A Novel Approach to Enhancing Secure Sharing Using Modified LSB Technique

R. Rajavarman[a], K. Mahalakshmi[b], V.S.Keerthi[c] , K.Karishma[d], and M.Aarthi[e]

[a] Assistant Professor, K.Ramakrishnan College of Technology, Trichy.
[b,c,d,e] UG Students, K.Ramakrishnan College of Technology, Trichy.

Article History: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 16 April 2021

Abstract: Multiple secret sharing is a method of dividing an image into random shares. With the help of improved LSB technology, the text message is hidden in the cover image and then split into multiple shares. XOR-based multi-secret sharing technique is utilized to securely send pictures from the source to the objective. Multiple secret images can be transmitted at the same time. The access key is shared via email to verify the integrity of the shared information. The secret image can only be displayed after the recipient receives all n shares and decrypts them**.**

Keywords: exclusive-or, secret sharing, multi secret sending, Stego-Image, visual secret sharing, visual cryptography, reversible data hiding(RDH-EI).

1. Introduction

Reversible information covering up is a kind of interactive media security that can give copyright recognizable proof and honesty affirmation for sight and sound substance on outsider stages, (for example, outsourced cloud storage). Encryption is the process of encrypting and decrypting media data before cloud outsourcing. Steganography is a technique that hides text, images, and audio in another text, image, or video. The secret sharing scheme that can be decrypted by people is called Visual Secret Sharing (VSS). The secret sharing scheme is a mechanism for sharing secret images among a group of participants, and because the hidden content is fully protected, it is very useful in many fields. Use multiple sharing technology based on visual encryption to share images. Share multiple secrets in one broadcast and avoid sharing a single secret image. It increases the embedding capability and also retains the restored image quality. [1] Lightweight cryptography. [2] RDH and cloud data management in privacy-preserving and data security. [3] Bit plane partition and MSB prediction. [4] Adaptive Bit-level data embedding and Checkboard prediction. [5] Decrease redundancy by pixel value reversible data hiding used for embedding images LSB-put up extra data. [6] Images encrypted by improved reversible data hiding encrypted images generated by XOR hiding key extract hidden data. [7] Improved Reversible data hiding in encrypted images based on reversing room after encryption and pixel prediction. [8] MSB technique for replacing in data hiding phase and gradient adjust prediction algorithm (GAP). [9] Embeds message by public key modulation mechanism for data extraction. [10] Encrypted image generation, data embedding, data extraction, image recovery.

2. Previous model

RHD-EI permits the worker to install different messages in the scrambled picture transferred by the substance proprietor and guarantees that the first substance can be reestablished without misfortune after being encoded on the beneficiary's page. The proposed model based on a secret partition with multiple data encoders includes three stages: image encryption stage, data encryption stage, and data extraction and image restoration stages. Not at all like the past model which contains just a single information stockpiling bureau, the proposed model contains numerous information stockpiling cupboards. In the proposed model, the first picture is changed over into numerous scrambled pictures with a similar size as the first picture, and the encoded pictures are dispersed to various information encoders for information encryption. Every information blender can autonomously insert information into the encoded picture. On the beneficiary's page, the first picture is remade from many labeled scrambled pictures and installed information. In composite mode, information extraction can't be performed on the scrambled space. Before removing information, the scrambled picture should be encoded with an encryption key, which demonstrates that the extraction of information is identified with the proprietor of the substance. For this situation, the beneficiary should get consent from the substance proprietor while checking the trustworthiness of the stamped encoded picture. In the separable mode, the encoded information can be removed straightforwardly from the scrambled picture without the KD encryption key, which implies that the information extraction has nothing to do with the substance proprietor. For this situation, the information checker can utilize the way to conceal the information on a case-by-case basis to refresh the inserted information.

2.1 Disadvantages

The proposed method is to change the operation of encrypted images with different marks, so it takes longer to decrypt. Sharing multiple shares will increase additional computational consumption. For content owners and recipients, the speed of expansion is increasing linearly.

3. Proposed System

The principal objective of the task is to set up a secure correspondence between sender and collector utilizing email and other specialized techniques. In this work, XOR-based multi-secret sharing is proposed to securely send pictures from the source to the objective.

This method eliminates basic VC security challenges, such as external codebook use, random sharing mode, pixel expansion in shared and restored images, lossy restoration of secret images, and limiting the number of images. The proposed method is an n-to-n multi-secret sharing scheme. Through this proposed work, multiple secret images can be transmitted at the same time. The SMS is created by the sender and hidden in the selected cover image file. The secret image can only be displayed after the recipient receives all n shares and decrypts them. Type the text and hide it in the image. This is done using the modified LSB method. Then use the XOR-based VC method to encrypt the image and send it to the receiver. The key used to encode the offer will be shipped off the beneficiary. The beneficiary will decode the offer utilizing a similar key utilized for encryption. Then, the improved LSB strategy will be utilized to extricate the concealed content from the reestablished picture

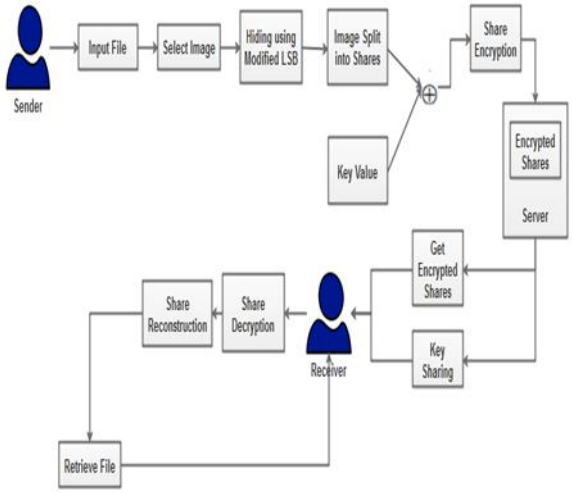


Fig 1.System Architecture Diagram.

3.1 Module List

- Enrolment and Data Sharing
- Image Upload and Hiding
- Share Split and Encryption
- Multi Secret Sending
- Share Decryption and File Extraction

3.2 Module Description

3.2.1 Enrolment and File Sharing

Enrollment is the process of registering in an application to obtain access authorization. Then, the sender can create a text message to share with the recipient. Hiding the secret text message is a process of embedding the secret text in the covering medium imperceptibly by modifying the elements of the covering medium to a minimum. The sender will generate the content to be sent to the receiver in this form.

3.2.2 Image Upload and Hiding

This process consists of selecting the cover media to hide the information. Here the images are used as support for the cover of the secret message. The cover image is also selected by the sender when creating the secret message. The original message is hidden in the cover media (image) to enhance the security of data sharing. The steganography image to be sent must be uploaded. The image should be any of the formats that support the image. A text is written and hidden inside a mystery image. This is done using the modified LSB method. The cover image is called a steganography image.

3.2.3 Share Split and Encryption

The uploaded image will be divided into several "N" shares according to the user's needs. "N" is the product of rows and columns. Here in this diagram, the number of shares is 16 (4 * 4). The maximum number of actions is fixed at 8 * 8. Separate image shares are separately encrypted using the XOR method. A key is used to encrypt shares. Encryption requires access to both encryption or encryption keys and encryption keys, but although the

encryption algorithm is very simple, it is almost indestructible. This key will be sent to the recipient. If a JPEG image is used, the encrypted partition will be in black and white. It looks like a QR code.

3.2.4 Multi Share Sending

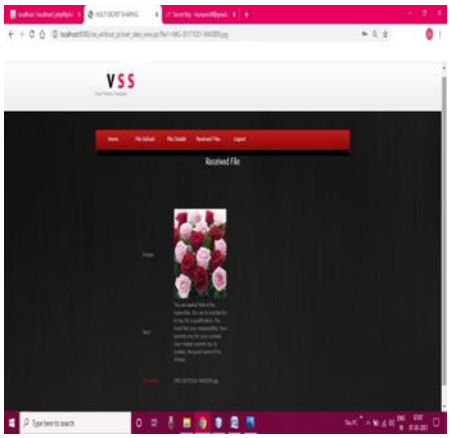
All individual encrypted shares will be stored in a folder. Using this form, all encrypted shares will be sent to the recipient in one transmission. This allows a single broadcast recipient to receive all shares at the same time. This helps to avoid missing information or shares and also saves transmission and reception time for both sender and recipient.

3.2.5 Share Decryption and Data Extraction

The receiver will receive all encrypted shares in one transmission. Each received share will be individually decrypted using the reverse XOR method. The key received by mail is used for this decryption process. The private key is used in the encryption and decryption process. The yield of this table will be a solitary offer in the decoded table. All individual unscrambling shares are the contribution of this module. These individual offers will be joined to shape the first (secret) picture. The reestablished picture can be viewed as a total single picture. The size of the first picture and the reestablished picture will be something very similar. The hidden text file will be recovered from the confidential image. The recipient will receive a secret message with cover text. The LSB method is used to retrieve the hidden text, and a specific key is generated during the email sending process and shared with the recipient. The recipient can decrypt the text using the shared key. The original message is then shown to the receiver.

3.3 Advantages

The mysterious picture and the reestablished picture will have a similar size. Multi-secret sharing is utilized to send various offers simultaneously. Utilize the XOR calculation to upgrade security.



4. XOR Using Modified LSB Algorithm

In the process of embedding the secret message, the cover image is divided into non-overlapping blocks of nine consecutive pixels. Based on these values of the nine pixels in each block, a difference is calculated. All possible differences are divided into a series of ranges. Then, the calculated difference value is replaced with a new value to merge the value of the auxiliary stream of the secret message. The number of bits that can be embedded in a pair of pixels is determined by the width of the interval and the difference. The method of embedding confidential information in confidential documents is called LSB insertion. In the proposed technique, the binary representation of secret data has been adopted, and the LSB of each byte is overlaid in the image. If you perform LSB with a 24-bit color image, the amount of editing will be reduced.

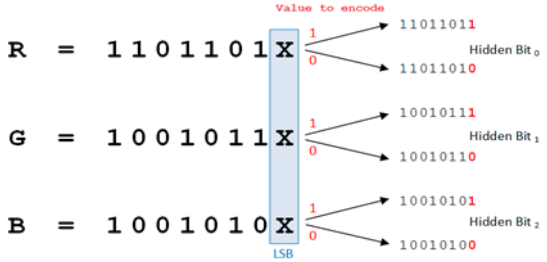


Fig 2. LSB Steganography.

4.1 LSB Encoding

Acquire the overall pieces of the encoded privileged intel and the bytes addressing the pixels of the mysterious picture. The counter beginnings from being set to 1, and successively gives the pixel byte file range, in which the mysterious message cycle in the LSB is accessible. Proceed with this interaction until the last piece of the mysterious message is reached. From that point forward, a pieced pattern of the message will be produced. The accessible pieces are assembled to shape the bytes so every byte addresses an ASCII character. The characters are put away in the content substance record addressing the scrambled inserted message. From that point onward, decoding and decompression should be performed. Get the message bits one by one, and then put a few bytes of the image into the LSB. Follow the same process until all the bits of the message are in the bytes of the photo. The resulting image is called "Stego-Image". It is designed for transmission over the Internet.

Algorithms to hide mysterious facts in the cover image:

Step 1: Read the cover media image and the secret information that needs to be embedded in the cover image.

Step 2: Organize secret facts.

Step 3: Use the key shared by the receiver and sender to convert the compressed secret into encrypted text content.

Step 4: Convert the compressed encrypted text content message into a binary form.

Step 5: Find the LSB value of each RGB pixel in the cover image.

Step 6: Embed the RGB pixel LSB secret data bit of the cover image.

Step 7: Continue the process until the confidential information is completely hidden in the cover document.

4.2 LSB Decoding

Get the overall pieces of the scrambled restricted data and the bytes addressing the pixels of the mysterious picture “stego-image”. The counter beginnings from being set to 1, and successively gives the pixel byte list range, in which the mysterious message digit in the LSB is accessible. Proceed with this cycle until the last piece of the mysterious message is reached. From that point forward, a pieced pattern of the message will be created. The accessible pieces are assembled to shape the bytes so every byte addresses an ASCII character. The characters are put away in the content substance record addressing the scrambled implanted message. From that point forward, decoding and decompression should be performed.

Algorithm to discover secret data from Stego image:

Step 1: Read the Stego picture.

Step 2: Find the LSB value of each RGB pixel of the incognito image.

Step 3: Find and retrieve the LSB of each RGB pixel of the incognito image.

Step 4: Continue the process until the message is completely extracted from the hidden image.

Step 5: Decompress the extracted secret facts.

Step 6: Use the shared key to decrypt the confidential record to obtain the original record.

Step 7: Rebuild confidential statistical information.

4.3 Encryption Algorithm

Even though XOR encryption is anything but a public key framework like RSA, it is practically indestructible through beast power strategies. It is helpless to modes, yet this burden can be maintained a strategic distance from by packing the record first (in delete mode). Exclusive encryption necessitates that both the code essayist and decryptor approach the encryption key, yet the encryption calculation is amazingly basic, yet it is practically indestructible. XOR encryption works by utilizing the Boolean (XOR) work. XOR is a parallel administrator (which implies it requires two boundaries, for instance, like the in addition to signing). Utilizing its name "exclusive OR", it is not difficult to deduce (right, no not exactly) assuming one of the two administrators (just one) is valid, it will bring valid back.

The thought behind XOR cryptography is that it is difficult to invert the activity without knowing the underlying estimation of one of the two autonomous factors. For instance, if you XOR two factors with obscure qualities, you can't tell from the yield what the qualities of these factors are. For instance, on the off chance that you play out the activity A XOR B and return TRUE, you won't know whether An is FALSE and B is TRUE or B is FALSE and An is TRUE. Likewise, regardless of whether it returns FALSE, it can't be resolved whether both are TRUE or FALSE.

Nonetheless, on the off chance that you know An or B, it is reversible, which is unique to legitimate and sensible OR. For XOR, if you play out the activity A XOR TRUE and the return esteem is TRUE, you realize that An is FALSE, and on the off chance that it returns FALSE, you realize that An is True. The guideline of XOR encryption is that if you have a scrambled string and encryption key, you can generally decode effectively. If you don't have a key, you should make an arbitrary key and attempt each key until the yield of the unscrambling program takes after intelligible content before you can decode it. The more it takes to make the encryption key, the more troublesome it is to break the key.

The genuine technique for utilizing "exclusive OR" encryption is to acquire the key and scramble the document by more than once applying the way to progressive portions of the record and filing the yield. Since the key is arbitrarily produced, the yield will be comparable to an irregular program. When a subsequent individual approaches the key, that individual can unscramble the record, however, without it, decoding is practically incomprehensible. For each piece added to the key length, you will twofold the number of endeavors to beast power the encryption.

XOR encryption is difficult to break through the so-called "brute force" method (brute force = use a random encryption key, hoping to find the correct key), but this encryption method is susceptible to pattern recognition. The pattern can be easily avoided by compressing the file before encrypting it (compression has made it unreadable, deleting the pattern).

The XOR encryption strategy doesn't utilize public keys, like RSA. Interestingly, both the individual who scrambles the record and the individual who needs to decode the document should have the encryption key. XOR encryption (as the name proposes) utilizes the Boolean mathematical capacity XOR. The XOR work is a parallel administrator, which implies that two boundaries are required when utilizing it. On the off chance that one of the two boundaries is valid and the other is bogus, the XOR capacity will bring valid back.

5. Conclusion

The proposed strategy portrays how to safely move secret pictures from source to objective. The sender should choose the picture to be sent covertly to the beneficiary. The mysterious picture is isolated into "n" parts. Each offer is scrambled utilizing the XOR activity. At that point, all encoded offers will be communicated to the beneficiary in a solitary transmission. The beneficiary should utilize the unscrambling key to decode the offer. After unscrambling, the individual offers will be converged to shape a reestablished (unique) picture. The reestablished picture will be a similar size to the original picture.

The confirmation and thinking behind this strategy ensure that the enemy cannot change the previous image without spoiling the previous image, which makes their security analysis more difficult and practical. Future work will explore verification in more detail.

6. Future Enhancement

In future work, different algorithms need to be used to improve authentication, and for shared images, the size of the restored image should be considered the same. The noise level should also be reduced, and multiple shares of encryption and decryption time can be calculated to improve performance.

References

1. Chen, Yu-Chi, Tsung-Hsuan Hung, Sung-Hsien Hsieh, and Chih-Wei Shiu. "A new reversible data hiding in encrypted image based on multi-secret sharing and lightweight cryptographic algorithms." IEEE Transactions on Information Forensics and Security 14, no. 12 (2019): 3332-3343.
2. Liao, Xin, and Changwen Shu. "Reversible data hiding in encrypted images based on an absolute mean difference of multiple neighboring pixels." Journal of Visual Communication and Image Representation 28 (2015): 21-27.
3. Wu, Hao-Tian, Zhiyuan Yang, Yiu-Ming Cheung, Lingling Xu, and Shaohua Tang. "High-capacity reversible data hiding in encrypted images by bit plane partition and MSB prediction." IEEE Access 7 (2019): 62361-62371.
4. Yi, Shuang, and Yicong Zhou. "Parametric reversible data hiding in encrypted images using adaptive bit-level data embedding and checkerboard based prediction." Signal Processing 150 (2018): 171-182.
5. Bartwal, Monika, and Rajendra Bharti. "Lossless and Reversible Data Hiding in Encrypted Images With Public Key Cryptography." Annals of Computer Science and Information Systems 10 (2017): 127-134.
6. Dragoi, Ioan Catalin, Henri-George Coanda, and Dinu Coltuc. "Improved reversible data hiding in encrypted images based on reserving room after encryption and pixel prediction." In 2017 25th European Signal Processing Conference (EUSIPCO), pp. 2186-2190. IEEE, 2017.

7. Qian, Zhenxing, Hang Zhou, Xinpeng Zhang, and Weiming Zhang. "Separable reversible data hiding in encrypted JPEG bitstreams." IEEE Transactions on Dependable and Secure Computing 15, no. 6 (2016): 1055-1067.
8. Liu, Jianyi, Kaifeng Zhao, and Ru Zhang. "A fully reversible data hiding scheme in encrypted images based on homomorphic encryption and pixel prediction." Circuits, Systems, and Signal Processing (2019): 1-21.
9. Zhou, Jiantao, Weiwei Sun, Li Dong, Xianming Liu, Oscar C. Au, and Yuan Yan Tang. "Secure reversible image data hiding over encrypted domain via key modulation." IEEE transactions on circuits and systems for video technology 26, no. 3 (2015): 441-452.
10. Wu, Han-Zhou, Yun-Qing Shi, Hong-Xia Wang, and Lin-Na Zhou. "Separable reversible data hiding for encrypted palette images with color partitioning and flipping verification." IEEE transactions on circuits and systems for video technology 27, no. 8 (2016): 1620-1631.