Combination of cryptography and steganography

Combination of Cryptography and Steganography for secure communication is a tool that combines both Cryptography methods and Steganography techniques for secure communication. The application is a cross-platform tool that can be effectively hide a message inside a digital video file. In the field of data communication, security has the top priority. Cryptography is one of the technique for secure plain text messages. Cryptography makes the necessary elements for secure communication namely privacy, confidentiality, key exchange and authentication but reveals the fact that communication is happening. Steganography takes cryptography a step farther by hiding the existence of the information.

Steganography plays a vital role in the data communication field in the future primarily in computer security and privacy on open systems such as internet.

The figure below[1], gives different applications of Steganography. Protection against detection (Data hiding) and protection against removal (Document Marking) are two major areas Steganographic methods are used. Steganographic Data hiding algorithms allows user to hide large amounts of information within digital files like Image, audio and video files. These forms of Steganography often used in conjunction with cryptography adding layers of security.

The Other major area of Steganography is document marking where the message to be inserted is used to assert copyright over a document. This can be further divided into watermarking and fingerprinting. Copyright abuse is the motivating factor in developing new document marking technologies

like digital watermarking and digital fingerprinting. “ Digital Watermarking is a way to hide a secret or personal message to protect a product’s copyright or to demonstrate data integrity”. “ Digital FingerprintingA is an emerging technology to protect multimedia from unauthorized redistribution. It embeds a unique ID into each user’s copy, which can be extracted to help identify culprits when an unauthorized leak is found” [2].

Neither Cryptography nor Steganography is a turnkey solution to privacy of open systems. To add multiple layers of security it is always a good practice to use both Cryptography and Steganography together.

While performing Cryptography we should know two formulas (Cipher formula, Decipher formula).

Rationale:

To make the communication more secure in this application we are implementing three layers of security like Steganography, Cryptography and Compression. The application first compresses the word document with secret message, and then encrypts the compressed file and uses the resulted file as the secret message to hide in the digital video file generating a Stego-object. The intended receiver de-embeds decrypts and decompresses the Stego-object respectively to get the hidden message. This paper also attempts to identify the requirements of a good Steganographic algorithm and briefly reflects on different types of steganalysis techniques. The application uses Tiny encryption algorithm and Discrete Cosine Transformation-Least significant bit algorithm for implementing Cryptography and Steganography respectively. The outcome of this project

is to create a cross-platform tool that can effectively hide a message (i. e. Word document) inside a digital video file. It is concerned with embedding information in a secure and robust manner.

REVIEW OF RELEVANT LITERATURE

Background to subject of study:

The idea of building this tool is to make the communication is such a way that no one can detect the message inside the stego-object. Earlier we have tools for different tools for Steganography and Cryptography. In this tool we are implementing three layers of security (Steganography, Cryptography and Compression) so that the communication can be more secure and all can be done in a single tool instead of using three different tools. Steganography has came into usage in 1990’s and it is still using in many ways by Governments, Private citizens, Business and Terrorist organizations for communication to share information and passwords.

Cryptography came into consideration in 18th century. The goal of cryptography is to make it possible for two communication entities to exchange a message in such a way that no third party can understand the message. Cryptography has been implementing from many days, in the World war Germany and USA. They have used it in sharing messages and implemented machines to implement cryptography.

Examples and critique of current research in the field:

There are many tools that are implementing Steganography now a days. The SARC (Steganography Analysis and Research Center has implemented three

tools in steganography like Steganography Analyzer Artifact Scanner, Steganography Analyzer Signature Scanner and Steganography Analyzer Real-Time Scanner. The Steganography Analyzer Artifact Scanner detect files and registry entries associated with steganography applications where as Steganography Analyzer Signature Scanner detect files containing steganography and extract the hidden information and the Steganography Analyzer Real-Time Scanner detect steganography artifacts and signatures in real-time over a network.

S-Tool is also one of the steganography tool that is using now a days. Its free to download and hides the data in an image or sound file. It compresses the data before encrypting and hides it in a image file or audio file.

PILOT RESEARCH STUDY

Hypothesis:

As sending the message will be easy for unauthorized persons to detect the information in the situations like passwords sharing and confidential information sharing. So for that we have methods that can make the information secure. By using Steganography and Cryptography techniques we can share the information more securely by hiding the information in other files. Even though we have some risks that are involved in these techniques we can be rectified to certain extend by using this tool.

Research method:

This application is implemented for secure transmission of data. In this application we have three layers of security like compression, Cryptography

and Steganography. We are using different type of algorithms in Cryptography and Steganography so that the hackers cannot identify which algorithm is supposed to be used.

In secret key Cryptography several algorithms are in operation like Data Encryption Standard (DES), Rivest Chipers (aka Ron’s Code), Advanced Encryption Standard (AES), Blowfish and CAST-128/256. In public key Cryptography we have Elliptic Curve Cryptography(ECC), ElGamal, Digital signature Algorithm (DSA), Diffie-Hellman and RSA algorithms mostly isong now a days. In hash function in Cryptography Hash of variable length(HAVAL), Tiger, RIPEMD, Secure Hash Algorithm (SHA) and Message Digest Algorith(MD) are in use. Tiny Encryption Algorithm is also one of the Feistel Cipher encryption algorithm that was designed in 1994 is used in Cryptography that uses mixed orthogonal algebraic groups like ADD, SHIFT and XOR.

In steganography we are using different Steganographic methods for hiding information into a video file like LSB (Discrete Cosine Transformation-List Significant Bit Encoding). Each frame in a video file holds a piece of secret message.

Cryptography use cipher algorithm for encryption and decryption of data. In the previous decades they were used ciphers like Scytale Transportation Cipher, Caesar Substitution Chiper, Zodiac Chiper and Vigenere Polyalphabetic Substitution.

Both Steganography and Cryptography are data security techniques, but the cryptography is implemented to data unread and Steganography in for data

unseen. Steganography can use Cryptography where as Cryptography cannot use Steganography. Steganography implemented to Cryptographic data will increase in security level.

Initial Results:

In this application i am implementing three layers of security to make the information more secure. There are no any tools that implementing all the three layers like compression, Cryptography and Steganograpy. We have tools for steganography like S-Tool and for Cryptography. By using one tool instead of using three tools will save time and money with more security. This is the basic advantage of this application. The help document will guide in Interface.

OUT LINE OF PROPOSAL

Aims of the Investigation:

The aim of this paper is to describe a method for integrating together cryptography and Steganography for secure communication using a Video file. The proposed system first compresses the secret message (i. e. word document) and then implements cryptographic algorithms to the compressed message. The resulted file is used as the secret message to be hidden in the digital video file. Once the video file is embedded with the secret message, it is sent to the intended receiver. The video file should be de-embedded, decrypted and decompressed to get the original secret message hence, adding three layers of security to the communication. I am

going to design a good Graphical User Interface (GUI) with help notes so that anyone can understand ahout the application easily.

Research Objectives:

The objective of this project is to hide secret messages (e. g. Text Phrase, word document) inside other harmless messages such as Image and Video file, in a way that does not allow any third party to even detect that there is a second secret message present in it. The application implements this by combining the Steganographic methods with Cryptographic techniques (i. e. Encryption, decryption) to make the transfer more secure. It is concerned with embedding information in a secure and robust manner. The Text file will be compressed and after that the compressed file will go on with a Cryptography and Steganography.

Methodologies:

Modules of the Application: The application has two modes of operation i. e. Sender and Receiver.

The three major modules for Sender mode of application are

Compression: The application first compresses the document to be transferred

Encryption: An Encryption algorithm encrypts the compressed file and the resulted file is used as secret message.

Embedding: The encrypted file is hidden in the Harmless Message (video file) using corresponding Steganographic algorithm, which generates a Stego Object, which is sent to the intended recipient.

The three major modules for the Receiver mode of application are

De-Embedding: The Stego Object is de-embedded generating an encrypted file.

Decryption: The encrypted file is decrypted using an the Encryption algorithm, and the resulted file is given to the compression module

De-Compression: The application then de-compresses the document and we have the Secret message.

Steganography and cryptography are closely related. “ Cryptography scrambles messages so they cannot be understood” Whereas, “ Steganography will hide the message so there is no knowledge of the existence of the message” [7]. Sending an encrypted message will arouse suspicion while an invisible message will not do so. The application developed in this project combines both sciences to produce better protection of the message. Even if the Steganography fails since the message is in encrypted form it is of no use for the third party, hence the information is secure.

In Cryptography we have used three types of methodologies and are implemented depending on the encryption Algorithm. They are Secret key Cryptography, Public key Cryptography and hash function. These 3 methods are briefly explained below.

Secret Key Cryptography:

Secret key Cryptography, also known as symmetric encryption uses same key for encryption and decryption. The sender uses key to encrypt the text and sends ciphertext to the receiver. The receiver applies the same key to decrypt the message and recover the Text.

K K

Text Ciphertext Text

E( ) D( )

K-key, E-Encryption, D-Decryption

Secret key Cryptography

The above figure shows the process of secret key cryptography. The biggest difficulty with this approach is the distribution of the key. Block ciphers can operate in one of the several modes. Electronic Codebook (ECB), Cipher Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB) are the most important modes. Data Encryption Standard (DES), Advanced Encryption Standard (AES), CAST-128/256, Rivest Ciphers (aka Ron’s Code), Blowfish are some of the Secret key cryptography algorithms [3].

Public-Key Cryptography:

K1 K2

Text Ciphertext Text

E( ) D( )

K-key, E-Encryption, D-Decryption

Public key Cryptography

Public key cryptography is a two-key cryptography system in which two keys are used in encryption and decryption for secure communication without having to share a secret key. One key is used to encrypt the text, designated the public key which can be advertised. The other key is used to decrypt the ciphertext to plaintext and is designated the private key which is never revealed to another party. This approach also called as asymmetric cryptography, because we use a pair of keys. The figure shows the process of the public cryptographic algorithms. Public key cryptography depends upon the one-way functions, which are easy to compute whereas their inverse function is relatively difficult to compute. RSA, Diffie-Hellman, Digital signature Algorithm (DSA), ElGamal, and Elliptic Curve Cryptography (ECC, are the examples of Public-key cryptography algorithms [3].

Hash Functions:

Hash functions, are also called as message digests and one-way encryption. Hash function algorithms do not use a key to carry out the encryption and decryption process. Instead, the algorithm computes a fixed length hash value based upon the text that keeps both the contents and the length of the message secure.

Tiny Encryption Algorithm is a Feistel cipher encryption algorithm that uses operations from mixed orthogonal algebraic groups like XOR, ADD and SHIFT.

David Wheeler and Roger Needham of the Cambridge University Computer Laboratory designed TEA in the year 1994.

A Feistel cipher is a block cipher with a structure known as a Feistel network. In a Feistel cipher, the data been encrypted is split into two halves. The function F( ) is applied to one half using a sub key and the output of F() is XORed with the other half and the two halves are swapped. Each round function follows the same pattern except for last round. A nice feature of a Feistel cipher is that encryption and decryption are identical i. e. the sub keys used during encryption at each round are taken in reverse order while decryption [4].

The main goal of TEA is to minimize memory footprint and maximize speed. TEA is simple to implement, has less execution time, and takes minimal storage space. TEA uses a large number of iterations rather than a complicated program.

Notation: Any number subscripted with “ h” represents a Hexadecimal number

e. g: 10h represents 16 in decimal values.

Notations for Bitwise Shifts and Rotations:

x << y: denotes logical left shift of x by y bits.

x >> y: denotes logical right shift of x by y bits.

x <<< y: denotes left rotation of x by y bits.

x >>> y: denotes right rotation of x by y bits.

XOR:

In computer science, an XOR is a mathematical operation that combines two bits. It returns value is TRUE if either of the two bits is TRUE, but false if both are equal. For our cryptography algorithm, we do an XOR combining two strings of bits. Say x and y are two string patterns then XOR for x and y is denoted by xA? A aˆ? y [4].

Integer Addition and Subtraction:

The operation of integer addition modulo 2n is denoted by and subtraction modulo 2n is denoted by. Where x, y A? E†E† Z2n (The value of n should be clear from the context)

The key is set at 128 bits and the key schedule algorithm splits the 128-bit key K into four 32-bit blocks K = (K[0], K[1], K[2], K[3]). The 128-bit key is enough to prevent simple search techniques being effective [4].

Encryption Routine:

The Encrypt Routine given in figure [4], is written in the C language and assumes a 32-bit word size. The 128 bit key is split into four parts and is stored in K[0] – k[3] and the Data is stored in v[0] and v[1].

```
void code(long* v, long* k) {

unsigned long y= v[0], z= v[1], sum= 0, /* set up */

delta= 0x9e3779b9, /* a key schedule constant */
```

```
n= 32 ;

while (n–> 0) { /* basic cycle start */

sum += delta ;

y += ((z <<4)+k[0]) ^ (z+sum) ^ ((z>> 5)+k[1]) ;

z += ((y <<4)+k[2]) ^ (y+sum) ^ ((y>> 5)+k[3]) ;

} /* end cycle */

v[0]= y ; v[1]= z ;

}
```

Encryption Routine for TEA

The constant delta is given as delta = (A? E†A? 5 -1) * 231 i. e. 9E3779B9h and is derived from the golden number ratio to ensure that the sub keys are distinct and its precise value has no cryptographic significance.

TEA uses addition and subtraction as the reversible operators instead of XOR. The TEA encryption routine relies on the alternate use of XOR and ADD to provide nonlinearity. The algorithm has 32 cycles (64 rounds). TEA is short enough to write into almost any program on any computer. TEA on one implementation is three times as fast as a good software implementation of DES, which has 16 rounds. The figure shown below [4], gives an overview of two rounds i. e. one cycle of TEA.

Key size: 128 bit key is split into four subkeys K = { K[0], K[1], K[2], K[3] }

Block size: 64 bits

Structure: Feistel Network

Rounds: Variable (64 Feistel rounds (32 cycles) is recommended).

.

Represents Integer addition modulo

Represents XOR

Represents logical left shift by 4 bits

Represents logical right shift by 5 bits

Two Feistel Rounds (one cycle) of TEA

Inputs for the Encryption routine: Plaintext P, Key K

The plaintext is split into two halves as P= (Left[0], Right[0])

Output for the Encryption routine: The cipher text is C

Where C=(Left[64], Right[64]).

The plaintext block is split into two halves, Left[0] and Right[0] and each half is used to encrypt the other half over 64 rounds of processing then combined to produce the cipher text block. Each round i has inputs Left[i-1] and Right[i-1], derived from the previous round, as well as a sub key K[i] derived from the 128 bit overall K.

The Output and the delta constant of the ith cycle of TEA are given as

Left [i+1] = Left[i]

F (Right[i], K [0, 1], delta[i]),

Right [i +1] = Right[i]

F (Right[i +1], K [2, 3], delta[i]),

delta[i] = (i +1)/2 * delta,

The sub keys K[i] are different from K and from each other.

The Round function F contains the key addition, bitwise XOR and both left and right shift operations, and given as

F(M, K[j, k], delta[i]) = ((M << 4) K[j]) A? A aˆ? (M delta[i]) A? A aˆ? ((M >> 5) K[k])

F – Round function and K[i] – key for the i’th round

Encryption Process for TEA

The keys K[0] and K[1] are used in the odd rounds and the keys K[2] and K[3] are used in even rounds. The round function of TEA encryption algorithm differs slightly from a classical Feistel cipher structure where integer addition modulo-2A? A? is used instead of XOR as the combining operator. The above figure[4] gives an overview of the encryption process for TEA.

Decryption Routine:

void decode(long* v, long* k) {

```
unsigned long n = 32, sum, y = v[0], z = v[1],

delta = 0x9e3779b9 ;

sum = delta <<5 ;

/* start cycle */

while (n–> 0) {

z – = (y <<4)+k[2] ^ y+sum ^ (y>> 5)+k[3] ;

y -= (z <<4)+k[0] ^ z+sum ^ (z>> 5)+k[1] ;

sum -= delta ; }

/* end cycle */

v[0] = y ; v[1] = z ; }
```

Decryption Routine for TEA

The decryption routine shown in the figure[4], is same as the encryption routine with the cipher text as input and the sub keys K[i] are used in the reverse order.

Inputs for the Decryption routine: Cipher text C, Key K

The cipher text is split into two halves as C= (DLeft[0], DRight[0])

Where Dleft[0]= ERight[64] and DRight[0]= Eleft[64]

Output for the Decryption routine: The plain text is P, Where C=(DLeft[64], DRight[64]).

F – Round function and K[i] – key for the ith round.

Decryption Process for TEA

The figure [4] gives the structure of the decryption algorithm for TEA. The intermediate value for the decryption process equals the corresponding value of the encryption process with the two halves of the value swapped. For example say the output of the nth round of the encryption process is ELeft[i] concatenated with ERight[i] then the input to the (64-i)th decryption round is DRight[i] concatenated with DLeft[i].

DCT-LSB (Discrete Cosine Transformation-List Significant Bit Encoding):

DCT-LSB is a Steganographic method is a substitution algorithm used for hiding information behind Video files. Each frame in the video holds a part of the secret message. Discrete Cosine Transform (DCT) transforms successive 8 A?- 8 pixel blocks of the frame into 64 DCT coefficients each. The DCT coefficients D(i, j) of an 8 A?- 8 block of image pixels p(x, y) are given by the formula below

Least Significant Bit (LSB) is a simple Steganographic method that takes the individual pixels of the frame and replaces the least significant bits with the secret message bits. It is by far the most popular of the coding techniques used. The process of LSB algorithm is shown in the figure below.

Embed

Extract

LSB Process

We can commandeer the least significant bit of 8-bit true color image to hold each bit of our secret message by simply overwriting the data that was already there. The impact of changing the least significant bit is almost imperceptible.

Input: message, cover image

Output: steganographic object containing message

while data left to embed do

get next DCT coefficient from cover file

if DCT A? aˆ°A 0 and DCT A? aˆ°A 1 then

get next bit from the Secret message

replace DCT LSB with message bit

end if

insert DCT into steganographic object

end while

Embedding Process of DCT-LSB

Input: steganographic object containing message

Output: message, cover image

while data left to extract do

get next DCT coefficient from Stego object

if DCT A? aˆ°A 0 and DCT A? aˆ°A 1 then

Extract the DCT LSB bit from the object

Copy to message file

end ifend while

Extracting Process of DCT-LSB

The above figures[5] gives algorithms for embedding and extracting secret information in video files using DCT-LSB algorithm respectively.

DEFLATE COMPRESSION ALGORITHM

DEFLATE is a no loss compressed data format that compresses data using a combination of the LZ77 algorithm and Huffman coding.

Independent of CPU type, operating system, file system, and character set

Compatible with widely used gzip utility

Worst case 5bytes per 32Kbyte block[6].

Ethical Considerations:

There are two possible ways of attacks on Steganography (Detection and Destruction) of embedded message. The properties of the file in which we are hiding information will differ when

hiding message into it. The Steganalysis will find it and analyse the stego object.

Steganalysis is the technique used to detect hidden messages in digital data like video or audio file steganographically[7]. Steganalysis is used to disrupt the steganographic elements to transfer by extracting, disabling or disrupting.

Detection: Most Steganographic techniques involve in changing the properties of original harmless messages like Image and Video files and the detection algorithms concentrate on detecting these changes [8]. Detecting the existence of a hidden message will save time in the message elimination phase by processing only those digital files that contains hidden information. Detecting an embedded message defeats the primary goal of Steganography techniques that is concealing the very existence of a message [8]. The algorithms vary in their approaches for hiding information. Without knowing which algorithm is used and which Stego-key is used, detecting the hidden information is quite complex.

Destruction or Defeating algorithms concentrate on removing the hidden messages from the Stego object [8].

Steganalysis techniques are similar to the cryptanalysis for the cryptography methods.

As we have discussed previously.

Harmless Message + secret message + stego-key = stega-object

Some of the known attacks for the Steganography are stego-only, known cover, known message, chosen stego, and chosen message.

In Cryptography there are many types of Cryptographic attacks. The attacks are done on the Cipher text. There are some of the ways to attack cipher text like Brute force attacks, Meet in the middle attack, Birthday attack and side channel attack[9].

Plan & Time Table:

ActivityNov ’09Dec’09Dec’09Dec’09Jan09Jan09Jan 09Jan09Feb 09

Selection of topic

XX

analyzing

XX

Research

XX

Literature review

XX

introduction

XX

Rationale

XX

methodologies

XX

Initial results

XX

conclusion

XX

After the approval of the research proposal the project will be started. The dissertation will be preceded according to the steps that are given by the supervisor.

Limitations and Scope:

Steganography is an effective way to obscure data and hide sensitive information. The effectiveness of Steganography is amplified by combining it with cryptography. By using the properties of the DCT-LSB Steganography algorithm for video file and combining it with the TEA cryptography standards, we developed a method, which adds layers of security to the communication. Steganographic methods do not intended to replace cryptography but supplement it.

The strength of our system resides in adding multiple layers of security. First the secret message i. e. word document to be transferred is compressed, encrypted and then embedded in a video file using Steganographic algorithm hence, adding three layers of security. The weakness of the system developed is the size of the secret file i. e. word document after compression should be less than the size of the Cover object i. e. Video file. Since we are using compression algorithm this happens only for huge documents.

As future work, we intend to study more steganalytic techniques i. e. detecting whether a particular file contains any form of embedding or not. We also plan to extend our system so that it can hide digital files in other digital files, for example hiding Audio files in Videos files etc.

Personal Development and Requirement:

Regarding this research a brief knowledge on steganography and cryptography and the methods that are used in embedding and de-embedding file. We should also have a brief idea on encryption and decryption algorithms in cryptography. In this application we can also encrypt strings and document files.

Resource reruirements:

This Application will work on any Microsoft Operating systems and the hard disk should have atleast of 4 MB memory. The RAM should be 256 Mega Bytes or higher.