Review Paper on Image Based Steganography and Java Encryption API

Deepika[1] Dr Sanjay Kumar[2]
[1,2]Department of Computer Engineering
[1,2]Jaipur National University, Jaipur, Rajasthan, India

Abstract—Cryptography and steganography are widely used techniques for manipulation of messages in order to hide that message. These techniques have a lots of applications in computer science and other related fields. They are used to protect important data like corporate data,e-mail messages, credit card information etc. More specifically, steganography is the art and science of communication in a way which hides the existence of the original message. A encryption is technique which converts the message in some other format. Encryption with java provides a very secure way of communication in any of the media Java provides a good set of APIs for cryptography. This paper presents different methods of steganography with introduction to available java APIs which can be suggested for more secure messages.

Key words: Image Based Steganography, Java Encryption API

I. INTRODUCTION

In the current trends most of the individuals prefer to use internet as the main medium for transferring data from one end to another end. Internet has provided us a lots of ways for ways for transmitting data for example emails, chat, FTP etc.

The data transition is very simple, fast and accurate using the internet. However, the main problems with sending data over the internet is the security threat it poses i.e. the confidential data can be stolen or hacked in many ways. Therefore it becomes very important to take data security into consideration, as it is one of the most essential factors that need attention during the process of data transferring. Security is one of the main interest area for a lots of researchers. Data security means protection of data from unauthorized users and providing high security to prevent data. The area of data security is one of the important are of study in computer science due to the massive increase in data transfer rate over the internet. In order to improve the security features in data transfers over the internet, many techniques have been developed in last few years like: Cryptography, Steganography and digital watermarking. While Cryptography is a method to conceal information by encrypting it to cipher texts and transmitting it to the intended receiver using an unknown key, Steganography provides further security by hiding the cipher text into a seemingly invisible image or other formats[11]

There are many types of steganography methods. Following different techniques can be used for steganography.

- Text Steganography
- Image Steganography
- Audio Steganography

A. *Text Steganography*

In Text steganography, text formatting or certain characteristics of text are altered for achiechiving steganography.

B. *Image Steganography*

In image steganography, the information can be sent by hiding it behind any of the image. So by general eyes information cannot be seen. There are various methods which are proposed for image steganography so that the information can be highly secured with minimum distortion of the image. To hide a message behind an image without changing its visible properties or distortion, the cover source can be altered in ”noisy” areas with many color variations, so less attention will be drawn to the modifications.

C. *Audio Steganography*

In audio steganography, the message is added behind digitized audio signal which result a very less amount of alteration of binary representation of the corresponding audio file. There are several methods are available for audio steganography.

II. IMAGE STEGANOGRAPHY

The main objective of paper is to discuss only image steganography. There are various methods of steganography:

- Least significant bit (LSB) method
- Transform domain techniques
- Statistical methods
- Distortion techniques

A. *Least Significant Bit (LSB) Method*

Least significant bit (LSB) insertion is a common and general way to hide information behind an image. In this method the Least significant bit of a byte is changed with a bit of message. This technique can be used for images, audios and video steganography. Even after alteration of images using LSB method, the resulting image will look same to the original image.. For example, if we want image steganography and wants to hide letter A in three pixels(assuming no compression). The original raster data for 3 pixels (9 bytes) may be

(00100011 11111001 11001010)
(00100001 11001000 11111001)
(11111000 00100011 11100001)

The binary value for A is 10000001. By Inserting the binary value for A in the three pixels would results in

(00100011 11111000 11001010)
(00100000 11001000 11111000)
(11111000 00100011 11100001)

The underlined bits are the only three actually changed in the 8 bytes used. On average, LSB requires that only half the bits in an image be changed. You can hide data in the least and second least significant bits and still the human eye would not be able to discern it[13].

1) *Encryption using Javax.crypto APIs*

The javax.crypto package defines a lot of classes and interfaces for various cryptographic operations. The main class is Cipher, which is used to encrypt and decrypt data.

All rights reserved by www.ijsrd.com

CipherInputStream and CipherOutputStream are utility classes that use a Cipher object to encrypt or decrypt streaming data.

Some of important classes of javax.crypto introduced in following section.

2) Cipher:

This class provides the functionality of a cryptographic cipher for encryption and decryption. It forms the core of the Java Cryptographic Extension (JCE) framework. In order to create a Cipher object, the application calls the Cipher's getInstance method, and passes the name of the requested transformation to it. Optionally, the name of a provider may be specified.

A transformation is of the form:

- "algorithm/mode/padding" or
- "algorithm"

(in the latter case, provider-specific default values for the mode and padding scheme are used). For example, the following is a valid transformation:

```
Cipher c =
Cipher.getInstance("DES/CBC/PKCS5Padding");
```

3) KeyGenerator:

This class provides the functionality of a secret (symmetric) key generator.

Key generators are constructed using one of the getInstance class methods of this class. There are two ways to generate a key: in an algorithm-independent manner, and in an algorithm-specific manner. The only difference between the two is the initialization of the object:

Every implementation of the Java platform is required to support the following standard KeyGenerator algorithms with the key sizes in parentheses:

- AES (128)
- DES (56)
- DESede (168)
- HmacSHA1
- HmacSHA256

4) SecretKey:

This interface contains no methods or constants. Its only purpose is to group (and provide type safety for) secret keys. Provider implementations of this interface must overwrite the equals and hashCode methods inherited from java.lang.Object, so that secret keys are compared based on their underlying key material and not based on reference.

Keys that implement this interface return the string RAW as their encoding format, and return the raw key bytes as the result of a getEncoded method call.

(The getFormat and getEncoded methods are inherited from the java.security.Key parent interface.)

5) SecretKeySpec:

This class specifies a secret key in a provider-independent fashion. It can be used to construct a SecretKey from a byte array, without having to go through a (provider based) SecretKeyFactory. This class is only useful for raw secret keys that can be represented as a byte array and have no key parameters associated with them, e.g., DES or Triple DES keys.

III. CONCLUSION

This paper presents general overview of steganography and cryptography with the different types of steganography. Also the methods discussed various efficiency aspects of LSB methods of steganography. The paper also provides summary of javax.crypto APIs which are important for the purpose of encryption and decryption using java.

REFERENCES

[1] Ali-al, H. Mohammad, A. 2010. Digital Audio Watermarking Based on the Discrete Wavelets Transform and Singular Value Decomposition, European Journal Of Scientific Research, vol 39(1), pp 231-239.
[2] Amirthanjan, R. Akila, R & Deepika chowdavarapu, P., 2010. A Comparative Analysis of Image Steganography, International Journal of Computer Application, 2(3), pp.2-10.
[3] Arnold, M. 2000. Audio watermarking: Features, applications and algorithms, Proceeding of the IEEE International Conference on Multimedia and Expo, pp 1013-1016.
[4] Bandyopadhyay, S.K., 2010. An Alternative Approach of Steganography Using Reference Image. International Journal of Advancements in technology, 1(1), pp.05-11.
[5] Siridevi,R. Damodaram, A. &Narasingham, S.,2009. Efficient Method of Audio Steganography by Modified LSB Algorithm and Strong Encryption Key with Enhanced Security. Journal of theoretical and applied information technology, 5(2), pp.25-31.
[6] ZaidoonKh, A. Zaidan,A.A. Zaidan, B.B & Alanazi.H.O., 2010. Overview: main fundamentals for steganography. Journal of Computing, 2(3), pp.40-43.
[7] C.P.Sumathi et al(2013) A Study of Various Steganographic Techniques Used for Information Hiding, International Journal of Computer Science & Engineering Survey (IJCSES) Vol.4, No.6.
[8] MamtaJuneja (2013), An Improved LSB based Steganography Technique for RGB Color Images, 2nd International Conference on Latest Computational Technologies (ICLCT'2013) June 17-18.
[9] MasoudNosrati(2013), An introduction to steganography methods, World Applied Programming, Vol (1), No (3) pp 191-195.
[10] Shamim Ahmed Laskar(2012), High Capacity data hiding using LSB Steganography and Encryption, International Journal of Database Management Systems (IJDMS) Vol.4, No.6.
[11] https://www.scribd.com/doc/48764974/Steganography-Data-hiding-using-LSB-algorithm.
[12] A.Joseph Raphael et al(2012), Cryptography and Steganography – A Survey, Int. J. Comp. Tech. Appl., Vol 2 (3) pp 626-630
[13] Himanshu Gupta et al(2013), Enhanced Data Hiding Capacity Using LSB- Based Image Steganography Method, International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 6 pp 212-214.
[14] java.oracle.com

All rights reserved by *www.ijsrd.com*