Covert Communication Techniques in the Digital World

Aisha Fernandes a, Dr. W. Jeberson b, *

Dept of Computer Science, Sam Higginbottom Institute of Agriculture, Technology & Sciences, Allahabad – 211007

Abstract

The boom of covert communication coincides with the appearance and growth of the Internet. The rapid spread of computer networks and shift to digitization of media has created a very favourable environment for covert communication. In this paper, some Covert communication techniques have been described. Covert or secure communication includes methods such as cryptography, steganography and watermarking. The process, some commonly used algorithms and potential applications of each of these techniques have been discussed below. Finally, the proposed method integrates the concepts of cryptography and steganography for covert communication.

Keywords: covert, cover, cryptography, steganography, watermarking

1. Introduction

Data security is the practice of keeping data protected from corruption and unauthorized access. Data is the raw form of information stored as columns and rows in databases, network servers and personal computers. This may be a wide range of information from personal files and intellectual property to market analytics and details intended to be top secret. To ensure that the data is transmitted securely, the data could either be transformed in a gibberish form or the highly secure data could be embedded into carrier files. The former method does not conceal the existence of data but the later is useful to hide the fact that there was data transmitted. Owing to the applications of covert communications, with regard to digital rights management, unobtrusive and anonymous communications etc, the fields of cryptography, steganography and digital watermarking have triggered interest among the corporate world too. Covert communication is broadly divided [1] as shown in Fig. 1.

2. Cryptography

Electronic communication is increasingly susceptible to eavesdropping and malicious interventions [2]. Cryptography has been the most common method of protecting digital content. Here the messages are encrypted so that only the rightful recipient could read them and verify their integrity and authenticity. The content was encrypted prior to transmit, and a decryption key was provided only to those who have purchased legitimate copies of the content or are authorized to view the content [2]. The encrypted file could then be made available via the Internet, but would be useless to a pirate without an appropriate key. Refer to figure 2 below.

* Corresponding author. aishafs@yahoo.com, jeberson@rediffmail.com.
2.1. Applications of Cryptography

- Secrecy in transmission
- Secrecy in storage
- Integrity in transmission
- Integrity in storage
- Authentication of identity
- Credential systems

2.2. Some Cryptographic Algorithms

2.2.1. DATA ENCRYPTION STANDARD (DES)

DES was the first encryption standard designed in 1973[3]. It is a block cipher which encrypts 64 bit plaintext at a time and produces a 64 bit ciphertext using a 56 bit key. It is a symmetric key algorithm which means that the same key is used for both encryption and decryption. DES can operate in CBC, ECB, CFB and OFB modes. DES has 16 rounds which mean a total of 16 processing steps are being applied on the input plaintext to produce cipher text. First, 64 bit data is passed through the initial permutation phase and then 16 rounds of processing takes place and finally the last step of final permutation is carried out on the input plain text which results in 64 bit cipher text [4].

The drawback of this algorithm was that it was easily prone to Brute Force Attack in which the hacker attempted to break the key by applying all possible combinations. In DES there were only 2^56 possible combinations which made it easy to crack. So DES is not so secure.

2.2.2. TRIPLE DES

The triple DES (3DES) algorithm was needed as a replacement for DES due to advances in key searching. TDES uses three rounds of DES encryption and has a key length of 168 bits (56 * 3). Either two or three 56 bit keys are used in the sequence Encrypt-Decrypt-Encrypt (EDE)[4].

2.2.3. AES

AES is a symmetric key algorithm, and was based on the design principle of a substitution-permutation network. AES had a fixed block of 128 bits, and a key size of 128, 192, or 256 bits [3].

AES operated on a 4x4 column-major order matrix of bytes. The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text.

2.2.4. Blowfish

Blowfish had a 64-bit block size and a variable key length from 32 bits up to 448 bits. It was a 16-round Feistel cipher and used large key-dependent S boxes. The algorithm kept two subkey arrays and four 256-entry S-boxes. The S-boxes accepted 8-bit input and in every round, and after the final round, each half of the data block was XORed with one of the two remaining unused P-entries. The F function split the 32-bit input into four eight-bit quarters, and used the quarters as input to the S-boxes. The outputs were added modulo 2^32 and XORed to produce the final 32-bit output [5].

Decryption was exactly the same as encryption, except that P1, P2,..., P18 were used in the reverse order.

2.2.5. RSA

RSA is a public key cryptography algorithm and was based on the difficulty of factoring large integers [6]. RSA involved a public key and a private key. The public key was known to everyone and was used for encrypting messages. Messages encrypted with the public key could only be decrypted using the private key.

The RSA algorithm involved three principle steps: key generation, encryption and decryption [6].

1. Choose two distinct prime numbers $p$ and $q$.
2. Compute $n = pq$.
3. Compute $\phi(n) = \phi(p)\phi(q) = (p - 1)(q - 1)$
4. Choose an integer $e$ such that $1 < e < \phi(n)$ and $\gcd(e, \phi(n)) = 1$
5. Determine $d$ as $d^{-1} \equiv e \pmod{\phi(n)}$.

$$e \equiv m^e \pmod{n}.$$  \hspace{2cm} (mod\ n).

$$m \equiv c^d \pmod{n}.$$  \hspace{2cm} (mod\ n).

3. Steganography

Steganography is the art and science of writing secret data in such a way that no one, except the intended receiver, knows of the existence of the data [7]. It utilized the typical digital media such as text, image, audio, video, and multimedia as a carrier or so called cover for secret data. Refer to Fig. 3 below.

3.1 Applications of Steganography

Steganography plays an important role in protecting the security of highly sensitive documents over the Internet in this era of terabit networks. It had various useful applications. However, like any other science it could be used for ill intentions. It had been propelled to the forefront of current security techniques by the remarkable growth in computational power, the increase in security awareness.
by, e.g., individuals, groups, agencies, government and through intellectual pursuit [8]. Some of its most prominent applications are

- Patient data concealment in digital images [8]
- Digital evidence analysis to combat child pornography
- Military and intelligent agencies that need unobtrusive communication.

Some of it's most prominent applications are

- Patient data concealment in digital images [8]
- Digital evidence analysis to combat child pornography
- Military and intelligent agencies that need unobtrusive communication.

3.2. Steganography Methods

Some of the methods used to embed secret data within a cover are

3.2.1. Least significant bit replacement method (LSB)

Least significant bit (LSB) insertion was a common, simple approach to embedding information in a cover image [9]. The least significant bit of some or all of the bytes inside an image was changed to a bit of the secret message [9]. When using a 24-bit image, a bit of each of the red, green and blue colour components could be used, since they are each represented by a byte.

In the extraction process, the LSB of the selected cover elements could be extracted and lined up to reconstruct the secret message. However it was vulnerable to steganalysis which is based on statistical analysis [10].

3.2.2. Transform domain methods

Embedded secret information in the transform space of a signal. E.g. in the frequency domain. These methods were more robust to various kinds of signal processing [9]. There existed many transform domain variations that included discrete cosine transformation (DCT), discrete wavelet transforms (DWT), etc. A trade off existed between the amount of secret information embedded and the robustness obtained.

The advantages of transform domain techniques over spatial domain techniques were their high ability to tolerate noises and some signal processing operations but on the other hand they have a low embedding capacity, were computationally complex and hence slower [10].

3.2.3. Spread spectrum and error control methods

In spread spectrum techniques, hidden data was spread throughout the cover-image making it harder to detect the process of spreading the bandwidth of a narrowband signal across a wide band of frequencies. After spreading, the energy of the narrowband signal in any one frequency band was low and therefore difficult to detect [9]. In spread spectrum image steganography the message was embedded in noise and then combined with the cover image to produce the stego image.

3.2.4. Echo hiding

Attempted to hide information in a discrete signal f(t) by introducing an echo f(t-\Delta t) in the stego signal c(t).

\[ c(t) = f(t) + \alpha f(t-\Delta t). \] (1)

Information was encoded in the signal by modifying the delay \Delta t between the signal and the echo. In the encoding step the sender chose the delay time \Delta t or \Delta t' in such a way that the echo signal was not audible for a human observer [9].

4. Digital Watermarking

The Internet is an excellent distribution system for digital media because it is inexpensive, eliminates warehousing and stock, and delivery is almost instantaneous. Content owners (especially large Hollywood studios and music labels) saw a high risk of piracy [11]. Watermarking had the potential to handle this risk using copyright marking - hiding copyright notices and serial numbers in the audio or video in such a way that they are difficult for pirates to remove.

Watermarks are imperceptible, inseparable from the cover in which they are embedded and undergo the same transformations as the cover [9]. A block diagram of digital watermarking is as shown below in figure 4.

![Figure 4. Working of digital watermarking](image)


- Broadcast monitoring
- Owner identification
- Proof of ownership
- Transaction tracking
- Authentication
- Copy control

4.2. Digital Watermarking Methods

Broadly divided into spatial domain methods and frequency domain methods.
4.2.1. LSB substitution

It is a spatial domain technique. Spatial embedding inserted message into image pixels. Here the LSB of the cover image in every pixel was replaced with the MSB of another image, which would be embedded in a secret / non perceptible way in the cover image. Let \( p_j \) be the \( j \)th bit of the watermark. Let \( w_i \) be the \( j \)th pixel of a cover image. LSB substitution simply converted the least significant bit of \( p_j \) to \( w_i \). Since only the LSBs were changed, the modification did not cause much perceptible noise to the cover.

4.2.2. The Discrete Cosine Transform (DCT) Technique

It is a frequency domain technique. It embeds a message by modifying the transform coefficients of the cover message [12]. DCT transformation separated the image into spectral sub-bands depending on the importance with respect to the image's visual quality. It could separate the images. These appeared in the upper left corner of the DCT, so cause local distortion along with edges. Middle frequencies modification cannot affect the image quality, so transform coefficients set threshold value in this area.

The JPEG compression technique utilized this property to separate and remove insignificant high frequency components in images. DCT was faster and the complexity of it was \( O(n \log n) \) operations.

4.2.3. The Discrete Wavelet Transform (DWT) Technique

Watermarking in the wavelet transform domain requires embedding watermark in the subbands of the cover image. There were four subbands created at the end of each level of image wavelet transformation: they were Low-Low pass subband (LL), High-Low (horizontal) subband (HL), Low-High (vertical) subband (LH) and High-High (diagonal) pass subband (HH) [13]. Subsequent level of wavelet transformation was applied to the LL subband of the previous one.

4.2.4. Spread Spectrum Watermarking

The watermark is transformed into a bit string \( b_1 b_2 \ldots b_{64} \). For each bit \( b_i \), a pseudorandom matrix \( R_i \) of integers \( [1-1] \) is generated. 64 different matrices consisting pseudorandomly of \( [1 -1] \) matrices depend on the bi of the watermark. A matrix \( +R_i \) was used if \( b_i \) represents a 0, a matrix \(-R_i\) was used if \( b_i \) represents a 1. Sum of all random patterns \( R_i \) defines the watermark \( W: W = \) watermarked image \( I \) [14].

\[
IW = I + kW
\]

5. Comparison and Analysis

The various algorithms discussed above were implemented using Matlab and Java on a Pentium IV, 2 GHz machine. The following observations were being made. The comparison table (Table 2) of commonly used encryption algorithms showed the supremacy of Blowfish algorithm over others based on key size and security. Some of the most commonly used cryptographic algorithms are DES, RSA, Blowfish, AES, RC4 / RC5 [15]. While DES, AES, Blowfish are symmetric algorithms, RSA is an asymmetric algorithm. It uses the concept of private-public key for encryption-decryption.

It was seen that among the symmetric algorithms, Blowfish had the best performance in terms of processing time. AES consumed more resources when data block size was large and thus required more processing power [16]. TripleDES required more time than DES because of its triple phase encryption. It was also noticed that symmetric key algorithms ran faster and required less memory than asymmetric key algorithms such as RSA. Refer to Table 1 below.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Execution time (in seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blowfish</td>
<td>2.20</td>
</tr>
<tr>
<td>DES</td>
<td>3.58</td>
</tr>
<tr>
<td>AES</td>
<td>4.15</td>
</tr>
<tr>
<td>3DES</td>
<td>7.58</td>
</tr>
<tr>
<td>RSA</td>
<td>6.10</td>
</tr>
</tbody>
</table>

Table 1. Processing time of encryption algorithms (36 mb data)

Some of the embedding, the Peak signal to noise ratio (PSNR) was calculated. The obtained results are given in Table 3 below.

Steganographic methods are broadly divided into spatial domain and transform domain methods. Spatial domain methods like LSB allowed a large payload but were prone to statistical attacks and were not robust against lossy compression, noise, cropping and rotation. The DCT (transform domain) method provided resistance against visual attacks and some statistical analysis methods. Steganography using a Wavelet transformation was found to preserve good quality with little perceptual artefacts [11]. The analysis of steganographic methods is tabulated in Table 3 below.

<table>
<thead>
<tr>
<th>Features</th>
<th>LSB</th>
<th>Transform domain</th>
<th>Spread spectrum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imperceptibility</td>
<td>Moderate</td>
<td>Good</td>
<td>Very good</td>
</tr>
<tr>
<td>Robustness</td>
<td>Poor</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>Capacity</td>
<td>High</td>
<td>medium</td>
<td>medium</td>
</tr>
</tbody>
</table>

It was also seen that considering a test cover image of size 512*512(Lena.jpeg), and a secret image of size 256*256, and using the simple LSB technique for embedding, the Peak signal to noise ratio (PSNR) was
found to be 39.06 db. Using DCT method, PSNR was found to be 40 db.

In watermarking too, transform domain methods were better than spatial domain. LSB (spatial domain) embedded watermarks lack even a minimal level of robustness and can be easily removed without degrading the image. Its main merit was that it had a high information embedding capacity. Embedding in the DCT domain (transform domain), the Lena.jpeg image was found to be resistant to JPEG compression as well as to significant amount of random noise. The wavelet domain proved to be highly resistant to both, compression and noise, with minimal amount of visual degradation. Refer to table 4 below.

<table>
<thead>
<tr>
<th>Table 4. Comparison of watermarking methods</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Robustness</td>
</tr>
<tr>
<td>Embedding capacity</td>
</tr>
<tr>
<td>Resistance to noise</td>
</tr>
</tbody>
</table>

After the various implementations and comparisons the following table (table 5) has been tabulated.

**6. Proposed methodology**

<table>
<thead>
<tr>
<th>Table 5. Comparison of covert communication techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Carriers</td>
</tr>
<tr>
<td>Secret data</td>
</tr>
<tr>
<td>Aim</td>
</tr>
<tr>
<td>Use of key</td>
</tr>
<tr>
<td>Features</td>
</tr>
<tr>
<td>Flexibility</td>
</tr>
<tr>
<td>Output file</td>
</tr>
<tr>
<td>Type of attack</td>
</tr>
<tr>
<td>Confidentiality</td>
</tr>
</tbody>
</table>

**7. Conclusion**

At times public communication demands the need of covert communication—a mechanism to communicate privately in a public environment. This document work presents a background on the various techniques that can be used to achieve covert communications. Cryptography deals with protecting the contents of a message by converting the data into an unreadable format using a key. Steganography studies ways to make communication invisible by hiding secrets in innocuous covers, whereas watermarking originated from the need for copyright protection of digital media. Steganography urges that the cover (image, text, audio or video) must be carefully selected and steganography methods usually struggle with achieving a high embedding rate.

As cryptography only transforms the data into gibberish form and transmits it, it tends to attract the intruders attention. On the other hand steganography made the task of the intruder even more difficult by hiding the very existence of the secret data. However, steganography could be a failure even if the presence of secret data is detected. Thus to enhance the security of communications over overt channels, it is proposed (Fig. 5) to first encrypt the secret data using a suitable encryption algorithm, compress it and then embed it into a cover using an appropriate steganographic method.

![Fig. 5. Integration of Cryptography and Steganography](image-url)

The main driving force for watermarking is concern over protecting copyright: as audio, video, and other works become available in digital form, the ease with which perfect copies can be made may lead to large scale unauthorized copying, and this is of great concern to the music, film, book, and software publishing industries. Each technique is unique in its own way, which might be suitable for different applications. There is no single algorithm that satisfies all of the requirements. Thus a trade-off exists in most cases, depending on which requirements are more important for the specific application.

This work also explains a comparison of some of the common algorithms and mentions some of the applications of cryptography, steganography and digital watermarking.
Finally, to enhance the security of communications over overt channels, a method integrating the concepts of cryptography and steganography was proposed.

References