# Stealth Steganography Using Message Digest Secure Hash Algorithm in Information Security

[1]Kavya Mandavilli , [2]Rambabu Mudusu
[1]Asst. Professor, [2]Assoc.Professor
[1]Department Of CSE, [2]Department Of CSE
[1]KG Reddy College Of Engineering And Technology , Hyderabad, Telangana , [2]KG Reddy College Of Engineering And Technology , Hyderabad, Telangana

***Abstract:*** Information Security is one of the most challenging issue in the internet,network applications E-commerce requirements .So the importance and the value of the exchanged data over the internet or other media types are increasing. The foremost solution to offer the necessary actions in preventing the data against the intruders, hackers and other unauthorized users is cryptography. Common threat to the information is security and this is designed to protect confidentiality, integrity, authentication and availability of computer system data from malicious attackers. A hash function is any function that can be used for padding the data of arbitrary size to the data of a standard size. The values returned by hash function are called as hash values, hash codes, digests, simply hashes. Steganography is the technique used in security. In the steganography data is hidden in the image file and it is transmitted to the receiver in the form of stego-image. This paper focuses on giving the brief usage of LSB technique used in steganography and also the hash algorithm along with the key used in cryptography. Original message is padded by hash function block wise and it produces a digested code. This digested code is considered as a cipher text by using LSB technique and this cipher text is embedded in an image and send it to the recipient (receiver). The receiver receives stego image which contains embedded message. The stego image is extracted using extraction techniques. It splits sender's message into blocks that are of the same size as the input for the hash function. The message is decomposed into block wise based on the fuction but the message block size is small. Digested algorithm includes the framework splits the original message into different blocks of size 512 or 1024 bits. The basic approach of hash algorithm will remain same at both the end users though the hash function changes for each digested algorithm. By this proposal we can provide two stages of security to the original message used in hash algorithm and steganography.

***IndexTerms - Hash Function, Message Digestion, Digital Signature, LSB Techniques, Embedding And Extraction Technique***.

## I. INTRODUCTION

STEGANOGRAPHY is a Greek word. The word Steganos means "covered" and graphia means "writing" [1],[8]. Now a days innovation in steganography has been grown vastly in terms of using the method of wrapping the message within an text or even in an image. Steganography and cryptography differs in terms of view , usage[2]. The important difference between steganography and cryptography is terms of hiding the information or image i.e., suspicion factor. Both cryptography and steganography are implemented at a time then an requirable amount of security will be obtained. Then , a secured procedure is built to essentially protect the isolation of confidential data transmission of the information which is needed. This kind of method is used to make the presence of a secret data appear invisible to unauthorized users such as key loggers,intruders , harmful tracking cookies which can grab the user's keystroke while entering users password and other personal information. Information stolen by malware has been widely used to capture user's password and other confidential data in order to use it for the process of hacking,the other personal possession especially unauthorized access such as falsification of credit cards, skimming ,other identity thefts [4]. Therefore, the proposed method can overcome this problem of embedding the data into Images so that while it is transferring it to the other party it looks like a normal file through the internet or a public domain during information exchange as in [8].

Hence , it is essential to prevent the action taken by other parties while transmitting data for safeguarding human to human communications. Basically, the secret data refers to a message which is saved as a text file that needs to be hidden. In this application, a GIF image will be chosen as a cover medium. The stego – image is the final product after a secret message is embedded in the cover object. A secret message will be concealed in a cover – image by applying an embedding algorithm to produce a stego – image. The transmission of the stego-image via a communication channel is performed b a sender to a receiver. To reveal the convert message that is concealed by the sender, the receiver needs to have the de-stego algorithm which is parameterized by a stego-key to extract the secret message. This is the purpose of a steganographic system where an attacker who does not possess the name of a file or the stego-key for accessing it definitely will not be able to determine whether the file is even present [7]. In an efficient system, a normal cover medium should not be distinguishable from a stego-object [6].steganography mechanism digital images have become common place and now here are these images more prevalent than on the world wide web in the internet [8]. Using digital images as a carrier medium is suitable for information hiding because of their insensitivity for the human visual system [11]. The vast majority of web pages are impressively sophisticated with color images and thus internet users browsing through the web no longer pay attention to sites containing images or to the downloading of images and data files from the web [8]. Besides, there is a large amount of redundant bits in an image. The redundant bits of an object are those bits that can be altered but the alteration cannot be visibly detected by human eyes [3].

## II. Literature survey

Steganography is a technique in which sender input an image called cover image or original image of any file format ( JPEG, BMP, DIP etc) in which he want to hide the secret message and text file containing secret data has to be embedded in an image file. Image containing the secret data is called stego image. Next phase is to select the stego key for encoding. In embedding process data is hidden by using exchange component alteration technique in which pixels have been replaced by key and secret message. Firstly key is converted into binary form and

its binary form is filled in the first component of first pixels. After then, secret message is converted into binary form and its binary form is filled in first component of next pixels. For more security of stego image patching technique is applied by contrasting process. Finally, the verification will be done by parameters PSNR and MSE for stego image of same format and size that gives better result in form of than existing techniques.

**Receiver's prospects:**

Steganography technique in which the sender sends a stego - image to the receiver or legitimate user. This user having the stego key to extract secret data from stego image. The legitimate user must have the same key with which the image is embedded. On stego image extracting process is applied b using patching techniques and exchange alteration techniques. Finally we get the secret data which is embedded.

**Hash function algorithm**

A hashing algorithm creates a hash code, also called a "message digest" or "message fingerprint". Hash codes are of limited use for communications security, because sender can replace both the hash code and the message received by the receiver, but they are an essential element of digital signatures for sharing keys between the sender and receiver.

In this section, we explain how hashing algorithms work, and provide some practical insight into choosing a suitable algorithm for the proposal.

**Creating a hash code**

At the heart of a hashing algorithm is a mathematical function that operates on two fixed-size blocks of data to create a hash code as shown in the below figure.

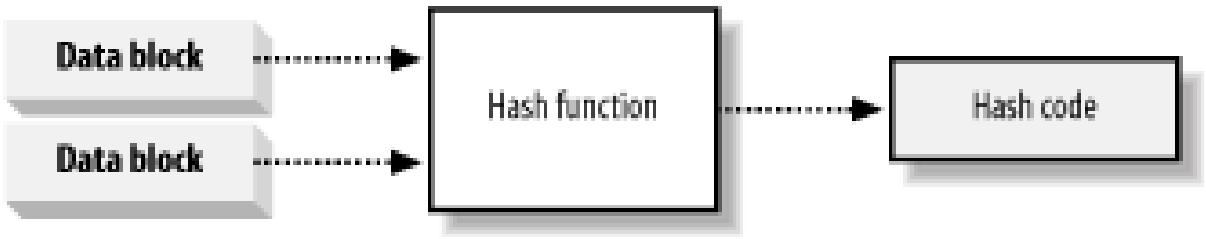


**Figure -1**: The hash function operates on fixed size blocks of data.

It break ups senders message into blocks that are the same size as the input for the hash function. The size of each data block varies depending on the algorithm but the blocks tend to be small the algorithms included in the framework break the messages into blocks of 512 or 1024 bits. The design of the hash functions for each hashing algorithm, but they all share the same basic approach.

The algorithm specifies a "seed" value that feeds into the hash function along with the first block of message data, thus producing our first hash code. Take this hash code and feed it into the hash function along with the second block of message data, creating a second hash Code. Feed the second hash code into the function along with the third message block, and repeat the process of hashing a data block along with the third message block, and repeat the process of hashing a data block along with the hash code of the previous block until processing of all the message data as shown in the figure.

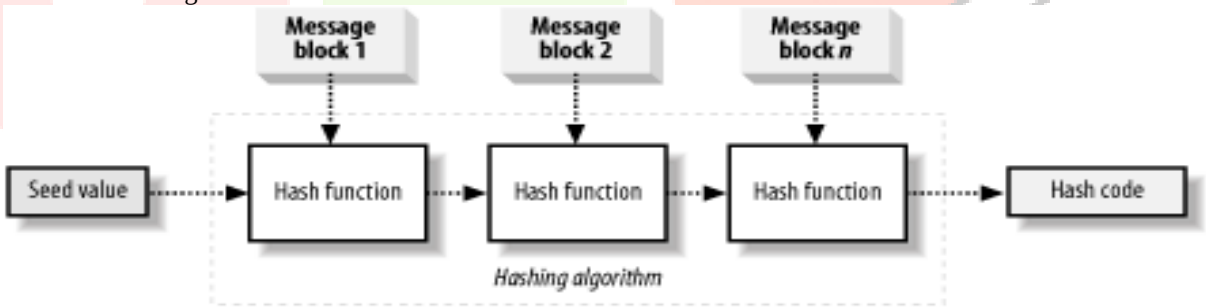


**Figure 2** : Creating the hash code by "chaining" the hash function.

The idea of a hash code acting as a message "fingerprint" is reasonable, because making even the smallest change in the message data changes the value of the final hash code. By "chaining" repeated calls to the hashing function, you can create a hash code that relies on the value of every single bit of the message. The output of hashing the first data block becomes an input to the next operation, which will alter the value of the second hash code, which will affect the result of the third operation, and so on. Known as "ripple effect" or an "avalanche" the huge impact that the smallest change has on the final hash code provides the fingerprint. Two messages that differ by a single bit of data produce very different hash codes.

**III. Proposed system**

Mechanism of proposed steganography technique is as follows

**Sender's prospects**

Phase – 1: Original image and text file: the original image is of any file format having 24 bits per pixel. Due to low computational complexity, it can be applied to very small images of (24 x 24) pixels as well as large images of (512 x 512) pixels. This technique can encode gray scale images as well as colored images directly, with R- G- B levels. After selecting image file, text file is selected which contains secret data.

Phase – 2: Stego-key is a secret variable key that shares sender and receiver. If the key is valid, then only receiver can decode image and retrieve secret data.

Phase -3: Original message is padded with hash code and the produced digested message is hidden in an image.

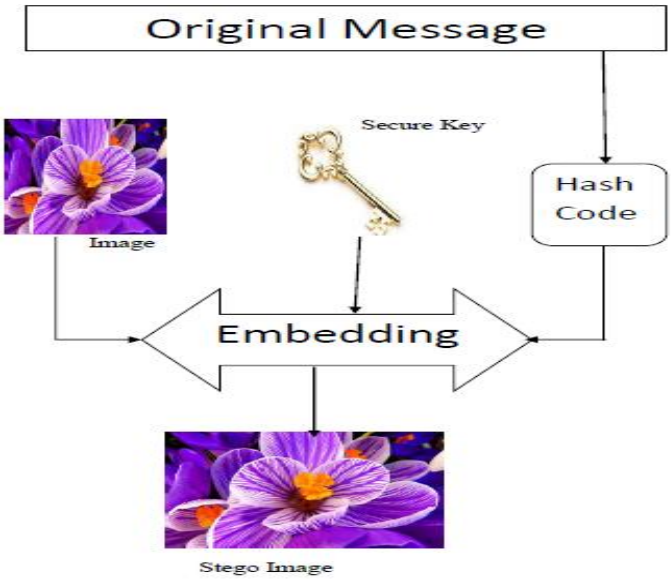


**Figure 3** : sender embedding the message by stego image

The original message is taken as a input and calculates the hash value bit wise.The original message is divided into blocks now the bit in the first block and the bit in the second block are X-OR'ed and in the same way all the bits in each and every block are X-OR'ed and produce one bit of cipher text . In the same way for all the other bits in each and every block values are calculated and each cipher text value is generated for all the bits .Now all the cipher text values of each and every bit are concatenated , at last a cipher message will be generated. Now this cipher text is embedded into an image using LSB techniques. The image is divided into blocks and each block is further divided into pixels. Each pixel is readed and taken in the binary format and last bit is replaced with a value 0 or 1 . Now the neighbor color will be appeared in the pixel which resembles no change in the color of the original image thus the stego image is generated which we can hide our confidential information . so the unauthorized user can't identify the difference between real and stego-image generated.

Phase - 4: Embedment of data and stego key in image is done by using proposed exchange component alteration and patching technique:

Step – ( i ) : The original image is representetd in image buffer as f ( i,j) that contains M x N pixels and a reconstructed image F ( i,j) where F is reconstructed by encoding the f (i,j). Extract all the pixels in the given image f(i,j) and store it in the array called Pixel-Array P(i,j).

Step – (ii) : Extract all the characters in the given text file ans store it in the array called Character- Array c(i,j).

Step – (iii) : Extract all the characters from the stego key ans store it in the array called Key-Array k( i,j).

Step – (iv) : Exchange first pixel with last pioxel of each row of Pixel – Array p(i,j) (i.e., LSB ↔ MSB).

Step – (v) : Replace first pixel of each row of Pixel – Array p(i,j) with first pixel of each row of Key-Array k( i,j). If there are more no. Of characters available in Key – Array,then place rest of remaining character in the next component of pixels array, otherwise follow step (vi).

Step – (vi) : The terminating symbol 0 indicates the end of the key in Key-Array k( i,j).

Step – (vii) : Place characters of Character –Array c(i,j) in each first component (blue channel) of next pixels by replacing it.

Step (viii) : Repeat step (vi) till all the characters of Character –Array c(i,j) has been embedded.

Step – (ix) : Again place some terminating symbol 0 in end of the Character –Array c(i,j) to indicate end of the data.

Step – (x) : A pseudo random generator is used to select two fields of the images ( or patches ), patch A and patch B.

Step – (xi) : All the pixels in patch A is lightened while the pixels in patch B is darkened. In other words, the intensities of the pixels in the patch A are increased by a constant value, while the pixels of the other patch B decreased with the same constant value. For each pair, let d be the difference between intensities of two pixels. Encode a bit of information in to the pair.

Step – (xii) : Let $d < 0$ represents 0 and $d > 0$ represents 1.if the pixels are in the wrong order, switch them. If d happens to be greater than a predefined threshold or if equal to 0, discard the pair and move on to next pair.

Step – (xiii) : Resultant image will hide all the characters that are given as input. Data is embedded using above algorithm and stego image is produced and sent to the receiver or legitimate user.

Step- (xiv) : Error metrics are computed on the luminance signal only so the pixel Values f (I,j) range between black (0) and white (255). Firstly, the mean squared error (MSE) of the reconstructed image is computed as follows:

**Receiver's prospects:**

Phase -1 : Stego image : the sender sends a stego image to the receiver or legitimate user. The legitimate user is having the stego key to decode secret data from stego image.

Phase – 2: Stego key after receiving image b receiver or legitimate user, the legitimate user must have the same shared key with the image is encoded.

Phase -3 : Extracting of stego image using proposed exchange component alteration and patching technique:

Decoding algorithm of exchange component alteration and patching technique includes following steps:

Step (i) : All the pixels in patch A is darkened while the patch B is lightened to original. In other words the intensities of the pixels in the one patch are decreased by a constant value, while the pixels of the other patch are increased with the same constant value.

Step (ii) : Consider three arrays. Character – Array c( i, j) , Key-Array k(i,j) and Pixel-Array p(i,j).

Step(iii) : Extract all the pixels in the given image f(i,j) and store it in the array called Pixel-Array p(i,j).

Step (iv) : Now, start scanning the first pixel and extract key characters from first (blue) component of the pixels and place it in Key – Array k(i,j). Follow phase-3 till we get terminating symbol, otherwise follow step (v).

Step (v) : If this extracted key matches with the key entered by the receiver, then follow the same procedure otherwise terminate the program by displaying message – key is not matching.

Step (vi): If the key is valid, then again start scanning next pixels and extract secret message characters from first (blue) component of next pixels and place it in Character Array c(i,j). Follow Step (vi) till we get terminating symbol, otherwise repeat the step.

Step (vii) : Exchange last pixel with first pixel of each row of Pixel-Array p (i, j) (i.e MSB ↔ LSB).

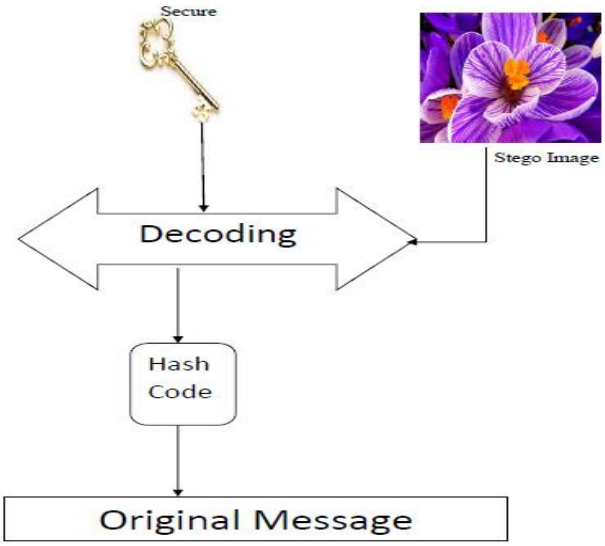


**Figure 4**: Receivers extraction of the stego image into an original message.

The stego image is received by receiver , now the receiver will extract the image using extraction methods. The extracted image is relieve the text in the form of hash code. Now the hash code is taken as input and calculates the hash value bit by bit block wise. The cipher message is divided into blocks now the bit in the first block and the bit in the second block are X-OR'ed and in the same way all the bits in each and every block are X-OR'ed and produce one bit of original text . In the same way for all the other bits in each and every block values are calculated and each original text value is generated for all the bits .Now all the original text values of each and every bit are concatenated , at last a original message will be generated which is exactly send by the sender.

## IV. Conclusion

The proposed steganography technique for gray and colored images will provide effective security for images , but this work will be further improved to other types of image formats (TIFF, JPEG 2000 etc) and these images has the aspect of the slightly contrary i.e., having greater resolution to other applications of proposed method. Therefore, future work includes the search for the plan which is not contrary to the other. Video files can also be used to transmit data, however time consumption increase in this case.

## V. References

1. C. Cachin, "An Information-Theoretic Model for Steganography", In 2nd Information Hiding Workshop, vol. 1525, pp. 306-318, 1998.
2. R Chandramouli, N. Memon, "Analysis of LSB Based Image Steganography Techniques", In IEEE pp. 1019-1022, 2001
3. W. Stallings. Cryptography and Network Security Ã¢Â□Â“ principles and practices. Pearson Education, Inc., 2003.
4. L. A. Bygrave, "The technologisation of copyright: implications for privacy and related interests," European Intellectual Property Review, vol. 24, no. 2, pp. 51Ã¢Â□Â“57, 2002.
5. H.W. Tseng and C.C. Chang, "Steganography Using JPEG-Compressed Images," In Fourth International Conference on Computer and Information Technology, pp. 12-17, 2004.
6. C.Y. Yang, "Color Image Steganography based on Module Substitutions," In Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing, vol. 2, pp.118-121, 2007.
7. S.G.K.D.N. Samaratunge, "New Steganography Technique for Palette Based Images," In Second International Conference on Industrial and Information Systems, pp. 335-340, Aug. 8 Ã¢Â□Â“ 11, 2007.
8. S .K. Moon and R.S. Kawitkar, "Data Security using DataHiding," International Conference on ComputationalIntelligence and Multimedia Applications, vol. 4, pp. 247-251, 2007.
9. J. He, S. Tang and T. Wu, "An Adaptive ImageSteganography Based on Depth-Varying Embedding," InCongress on Image and Signal Processing, vol. 5, pp. 660-663, 27-30 May 2008.
10. J.G. Yu, E.J. Yoon, S.H. Shin and K.Y. Yoo, "A NewImage Steganography Based on 2k Correction and Edge-Detection," In Fifth International Conference onInformation Technology: New Generations, pp. 563-568,2008.
11. W. N. Lie and L. C. Chang." Data hiding in images withadaptive number of least significant bits based on thehuman visual system." Proc. ICIP '99, 1:286Ã¢Â□Â“290, 1999.
12. H. T. S. M. Kharrazi and N. Memon. "ImageSteganography: Concepts and Practice". WSPC, 2004.

13. AlkhraisatHabes, "Information transmissions in computernetwork. Information hiding in bmp image Implementationanalysis and evaluation" (Jan.2006)
14. S. K. Moon, V. N. Vasnik, "Application of steganographyon image file", National conference on Recent trends inElectronics, pp. 179-185
15. Chns J Mitchell " The Institution of Electrical EngineersPrinted and published by the IEE". Savoy Place, LondonWC2R OBL, UK, 1995.