Implementation and Analysis of Email Messages Encryption and Image Steganography Schemes for Image Authentication and Verification

Ms.B.Veera Jyothi
CBIT,
Hyderabad, India.

Dr.S.M.Verma
Rayalaseema University,
Kurnool, India.

Dr.C.Uma Shanker
Rayalaseema University,
Kurnool, India

ABSTRACT
Image Steganography is to hide messages or information within other information in such a way as to not be detectable. This makes use of the fact that there is a large amount of data being transferred every second, making it impossible to scan all the information for hidden messages. Typical cryptographic methods obscure the information, but it is still very obvious that a message is being sent. Steganography attempts to correct this flaw so an observer is unable to know if a message is being sent or not. This can be used in addition to traditional cryptographic methods, so the security will only be enhanced, assuming that the traditional methods are being used with the same rigor as before.

Steganography in images is each pixel is encoded as a series of numbers which represent the red green and blue values which make up the color for that pixel. Since a slight change in this color scheme is not detectable by the human eye, it can be used to hide information. This is usually accomplished by changing the least significant bit, or LSB, for each pixel to correspond to the bits of the hidden message

General Terms: Image Processing ; Pattern Recognition ; Network Security ; Machine Intelligence;

Keywords: cryptography; digital signature ; Image Steganography ; watermarking;

1. INTRODUCTION
Email messages are not protected as they move across Internet. Messages can be undelivered or intercepted and read by unauthorized or unintended individuals. Email can also be surreptitiously modified even forged creating the impression that a person made a statement that she did not. Ordinary Internet email does not provide techniques for assuring integrity, privacy or establishing authorship. A digital signature or digital signature scheme is a type of asymmetric cryptography used to simulate the security properties of a handwritten signature on paper. Modern steganography is the ability to hide information in an electronic source. We focus on novel data hiding techniques provided by the field of steganography to authenticate an encrypted digital signature, hidden in a digital image. There are no algorithms existing currently to secure email messages which use encryption and image steganography techniques together. In this paper we discuss the implementation of an algorithm which uses these two techniques together and analyze the performance of the system.

2. BACKGROUND
Early attempts at image authentication are mainly based on fragile watermarking schemes where the multimedia data is treated as digital bits and signed using traditional cryptographic techniques. The signature information is inserted into spatial domain DCT Domain, and Wavelet Domain.

Signature insertion algorithms cannot tolerate any change, however minor; otherwise the image gets rendered inauthentic. As a result, even the slightest use of JPEG lossy compression results into an inauthentic image. To use content based authentication scheme that can tolerate minor modification such as JPEG compression. The goal in the content based authentication is to verify the content of an image, not its representation.

Steganography is applicable to, but not limited to, the following areas.

1) Confidential communication and secret data storing

2) Protection of data alteration

3) Access control system for digital content distribution

4) Media Database systems

Steganography is the dark cousin of cryptography, the use of codes. While cryptography provides privacy, steganography is intended to provide secrecy. Privacy is what you need when you use your credit card on the Internet -- you don't want your number revealed to the public. For this, you use cryptography, and send a coded pile of gibberish that only the web site can decipher. Though your code may be unbreakable, any hacker can look and see you've sent a message. For true secrecy, you don't want anyone to know you're sending a message at all.

Early steganography was messy. Before phones, before mail, before horses, messages were sent on foot. If you wanted to hide a message, you had two choices: have the messenger memorize it, or hide it on the messenger. In fact, the Chinese wrote messages on silk and encased them in balls of wax. The wax ball, "la wan," could then be hidden *in* the messenger.

Herodotus, an entertaining but less than reliable Greek historian, reports a more ingenious method. Histaeus, ruler of Miletus, wanted to send a message to his friend Aristagorus,

urging revolt against the Persians. Histaeus shaved the head of his most trusted slave, and then tattooed a message on the slave's scalp. After the hair grew back, the slave was sent to Aristagorus with the message safely hidden.

Later in Herodotus' histories, the Spartans received word that Xerxes was preparing to invade Greece. Their informant, Demeratus, was a Greek in exile in Persia. Fearing discovery, Demeratus wrote his message on the wood backing of a wax tablet. He then hid the message underneath a fresh layer of wax. The apparently blank tablet sailed easily past sentries on the road.

A more subtle method, nearly as old, is to use invisible ink. Described as early as the first century AD, invisible inks were commonly used for serious communications until WWII. The simplest are organic compounds, such as lemon juice, milk, or urine, all of which turn dark when held over a flame. In 1641, Bishop John Wilkins suggested onion juice, alum, ammonia salts, and for glow-in-the dark writing the "distilled Juice of Glowworms." Modern invisible inks fluoresce under ultraviolet light and are used as anti-counterfeit devices. For example, "VOID" is printed on checks and other official documents in an ink that appears under the strong ultraviolet light used for photocopies.

During the American revolution, both sides made extensive use of chemical inks that required special developers to detect, though the British had discovered the American formula by 1777. Throughout World War II, the two sides raced to create new secret inks and to find developers for the ink of the enemy. In the end, though, the volume of communications rendered invisible ink impractical.

With the advent of photography, microfilm was created as a way to store a large amount of information in a very small space. In both world wars, the Germans used "microdots" to hide information, a technique which J. Edgar Hoover called "the enemy's masterpiece of espionage." A secret message was photographed, reduced to the size of a printed period, and then pasted into an innocuous cover message, magazine, or newspaper. The Americans caught on only when tipped by a double agent: "Watch out for the dots -- lots and lots of little dots."

Modern updates to these ideas use computers to make the hidden message even less noticeable. For example, laser printers can adjust spacing of lines and characters by less than 1/300th of an inch. To hide a zero, leave a standard space, and to hide a one leave 1/300th of an inch more than usual. Varying the spacing over an entire document can hide a short binary message that is undetectable by the human eye. Even better, this sort of trick stands up well to repeated photocopying.

All of these approaches to steganography have one thing in common -- they hide the secret message in the physical object which is sent. The cover message is merely a distraction, and could be anything. Of the innumerable variations on this theme, none will work for electronic communications because only the pure information of the cover message is transmitted. Nevertheless, there is plenty of room to hide secret information in a not-so-secret message. It just takes ingenuity.

The majority of today's steganographic systems uses multimedia objects like image, audio, video etc as cover media because people often transmit digital pictures over email and other Internet communication[3]. In modern approach, depending on the nature of cover object, steganography can be divided into five types:

• Text Steganography

• Image Steganography

• Audio Steganography

• Video Steganography

• Protocol Steganography

2.1 Text Steganography

Since everyone can read, encoding text in neutral sentences is doubtfully effective. But taking the first letter of each word of the previous sentence, you will see that it is possible and not very difficult. Hiding information in plain text can be done in many different ways [4].

Many techniques involve the modification of the layout of a text, rules like using every n-th character or the altering of the amount of white space after lines or between words [24]. The last technique was successfully used in practice and even after a text has been printed and copied on paper for ten times, the secret message could still be retrieved. Another possible way of storing a secret inside a text is using a publicly available cover source, a book or a newspaper, and using a code which consists for example of a combination of a page number, a line number and a character number. This way, no information stored inside the cover source will lead to the hidden message. Discovering it relies solely on gaining knowledge of the secret key.

2.2 Image Steganography

To hide information, straight message insertion may encode every bit of information in the image or selectively embed the message in "noisy" areas that draw less attention—those areas where there is a great deal of natural color variation. The message may also be scattered randomly throughout the image. A number of ways exist to hide information in digital media. Common approaches include

• Least significant bit insertion

• Masking and filtering

• Redundant Pattern Encoding

• Encrypt and Scatter

• Algorithms and transformations

Each of these techniques can be applied, with varying degrees of success.

2.3 Least significant bit insertion

Least significant bit (LSB) insertion is a common and simple approach to embed information in an image file. In this method the LSB of a byte is replaced with an M's bit. This technique works well for image, audio and video steganography. To the human eye, the resulting image will look identical to the

cover object. For example, if we consider image steganography then the letter A can be hidden in three pixels (assuming no compression). The original raster data for 3 pixels (9 bytes) may be

(00100111 11101001 11001000)

(00100111 11001000 11101001)

(11001000 00100111 11101001)

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

(00100111 11101000 11001000)

(00100110 11001000 11101000)

(11001000 00100111 11101001)

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. The resultant image for the above data insertion and the original cover image are given below.

Fig.2. 1: The cover image

Fig. 2.2: The stego-image (after A is inserted)

2.4 Masking and filtering

Masking and filtering techniques are mostly used on 24 bit and grey scale images. They hide info in a way *Masking and filtering* Masking and filtering techniques are mostly used on 24 bit and grey scale images. They hide info in a way similar to watermarks on actual paper and are sometimes used as digital watermarks. Masking images entails changing the luminance of the masked area. The smaller the luminance change, the less of a chance that it can be detected. Observe that the luminance in Figure 2 is at 15% in the mask region if it was decreased then it would be nearly invisible .

Fig. 3: Masking

Masking is more robust than LSB insertion with respect to compression, cropping, and some image processing. Masking techniques embed information in significant areas so that the hidden message is more integral to the cover image than just hiding it in the "noise" level. This makes it more suitable than LSB with, for instance, lossy JPEG images.

Redundant Pattern Encoding

Patchwork and other similar tools do redundant pattern encoding, which is a sort of spread spectrum technique. It works by scattering the message throughout the picture. This makes the image more resistant to cropping and rotation. Smaller secret images work better to increase the redundancy embedded in the cover image, and thus make it easier to recover if the stego-image is manipulated .

2.5 Encrypt and Scatter

The Encrypt and Scatter technique tries to emulate white noise. It is mostly used in image steganography. White Noise Storm is one such program that employs spread spectrum and frequency hopping. It does this by scattering the message throughout an image on eight channels within a random number that is generated by the previous window size and data channel. The channels then swap rotate, and interlace amongst each other. Each channel represents one bit and as a result there are many unaffected bits in each channel. This technique is a lot harder to extract a message out of than an LSB scheme because to decode you must first detect that a hidden image exists and extract the bit pattern from the file. While that is true for any stego-image you will also need the algorithm and stego key to decode the bit pattern, both of which are not required to recover a message from LSB. Some people prefer this method due to the considerable amount of extra effort that someone without the algorithm and stego-key would have to go through to extract the message. Even though White Noise Storm provides extra security against message extraction it is just as susceptible as straight LSB to image degradation due to image processing [1, 5].

2.6 Algorithms and transformations

LSB modification technique for images does hold good if any kind of compression is done on the resultant stego-image e.g. JPEG, GIF etc JPEG images use the discrete cosine transform to achieve compression. DCT is a lossy compression transform because the cosine values cannot be calculated exactly, and repeated calculations using limited precision numbers introduce rounding errors into the final result. Variances between original data values and restored data values depend on the method used to calculate DCT.

3. PROPOSED SYSTEM

Proposed system contains two distinct algorithms for encoding and decoding:

3.1 Encoding of email message

The coding algorithm is composed of two steps which are the encryption and the data hiding step. For each block composed of n pixels of an image of N pixels, we apply the AES encryption algorithm by block. During the data hiding step, in each cipher-text we modify only one bit of one encrypted pixel of the image. We used bit substitution-based data hiding method in order to embed the bits of the hidden message. For each block, the secret key k is used as the seed of the pseudo-random number generator (PRNG) to substitute the bit of a pixel with the bit to hidden. At the end of the coding process we get a marked encrypted image. Since we embed 1 bit in each block of n pixels, the embedding factor is equal to 1/n bit per pixel.

3.2 Decoding of email message

The decoding algorithm is also composed of two steps which are the extraction of the message and the decryption removing. The extraction of the message is very simple: it is just enough to read the bits of the pixels we have marked by using the secret key k and the same PRNG. But after the extraction, each marked cipher-text is still marked. The problem is then to decrypt the marked encrypted image. The decryption removing is done by analyzing the local standard deviation during the decryption of the marked encrypted images. For each marked cipher-text, we apply the decryption function for the two possible values of the hidden bit (0 or 1) and we analyze the local standard deviation of the two decrypted blocks. In the encrypted image, the entropy must be maximal and greater than the original one.

4. EXPERIMENTAL ANALYSIS

Digitally signs a message, hides it in a randomly generated image or a user selected image and sends the image instead of the message. Encrypting the message before hiding is also an option. Only a person with valid password and this software(Algorithm) can extract the message.

4.1 Sender Side:

1. The message is first hashed using SHA256 Double hash Algorithm.
2. A digital signature is generated using RSA with senders private signing key.
3. The signature is then appended to the message.
4. The user is prompted for a password.
5. The password is hashed and the hash value is used to seed a PRNG.
6. The output of PRNG is used to select a random bit from a random pixel.
7. The message bit is then written into that bit by a simple XOR operation.
8. The image is then sent as an attachment instead of the message.

4.2 Receiver Side:

1. The image is downloaded from the mail server.
2. The senders Public Verification key is obtained from the key server.
3. The user is prompted for password.
4. The password is hashed and the hash value is used to seed a PRNG.
5. The output of PRNG is used to extract the message from the image.
6. The message is hashed using SHA256
7. Double hash and the digital signature is verified using senders public verification key.

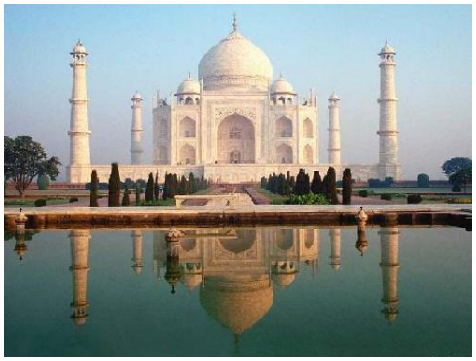

Figure 4.1: Original image

Considering the scenario where Sender wants to send a message to Receiver hidden in an image so that only Receiver can extract the message from the image.

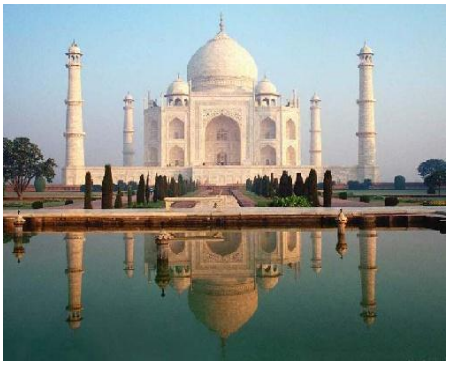

Figure 4.2: Image after message is hidden

If the Sender used to choose the default image ,which in this application is Randomly generated image

Figure 4.3: Random image after message is hidden

4.3 Sender's gmail

Sent items folder

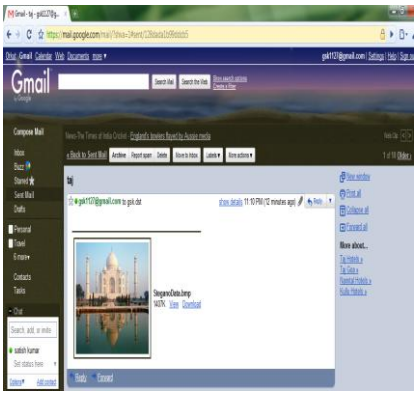

Figure 4.4: Sender's mail box

As it can be seen, only image was sent not the message

`Receiver's Inbox

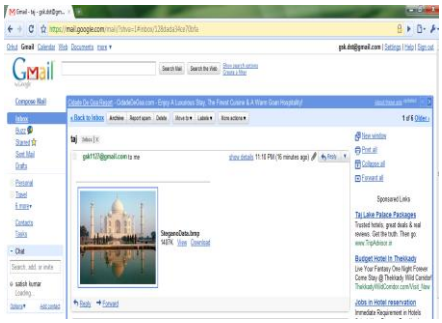

Figure 4.5: Receiver's inbox

As it can be seen, receiver's inbox has only image not message.

4.4 Email read by Receiver after running application

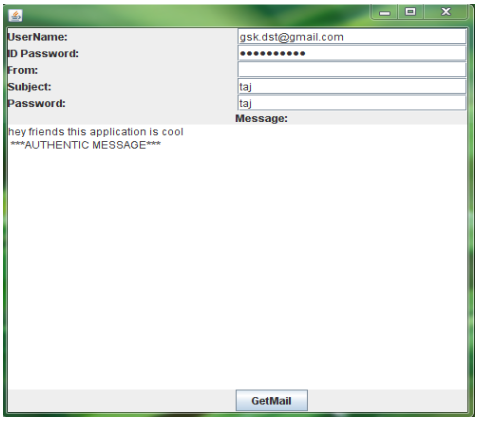

Figure 4.6: Retrieving the message from the image

The retrieved message is displayed in the message box.

4.5 Security

If some one hacks into receivers mail and tries to extract message from image without knowing the correct password

4.6 When Authentication failed shows error message

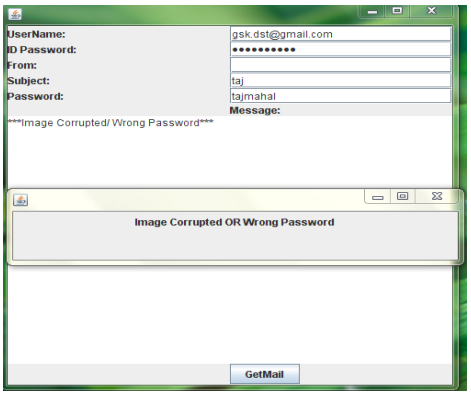

Figure 4.7: Error message

When someone tries to hack the emails and tries to extract the message without knowing the correct password then a message is displayed showing that image corrupted or wrong password

5. CONCLUSIONS

As Steganography only hides the data into another medium (for e.g. digital image), once noticed the hidden data could be retrieved with known algorithms. Thus Steganography must be used in conjunction with Cryptography to combine data obfuscation and data hiding properties to make Message Communication between Sender and Receiver even more secure. However, Steganography alone gives the clear advantage over cryptography such that messages do not attract attention to themselves, to messengers, or to recipients.

There are several algorithms for hiding/retrieving the data from the target image but most of them suffer from some amount of loss of hidden data while retrieving it. All three research papers propose techniques to improve the process of hiding data in order to better retrieve it suffering minimal data loss.

6. FUTURE ENHANCEMENTS

Following are the recommendations. Considering the incredible amount of research work currently in progress to make Steganography as a secure transmission medium, makes us believe that in future we will be able to apply Steganography to number of scenarios such as confidential image transmission over the network.

Currently, the better known techniques of Steganalysis only makes it hard to detect the possibility of hidden data in an anonymous image currently being transmitted over the network and thus we must improvise Steganalysis process such that it will become much easier to detect even small messages within an image

Finally, the fact that adding hidden data adds random noise to the target image making it hard to recover the hidden data, so it follows that a properly tuned noise detection algorithm could recognize whether or not a picture had steganographic data or not.

7. ACKNOWLEDGEMENT

This is to acknowledge that sincere thanks to my guides Dr.S.M.Verma. Dr.C.UmaShanker Dr.ChennakeshavaRao, Principal, CBIT and management CBIT And all others who assisted me in bringing out this work successfully.

8. REFERENCES

[1] Ping Wah Wong; Memon, N., "Secret and public key image watermarking schemes for image authentication and ownership verification", Image Processing, IEEE Transactions on, Volume 10, Issue 10, Page(s):2-8

[2] Dekun Zou, Chai Wah Wu, GuorongXuan, Yun Q. Shi, "A content-based image authentication system with lossless data hiding", Multimedia and Expo, International Conference on, Volume: 2, Page(s):1-3

[3] W. Puech, M. Chaumont, and O. Strauss, "A reversible data hiding method for encrypted images", Security, Forensics, Steganography, and Watermarking of Multimedia Contents X. Edited by Delp, Edward J., III; Wong, Ping Wah; Dittmann, Jana; Memon, Nasir D. Proceedings of the SPIE, Volume 6819, (2008), Pages(s): 2-5

[4] Debnath Bhattacharyya1, Poulami Das1, Samir Kumar Bandyopadhyay, and Tai-hoon Kim **"**Text Steganography: A Novel Approach" (2009)

[5] Wayner, Peter (2002). "Disappearing cryptography: information hiding: steganography & watermarking". Amsterdam: MK/Morgan Kaufmann Publishers. ISBN 1-55860-769-2.

[6] Wayner, Peter (2009). "Disappearing cryptography 3rd Edition: information hiding: steganography & watermarking". Amsterdam: MK/Morgan Kaufmann Publishers. ISBN 978-0123744791.

[7] Petitcolas, Fabian A.P.; Katzenbeisser, Stefan (2000). "Information Hiding Techniques for Steganography and Digital Watermarking". Artech House Publishers. ISBN 1-58053-035-4.

[8] Johnson, Neil; Duric, Zoran; Jajodia, Sushil (2001). *Information hiding: steganography and watermarking: attacks and countermeasures*. Springer. ISBN 978-0-7923-7204-2.

[9] Westfeld, A. (2001). F5-a steganographic algorithm: High capacity despite better steganalysis. In Proc. 4th Int'l Workshop Information Hiding, pages 289–302.

[10]. W. Brown and B.J. Shepherd, Graphics File Formats: Reference and Guide, Manning Publications, Greenwich, Conn, 1995.

[11]. E. Koch, J. Rindfrey, and J. Zhao, "Copyright Protection for Multimedia Data," Proc. Int'l Conf. Digital Media and Electronic Publishing, Leeds, UK 1994.