statistically Invisible Steganography", *Proceedings of the Third International Conference on Internet Multimedia Computing and Service*, pp. 78-81, 2011.

[2] Mei-Ching Chen, S. S. Agaian, C. L. P. Chen and B. M. Rodriguez, "Alpha-trimmed image estimation for JPEG steganography detection", *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pp. 4581-4585, 2009.

[3] Hassan Mathkour, Batool Al-Sadoon and Ameur Touir, "A New Image Steganography Technique", *Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1-4, 2008.

[4] D. R. L. Prasanna, L. Jani Anbarasi and M. Jenila Vincent, "A Novel Approach for Secret Data Transfer using Image Steganography and Visual Cryptography", *Proceedings of the International Conference on Communication, Computing & Security*, pp. 596-599, 2011.

[5] Khan Farhan Rafat and Muhammad Sher, "Digital Steganography for ASCII Text Documents Ph.D. research proposal", *Proceedings of the 7th International Conference on Frontiers of Information Technology*, 2009.

[6] K. S. Babu, K. B. Raja, K. Kiran Kumar, Manjula Devi T. H., Venugopal K. R and L. M. Patnaik, "Authentication of Secret Information in Image Steganography", *IEEE Region 10 Conference*, pp. 1-6, 2008.

[7] Nicholas J. Hopper, Luis von Ahn and John Langford, "Provably Secure Steganography", *IEEE Transactions on Computers*, Vol. 58, No. 5, pp. 662-676, 2009.

[8] R. S. Gutte, Y. D. Chincholkar and P. U. Lahane, "Steganography for Two and Three LSBs Using Extended Substitution Algorithm", *ICTACT Journal on Communication Technology*, Vol. 4, No. 1, pp. 685-690, 2013.

[9] ShengDun Hu and KinTak U, "A Novel Video Steganography based on Non-uniform Rectangular Partition", *Proceedings of IEEE International Conference on Computational Science and Engineering*, pp. 57-61, 2011.

[10] Che-Wei Lee and Wen-Hsiang Tsai, "A Secret-Sharing-Based Method for Authentication of Grayscale Document Images via the Use of the PNG Image With a Data Repair Capability", *IEEE Transactions on Image Processing*, Vol. 21, No. 1, pp. 207-218, 2012.

[11] Herve Chabanne Morpho, Mehdi Tibouchi Ecole normale superieure, "Securing E-passports with Elliptic Curves", *IEEE Security & Privacy*, Vol. 9, No. 2, pp. 75-78, 2011.

[12] Tsung-Yuan Liu and Wen-Hsiang Tsai, "Quotation Authentication: A New Approach and Efficient Solutions by Cascaded Hashing Techniques", *IEEE Transactions on Information Forensics and Security*, Vol. 5, No. 4, pp. 945-954, 2010.

[13] Daniela Stanescu, Valentin Stangaciu, Ioana Ghergulescu and Mircea Stratulat, "Steganography on Embedded Devices", *Proceedings of 5th International Symposium on Applied Computational Intelligence and Informatics*, pp. 313-318, 2009.

[14] Tao Zhang, Wenxiang Li, Yan Zhang and Xijian Ping, "Detection of LSB Matching Steganography Based on Distribution of Pixel Differences in Natural Images", *Proceedings of International Conference on Image Analysis and Signal Processing*, pp. 548-552, 2010.

[15] Ge Huayong, Huang Mingsheng and Wang Qian, "Steganography and Steganalysis based on Digital Image", Proceedings of 4th International Congress on Image and Signal Processing, pp. 252-255, 2011.