ISSN 2278-3091
International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 515- 518 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

Secure Communication using Modified S-DES with Steganography

Prashanti.G[1], Sandhyarani.K[2]

[1]Department of Information Technology, VLITS, Vadlamudi, Guntur Dist, AP, India, prashantiguttikonda77@gmail.com
[2]Department of Computer Science and Engineering, VLITS, Vadlamudi, Guntur Dist, AP, India,sandhyakaviti@gmail.com

Abstract: In this paper, a novel steganographic method is proposed that is based on the LSB. Here, not only the message is sent secretly by hiding in the image but also the integrity of secret message can also be checked. First by using SHA-1 hash function the hash value of the message is generated. This hash value is hidden in the 4LSB of the red component of the pixel. Secondly, the secret message is encrypted using modified simplified DES algorithm in which a # function is introduced instead of XOR function. This encrypted message (cipher text) is then hidden in the 4LSB of the green and blue components of the pixel. At the receiver side, 4LSB of the red component of the pixel are extracted and placed in an array which is hash value that checks the integrity of message. The 4LSB of the green and blue components of the pixel are extracted (cipher text) on which modified simplified DES algorithm is applied which gives the original message (plaintext). The SHA-1 is applied on this original message resulting in a hash value. This hash value is compared with the hash value in the array. If both are same that means that data has arrived with integrity with no change on the original message. This paper provides a method for hiding data as well as a method for verifying the integrity of data using spatial domain steganography.

Key words: cipher text, hash value, SHA-1, simplified DES algorithm, spatial domain steganography**.**

INTRODUCTION

As the communication has developed the need for providing security to the information has become an important issue. Three information hiding techniques that are interlinked are steganography, watermarking and cryptography. Steganography is method of hiding information in any digital media like images, audio, video and its main objective is secret communication where as watermarking is hiding information mostly in images/audio files for achieving copyright preservation [1]. Cryptography is another branch where information hiding is text based, with some extensions to image files whose ultimate goal is data protection. The work presented here revolves around steganography in images.

RELATED WORK

Cryptography and steganography methods when combined provide dual security thus enhancing the security of information. Three types of cryptography methods available are:

Symmetric cryptography: where one key is used for encryption and decryption.

Asymmetric cryptography: where two keys namely private key and public key is used for encryption and decryption.

Quantum cryptography: where photons are used to transmit a key for encryption and decryption.

Some steganography methods available are:

Text steganography: where secret information is hidden in the text of the file.

Image steganography: Here, secret information is hidden in the images in such a way that the image is not distorted [1].

Audio & video steganography: In this method, secret information is hidden in audio and video file.

By using symmetric cryptography the message is encrypted and is then hidden in image using image steganography. This provides high level of security because first the hacker has to retrieve the message that is hidden in the image if it is compromised, and then there is another level of security of decrypting the message.

Cryptographic hash functions generate a hash value of fixed size by taking a message as input. MD5 AND SHA-1 are commonly used cryptographic hash functions.

The hash value generated from the hash function called as message digest is used for authentication and checking the integrity of message. Checking or verifying the integrity of the message means detecting if changes or modification are made to the message. This is done by calculating and comparing the hash value of the message before and after transmission of the message. If both the hash values (before and after transmission) are equal this means that the message is received as it is sent by the sender with no changes.

PROPOSED METHOD

This proposed method, is based on SHA-1 algorithm which produces a fixed length hash value along with a modified S-DES algorithm comprising of s-box, secret key, and #operation [5].

A. Encoding Algorithm

- The SHA-1 algorithm is applied to the secret message which is as follows [4].
  - Padding
    - To get a final block of 448 bits the message is padded with a 1 following zeros and appends the size of the original message as an unsigned 64 bit integer.
    - Initialize the 5 hash blocks (h0, h1, h2, h3, h4) to the specific constants defined in the SHA1

ISSN 2278-3091
International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 515- 518 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

standard.

- Hash (for each 512bit Block) for the message schedule set an 80 word array

- The remaining words needed for the algorithm are generated as follows
- Perform the following operation word [i-3] XOR word [i-8] XOR word [i-14] XOR word [i-16] then rotated 1 bit to the left.
  - Loop 80 times doing the following.
- Calculate SHA function () and the constant K (these are based on the current round number).
  - e=d
  - d=c
  - c=b (rotated left 30)
  - b=a
  - a = a (rotated left 5) + SHAfunction() + e + k + Word [i].
  - Add a, b, c, d and e to the hash output.
- Outputs the concatenation (h0, h1, h2, h3, h4) which is the message digest.
- Take a cover image and select the four LSB bits of the Red component of the first pixel and replace with the first four binary equivalent hash values. Select the four LSB bits of the Red component of the second pixel and replace with the next four binary equivalent hash values. Repeat this process till the end of the hash value.
- Now take the secret message and encrypt with the modified S-DES algorithm thus generates the cipher text.

Modified S-DES Algorithm

The keys K1, K2 which are 8-bit round keys are generated as follows [6]

- Permute K as defined in P10

P10									
3	5	2	7	4	10	1	9	8	6

- Perform Left shift by 1 position to both the left and right halves and rearrange the halves with P8 to produce K1

P8							
6	3	7	4	8	5	10	9

- Perform Left shift by 2 positions the left and right halves and rearrange the halves with P8 to produce K2

The steps for encryption are:

- Apply the initial permutation, IP

IP							
2	6	3	1	4	8	5	7

- The input from step 1 is in divided in to two equal parts, left (L) and right (R) bits.
- Expand and permute R using E/P

E/P							
4	1	2	3	2	3	4	1

- XOR input from step 3 with K1
- The left halve of step 4 are applied into S-Box S0 and right halve into S-Box S1:

$$S0 = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 2 \end{bmatrix} \end{array} \qquad S1 = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix} \end{array}$$

- Rearrange outputs from step 5 using P4

P4			
2	4	3	1

- Apply #operation to the output from step 6 with L bits from step 2.
- The left bits are divided into 2 halves and each indicates the table numbers (0, 1, 2, 3) to be mapped [2].
- The output from step 6 indicated the row and column of the table. Now, taking these table number, row and column values mapping is done in the #tables [3]. The #tables are shown below

#0	0	1	2	3
0	3	2	1	0
1	2	3	0	1
2	1	0	3	2
3	0	1	2	3

#3	0	1	2	3
0	1	0	3	2
1	0	1	2	3
2	3	2	1	0
3	2	3	0	1

#2	0	1	2	3
0	2	3	0	1
1	3	2	1	0
2	0	0	2	3
3	1	1	3	2

#1	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

Truth table for # operation

- Now we have the output of step 7 as the left half and the original R as the right half.
- Switch the halves
- E/P with right half
- XOR output of step 9 with K2
- Input to s-boxes
- Rearrange output from step 11 using P4
- Apply # operation to the output of step 12 with left halve from step 8
- Input output from step 13 and right halve from step 8 into inverse IP thus resulting a cipher text.

IP^{-1}							
4	1	3	5	7	2	8	6

- Repeat this process until the entire message is encrypted.

- The cipher text obtained from the previous step is to be hidden in the cover image. For this, select the 4LSB of green and 4LSB blue components of the first pixel and replace it with the 8 bits of the cipher text. Continue this process taking the second, third… pixels until the entire cipher text is hidden [7].

ISSN 2278-3091
International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 515- 518 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

- Obtain the stego image consisting of hash value in red components and encrypted secret message in the green and blue components of the pixel.

B. Decoding Algorithm

1. Take the stego image and extract the 4LSB of the red component of each pixel. Place them in an array A (hash value).
2. Extract 4LSB of the green and 4LSB of the blue component of each pixel and place them in an array A (cipher text).
3. Take 8 bits from the array A (cipher text) and apply it to the modified des algorithm.
4. Repeat step 3 for all the bits in array A (cipher text) finally we obtain the secret message.
5. Take the secret message obtained from the previous step and apply a sha-1 algorithm to it. Here we obtain a hash value.
6. Compare the hash value obtained in the step5 to the hash value in the array A (hash value). If both are same that means secret message has received with entirety with no modification.

EXPERIMENTAL RESULTS

Suppose the secret message is abcd.

Encoding Function:

The hash value for the above message using SHA-1 is
81fe8bfe87576c3ecb22426f8e57847382917acf

The binary equivalent is
10000001111111101000101111111110100001110101011101101100001111101100101100100010010000100111111100011100101011110000100011100111000001010010000101111010110.

Let the cover image be shown in Figure 1.

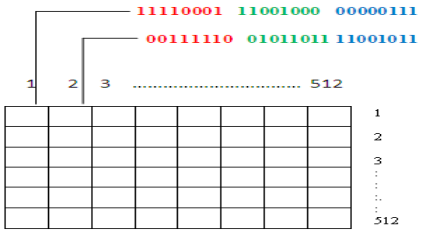

Fig 1: Cover image

After embedding the hash value we get the following image (Figure 2) in which the red components are altered.

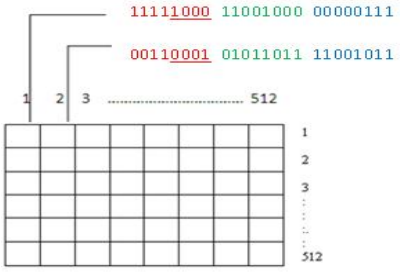

Fig 2: Hash value embedded in the cover image

The binary hash values are replaced with 4 LSB of red component of the pixel as shown below and continue this process for the entire hash value.

The secret message is encrypted as follows:

Convert the secret message into binary code as follows:

01100001 01100010 01100011 01100100 divide these bits into 8-bit blocks and apply each block to the modified S-DES algorithm (Figure 3).

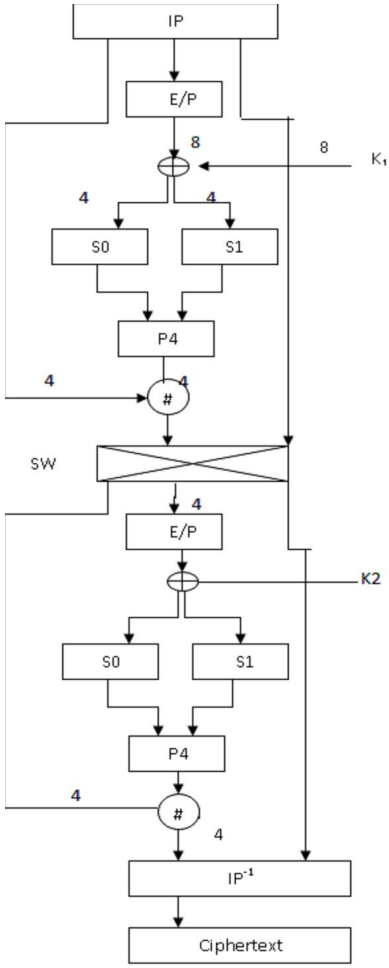

Fig 3: Modified S-DES algorithm with# operation

For example consider an 8 Bit input to the Encoding Function

$(97)_{10} = (01100001)_2$, let K_1= 10100100 & K_2= 01000011.

I / P = 1 0 1 0 0 1 0 0

L = 1 0 1 0

R = 0 1 0 0

E/P = 0 0 1 0 1 0 0 0

X-OR K_1 = 1 0 0 0 1 1 0 0

S-Box Output = 0 0 0 1

P_4= 01 00

P_4# L = 1 1 1 1

SW Input = 1 1 1 1 0 1 0 0

SW Output = 0 1 0 0 1 1 1 1

This output of SW input to the second occurrence of function fk using same S-Box but different key i.e. K_2.

E/P= 1 1 1 1 1 1 1 1

ISSN 2278-3091
International Journal of Advanced Trends in Computer Science and Engineering, Vol.3 , No.5, Pages : 515- 518 (2014)
Special Issue of ICACSSE 2014 - Held on October 10, 2014 in St.Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

X-OR K_2=1 0 1 1 1 1 0 0
S-Box Output = 0 1 0 1
P_4= 1 1 0 0
P_4# L = 1 1 0 0
IP^{-1} = 0 1 0 1 1 1 1 1
Final O/P = 0 1 0 1 1 1 1 1.

Similarly, for the next block 01100010 we get the final O/P as 11000010. Continue this process for the remaining remaining blocks of the message. Now this encrypted message is to be hidden into the image for secure communication. And this is done by hiding encrypted message in the four LSB of green and blue components of the pixel. Finally we get a stego image (Figure 4) consisting of hash value in the red component and encrypted message in the green and blue components of the pixel.

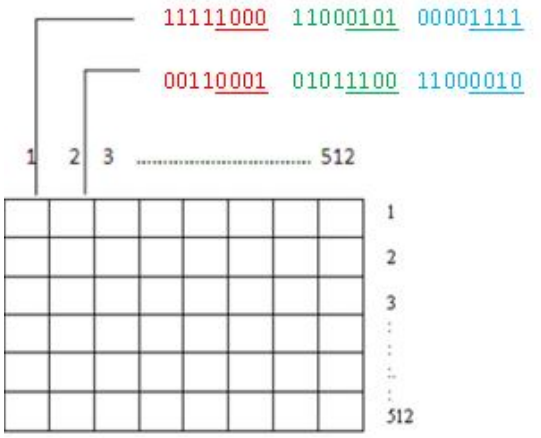

Fig 4:Stego Image

Decoding Function

From the below stego image (Figure 5) the 4 LSB bits of the Red components are placed in an array A (hash value) = 1000 0001……

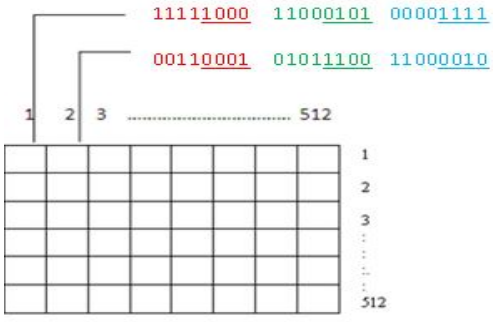

Fig 5: Stego image from which message has to be retrieved

The 4 LSB bits of the green and blue components of the pixel are placed in the array A (cipher text)=0101111111000010…
Now decrypt the bits in the A (cipher text) using modified DES algorithm and hence we obtain the original message (abcd).
This original message is applied to the SHA-1 algorithm and we obtain a hash value which is converted into binary equivalent which is as follows
10000001111111101000101111111110100001110101011101101100001111101100101100100010010000100110111110001110010101111000010001110011100000101001000101111101011001111.

The hash value in A is
10000001111111101000101111111110100001110101011101101100001111101100101100100010010000100110111110001110010101111000010001110011100000101001000101111101011001111.
From the above values we can observe both are same and hence the integrity is verified.

CONCLUSION

In this method we used the LSB steganography for hiding secret data because it produces less distortion and image quality is also maintained as a result the opponent cannot know that secret message is hidden in the image. Along with this we have included another security level by encrypting the secret message with a modified s-des algorithm. So, even if the opponent gets secret message from the image, he has to decrypt the message using modified s-des algorithm which is quite difficult because of the use of # table mapping. Our approach also provides a method for verifying the integrity of the secret message. By this the receiver can check whether the secret message has arrived entirely or not.

.

REFERENCES

[1] Yam Bern Jina Chanu, Themrichon Tuithung and Kh.Manglem Singh, A Short Survey on Image Steganography and Steganalysis Techniques.

[2] Hala Bahjat Abdul Wahab, Abdul Monem S. Rahma, Proposed New Quantum Cryptography System Using Quantum Description techniques for Generated Curves, International Conference on security and management, SAM2009, July 13-16 2009, Las Vegas, USA, SAM 2009.

[3] Afaf M. Ali Al-Neaimi, Rehab F. Hassan, New approach for Modifying Blowfish Algorithm Using 4-States keys, 5th International Conference on Information Technology, 2011.

[4] www.cd.rit.edu/.../48/TeamFlux.pdf.Rochester institute of technology SHA1 Desription.

[5] S. M. Masud Karim, Md. Saifur Rahman, Md. Ismail Hossain, A New Approach for LSB Based Image Steganography using Secret Key, Proceedings of 14th International Conference on Computer and Information Technology (ICCIT 2011) 22-24 December, 2011, Dhaka, Bangladesh.

[6] ict.siit.tu.ac.th/~sgordon/reports/simplified-des-example.pdf.Simplified Des Example

[7] Badrinath R, Anand PS, MSB constrain based variable embedding, IEEE-201S0 ICCCNT'12 26th-28th July 2012, Coimbatore, India.