LockShield: Image Steganography with cryptography for client-side cloud storage

Amita V. Shah[1], Pooja Shah[2]

[1]Ph.D. Scholar, Computer/IT Engineering, Gujarat Technological University, Gujarat, India
[2]Assistant Professor, Shankersinh Vaghela Bapu Institute of Technology, Gujarat, India

Abstract*: Steganography translates to "covered in concealed writing". By using computer-based steganography, it is possible to alter so-called digital carriers, such as images and sounds. If successful, the alterations represent the concealed message but have no obvious effect on the carrier. Information can be concealed in carriers including photos, audio files, text files, videos, and data transmissions via steganography. When the message is concealed within the carrier, a stego carrier, such as a stego-image, is created. Hopefully, the human senses will perceive it as closely as possible to the original carrier or cover image. The most common carrying media is images. The following describes how they are used for steganography. First, the message might be encrypted. They are utilized for steganography in the manner listed below. Initially, the message might be decrypted. The secret message to be told is concealed within a graphic file by the sender. As a result, a phenomenon known as stego-image is created. A stegokey or any additional secret information may be required throughout the hiding procedure. The addressee then receives this stego-image. Another major problem is storing the data on Cloud Storage services, which are used by many people. Today almost everybody and most software make use of cloud services directly or indirectly. There are various cloud service providers such as Google Drive, One Drive, and Dropbox. Various parameters surround the cloud such as bandwidth, location, and pricing. But security is one of the most important aspects that's why most cloud services make use of service-side encryption, for example, Google Drive uses the AES 256 encryption algorithm. But if a third party has illegitimate access to the cloud, then the data is vulnerable. The following abstract proposes a review of research on cloud security issues and their solutions specifically through the implementation of client-side data encryption using techniques such as image steganography and cryptography. In the literature on picture encryption, numerous cryptosystems have been introduced to increase communication security. This research suggests a revolutionary picture encryption approach based on a faster algorithm than existing ones. The proposed method ensures that the secret key is never disclosed during the encryption process. Steganography, symmetric encryption, and asymmetric encryption theories serve as its foundations. The secret key is then hidden in the ciphered image using an at least significant bits steganographic strategy after being initially encrypted using an asymmetrical technique to cypher the image.

Keywords: Client-Side Encryption, Data Security, Steganography, Cryptography, RSA

INTRODUCTION

An essential part of data security is encryption. It makes sure that if the data is protected and also an unauthorized individual cannot read it and cannot use it inappropriately. File transfer encryption refers to the process of encrypting data as it is transferred between devices. This type of encryption operates by encrypting the data in a way that is oblivious to humans and then decrypting it once it has reached its terminus. The two encryption methods that are typically utilized in the encryption-decryption process by the papers are image steganography and cryptography. The art of image steganography involves obscuring data—text, images, or even videos—within a cover image.

The hidden information is concealed such that it cannot be seen by human sight. Cryptography is the use of codes to safeguard data and communications so that only the intended receivers can decipher and process them. As a result, unauthorized persons are blocked from accessing information.

Furthermore, steganography obscures the existence of the data, whereas cryptography conceals the meaning

of the data. Even though they are different approaches, they might very well be combined to achieve the best of both worlds in the same situation. The project is founded on the same idea, assisting users in securely transferring data across locations with the use of image steganography and cryptography without any problems and with great ease.

The analysis of the papers is done to get proposed enlightenment about the analysis done to perceive the idea to build a full-fledged project for client-side encryption and data storage strategies. This review is also done to expand our pre-existing understanding of data encryption techniques and cloud storage capabilities and also for gaining more perspicuity on how things work in the cloud so that we could patch the flaws in the existing system, in our project.

LITERATURE SURVEY

The suggested method enables the identification and expulsion of malicious cloud users. As a result, privacy can be maintained at a lower computational cost. There are four main aspects, including user tracking, file permission & policy file creation, data owners' EHR upload, and profile creation & key generation [1]. Tiger Hash-based. Kerberos Biometric Blowfish Authentication (TH-KBBA) Mechanism is used to access server data. It consists of three steps: registration, authentication, and granting of tickets [1]. When a client. requests data access from CS, the server uses blowfish decryption to decrypt the ticket and check the user ID [2]. The current fad is to keep databases and applications in a cloud setting. The tracing of unauthorized users (hackers) who attempt to access the resources is made possible by big data security in the cloud. It is demonstrated that work is extremely efficient and prevents unauthorized users from accessing the data [3]. Cloud computing and encryption are frequently spoken together. But transferring data to the cloud is a significant change that requires real effort. This research study introduces and enforces the notion of symmetric key encryption. It provides larger files with improved security and performance, researchers say [4].

Steganography, which essentially involves disguising the cipher text in images or text, is a current method for protecting user data. Images and sounds play the role of a cover item, while a password or numerical value serves as the Stego key. Data like genetic codes, DNA sequences, nuclear codes, etc. are protected via steganography [5]. The system uses a fusion of visual cryptography and image steganography. The major goals of this system are to simultaneously strengthen secret image security and retain stego image quality. Encoding and secret decoding images both involve the use of Sharel images. 24-bit color pictures are employed in this system [6]. An image can be represented using a matrix. This three-dimensional matrix of individual RGB (red, green, and blue) values can be used to represent an RGB image. We can encrypt a picture by making some changes to the entire image matrix, and we can decode an image by doing the opposite [7]. Cryptography is seen as being exceedingly brittle yet tremendously helpful, as cryptographic systems can be broken by a single programming or specification error. Digital signatures, in contrast, did not exist before the development of computers. The "digitalization" era marked the beginning of the interaction that established the connection between signature and encryption. [8].

The key access is only available to the sender and the receiver. However, an individual can track down the transmission and can decode the encrypted message. To overwhelm the weakness, image steganography comes in handy. It helps in increasing the robustness. The computation complexity improves drastically. The many steganographic approaches, including image, audio, and video steganography, must concentrate on their ability to conceal information, their ability to be detected, their level of visibility, and their resistance to malicious and inadvertent attacks [9]. In this research, they have presented two new methods that combine steganography and cryptography to both encrypt data and hide its encrypted contents in other media. In the first method, they protected an image by encrypting it with the S-DES algorithm and a secret key, then they hid this text in another image. For security, they used the S-DES algorithm and an image key to directly encrypt an image. The information obtained in this way is then concealed inside another image. Both of these methods have been examined, and it has been found that they also successfully thwart the possibility of steganalysis [10]. There has been a significant increase in text and multimedia data transfer through the internet as a result of significant developments in internet technology. Because of this, data security is a crucial requirement. Data security is achieved through the use of cryptography and steganography. Steganography is the art of concealing sensitive data in another cover medium, such as an image, audio file, or video. The art of turning legible data into an unreadable format is known as cryptography. To increase the security of data, steganography, and cryptography can be combined. In this research, a novel approach is put forth that

combines steganography and cryptography to secure 24-bit color images. This technique employs a randomized LSB-based mechanism to conceal one image within another [11]. It is advantageous since the hacker will find it difficult to locate the pixels with encoded information because all of the pixels have noise. The system is set up such that only the user with whom the shared login credentials have been shared can access the image. Instead of placing each bit in one of the three RGB components of a pixel, they place a secret message byte in place of one of the three bytes. This keeps them from discovering the number of bits hidden in a pixel and the location of the message bits.[12]

In some cloud systems, it is necessary to protect data from hacking or loss when uploading to the cloud. This research paper uses client-side encryption through the combination of AES and Secure Hash Algorithm with Initial Vector (see Fig 16). It also discusses various cloud issues such as data loss, data breaches, insecure APIs, malicious insiders, and account hacking. The output of SHA-256 with the password is used as the secret key to the AES algorithm. As it's symmetric the same process is used for decryption. This emphasizes the usage of client-side encryption of data before uploading it to the cloud as the security of the data depends on the cloud service provider [13]. This research paper gives a review of the various challenges in secure cloud storage. This study examines the methods currently used in cloud security to assess the three key factors of integrity, authentication, and confidentiality. Some of the major challenges in cloud storage are data leakage, access credentials, the performance of encryption and decryption, and data security transmission [14]. While client-side encryption for cloud storage security could be a great solution to tackle some of the problems such as data breaches and could be also used for data hiding. There could be also a size overhead when doing encryption so this paper uses a GZIPSTREAM algorithm for the compression of the data since it could be large[15]. As discussed above LSB method is one of the most commonly used methods for image steganography this research paper uses a modified LSB approach. They propose using a 24-bit RGB image for steganography. They take 4 LSBs of RGB attributes for encryption. A mapping function is used for the selection of LSBs. The mapping function is responsible for adding variations in the array of selections. The mapping function employed here is a modulo function. Which is then combined with RSA encryption [16]

Implementation

Fig 1 flowchart illustrates how the proposed system provides functionality to the user. In this flowchart, it is shown how would the whole process work and what are alternate options through which the user can choose giving him different results and giving an idea of what would the final result look like.

It has mainly three steps: -

1. Image Selection
2. Data Hiding
3. Decryption

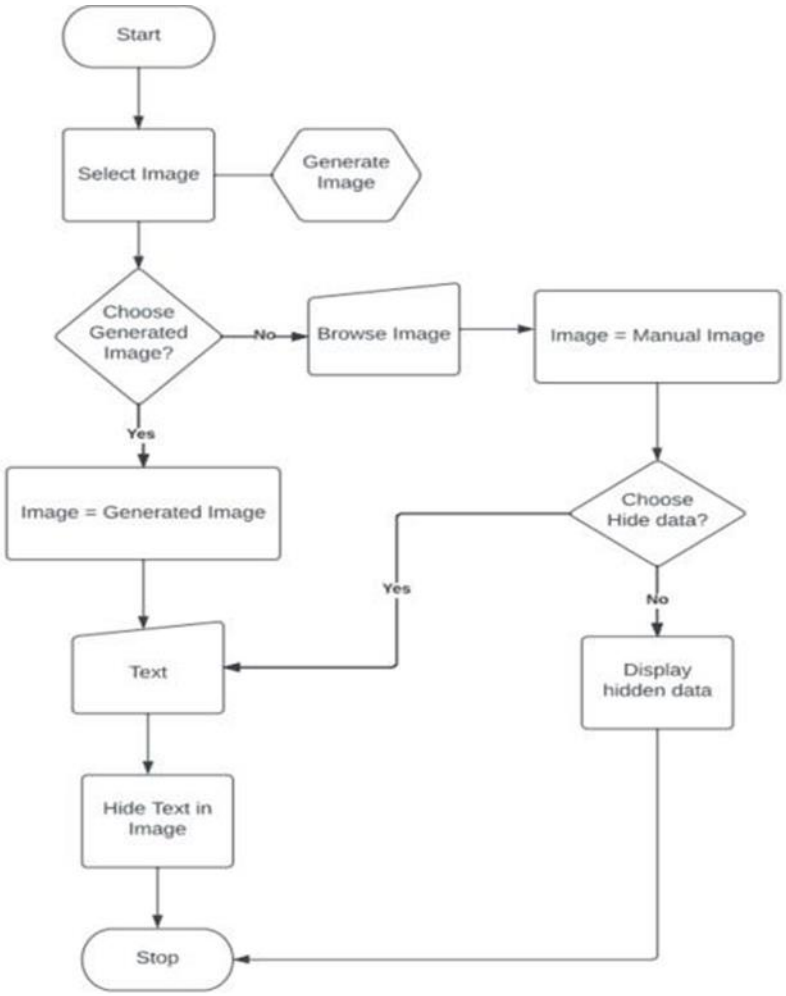

Fig 1: Flowchart

The whole process is divided into 3 parts: -

A) Image selection: There are two ways of using an image. Firstly, the user can browse the machine to use the desired image. The second way is to for the user to select a random image created by our software. The proposed system uses Easy Path Wavelet Transform to generate a random image that contains some pattern.

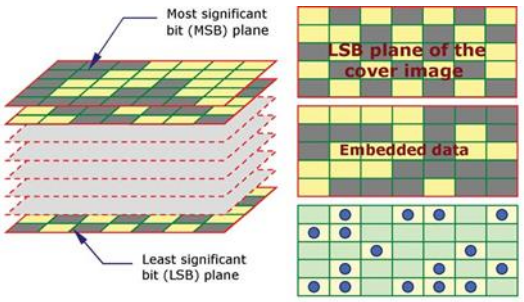

Fig 2: LSB approach

B) Data hiding: The main goal of this stage is to conceal the input data inside the image. For this, a technique that uses the image's LSBs (Least Significant Bits) for data concealing is used. In this method, the data bits are stored inside the last bits of each pixel so that they remain hidden while keeping the image precise as it was before possible. The change is impossible to detect by the human eye.

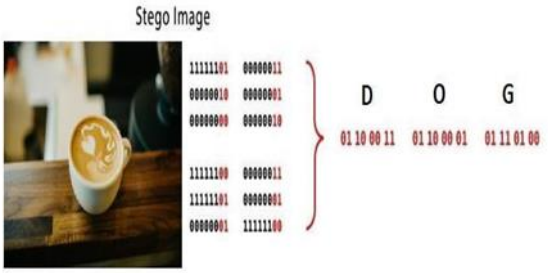

Fig 3: LSB method - Encryption

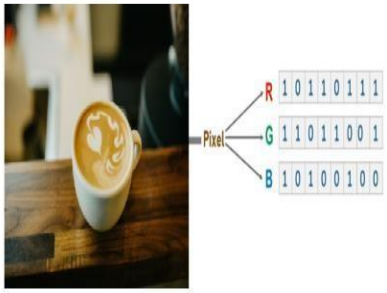

Fig 4: LSB method - Decryption

C) Decryption: For the decryption process the same method is used which was used for data hiding. In this basically, the LSBs (Least Significant Bits) from the image are extracted and then they are converted into text.

The formula for the LSB is LSB=FSR/2N.

which is frequently used for commercial devices, is defined in the IEEE Standard 1241-2010. So, it can be regarded as the appropriate definition.

In integrated circuit design, it can occasionally make sense to define the LSB differently (LSB=FSR/(2N1)) when creating an ADC as a component of a signal processing chain or as a smaller ADC.

The explanation is that, depending on the code transition levels applied, there are two protocols for ADCs. One is the so-called "mid-tread convention," where the initial transition happens at LSB/2 and FSR/2 is located directly in the center of a code. The second convention is the mid-riser convention, where the first transition happens at LSB and FSR/2 occurs at transition.

Here are the transfer functions for both types; the solid lines denote the mid-tread type's range, while the dashed lines denote the mid-riser type.

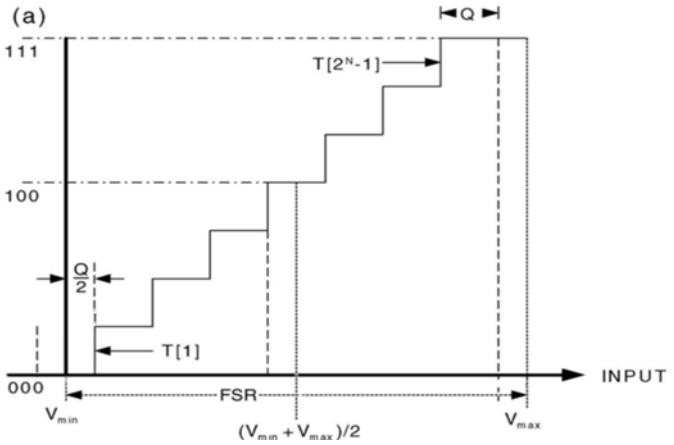

Fig 5: LSB working with Time constraints

According to the graph, the initial transition for the mid-tread type is at 1/2 LSB, whereas the last transition takes place at 3/2 LSB below FSR. Sometimes the last transition is made 1/2 LSB below the maximum voltage in order to create a symmetric transfer function. Hence, the top end loses one LSB.

The LSB in this scenario would be: FSR/(2N-1).

IV. Experimental Results

The system described below in fig 4 is developed using python as the backend as well as the frontend. In the backend, the framework used is Flask to connect with the frontend which is made using HTML, CSS, Javascript, and ReactJS, and the backend is made using Python, and python libraries like, matplotlib, cryptography, RSA, etc., The suggested system's system design is shown in the picture below.

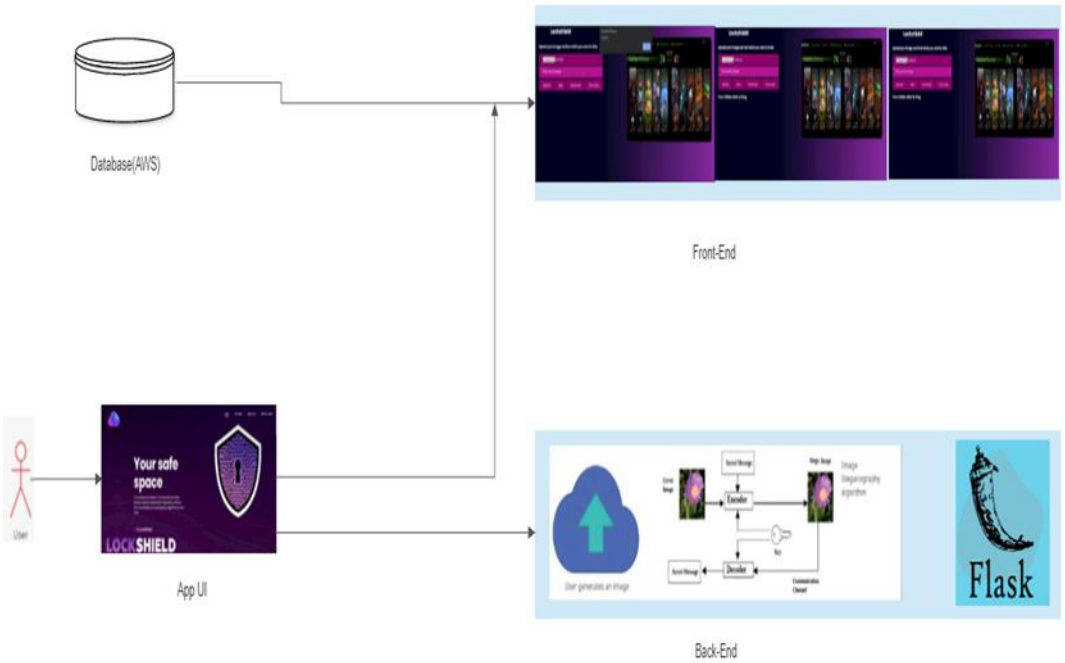

Fig 6: System Design

In fig 7, it is the home page of the LockShield platform, where one can read about the working of the app and learn about the basic instructions. It makes the user familiar with the platform.

Fig 7: Homepage

As we can see in fig 8 the user has selected to generate a random image as well as also given the secret message. Other buttons which can be seen on the screen are the upload button, with the help of which the user can submit the image to the server. Also, it has a Hide button, with which desired information can be hidden in the given image and also download the image. After this with the help of the Show date, button one can see the data which was encrypted in the image. The software successfully generates an image or lets the user choose the image. Afterward, the user can write data and hide it onto the image and can also decrypt it later. This is the app UI, where all the basic functionalities can be previewed and further action can be taken.

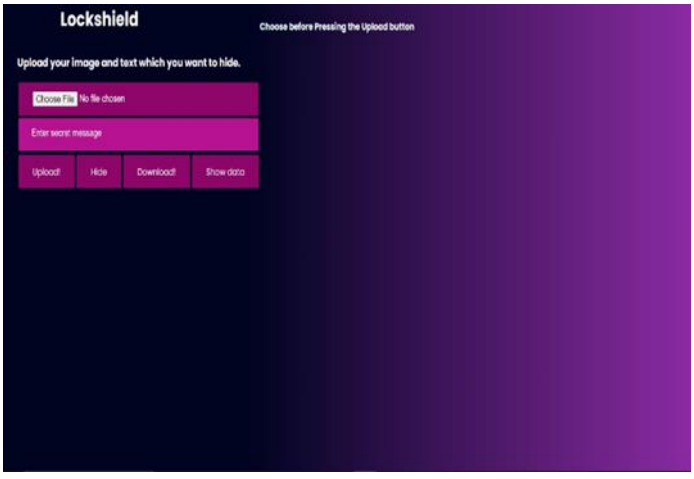

Fig 8: Application User Interface

Fig 9 demonstrates the working of GUI as it also gives the user comprehensive feedback about what is going on. The image is successfully saved with the data hidden in it. Here, also the notification that image is successfully uploaded is also received.

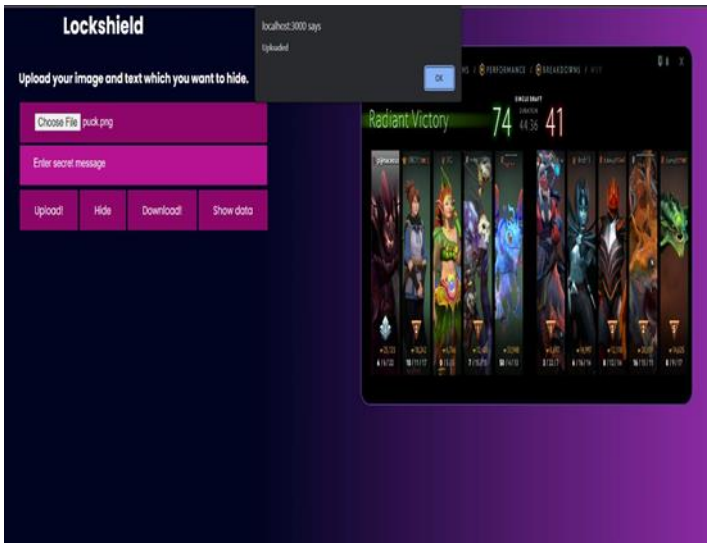

Fig 9: Hiding the data

In fig 10, after clicking the Show data button, the secret message is displayed, showcasing its functionality. This message is securely stored in a particular image and can be viewed only by accessing that image only.

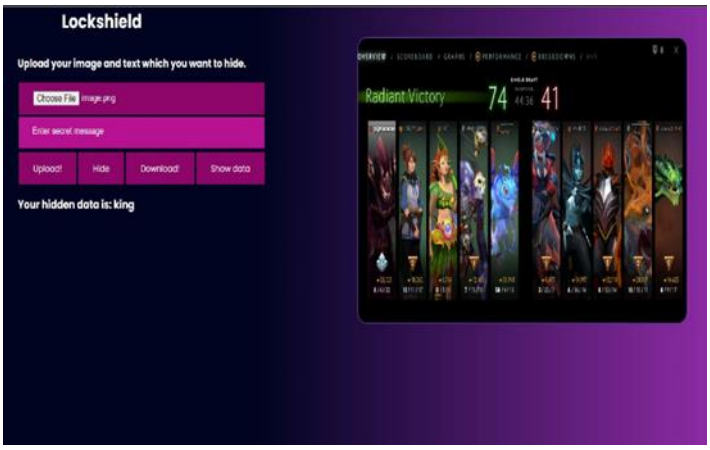

Fig 10: Showing the data

CONCLUSION

Hence, we can conclude that our method is more secure than most of the existing systems, we use a double layer of protection in case one fails. First in-text cryptography, where the text is saved in encrypted form and can only be decrypted by the private key. After that, the encrypted message is stored in the image using the LSB approach, because of which it is very difficult for any person to identify the image from any other normal image. We have used services like React and AWS because you may choose the operating system, programming language, web application platform, database, and other services you require there, and react will effectively update and render the precise components as your data changes. You get a virtual environment through Amazon that you may fill with the programmes and services your application needs. This type of platform can be used to share images that contain sensitive information like credit cards, bank details, and property papers which are at risk of being viewed by a hacker if sent in raw form.

REFERENCES

[1] S. K. S. Raja, A. Sathya, and L. Priya, "A Hybrid Data Access Control Using AES and RSA for Ensuring Privacy in Electronic Healthcare Records," 2020 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS), 2020, pp. 1-5, doi: 10.1109/ICPECTS49113.2020.9337051.

[2] K. M. Prabha and P. V. Saraswathi, "TIGER HASH KERBEROS BIOMETRIC BLOWFISH USER AUTHENTICATION FOR SECURED DATA ACCESS IN CLOUD," 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I- SMAC)I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on, 2018, pp. 145-151, doi: 10.1109/I-SMAC.2018.8653713.

[3] V. P. Lalitha, M. Y. Sagar, S. Sharanappa, S. Hanji and R. Swarup, "Data security in cloud," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017, pp. 3604-3608, doi: 10.1109/ICECDS.2017.8390134.

[4] A. Musa and A. Mahmood, "Client-side Cryptography Based Security for Cloud Computing System," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), 2021, pp. 594-600, doi: 10.1109/ICAIS50930.2021.9395890.

[5] R. Jain and J. Boaddh, "Advances in digital image steganography," 2016 International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH), 2016, pp. 163-171, doi: 10.1109/ICICCS.2016.7542298.

[6] M. A. Islam, M. A. -A. K. Riad and T. S. Pias, "Enhancing Security of Image Steganography Using Visual Cryptography," 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2021, pp. 694-698, doi: 10.1109/ICREST51555.2021.9331225.

[7] Ashutosh and S. D. Sen, "Visual Cryptography," 2008 International Conference on Advanced Computer Theory and Engineering, 2008, pp. 805-807, doi: 10.1109/ICACTE.2008.184.

[8] A. M. Qadir and N. Varol, "A Review Paper on Cryptography," 2019 7th International Symposium on Digital Forensics and Security (ISDFS), 2019, pp. 1-6, doi: 10.1109/ISDFS.2019.8757514.

[9] H. Arora, C. Bansal and S. Dagar, "Comparative study of image steganography techniques," 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2018, pp. 982-985, doi: 10.1109/ICACCCN.2018.8748451.

[10] V. Sharma and Madhusudan, "Two new approaches for image steganography using cryptography," 2015 Third International Conference on Image Information Processing (ICIIP), 2015, pp. 202-207, doi: 10.1109/ICIIP.2015.7414766.

[11] R. S. Phadte and R. Dhanaraj, "Enhanced blend of image steganography and cryptography," 2017 International Conference on Computing Methodologies and Communication (ICCMC), 2017, pp. 230-235, doi: 10.1109/ICCMC.2017.8282682.

[12] K. C. Nunna and R. Marapareddy, "Secure Data Transfer Through Internet Using Cryptography and Image Steganography," 2020 SoutheastCon, 2020, pp. 1-5, doi: 10.1109/SoutheastCon44009.2020.9368301.

[13] M. M. Islam, M. Z. Hasan and R. A. Shaon, "A Novel Approach for Client Side Encryption in Cloud Computing," 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), 2019, pp. 1-6, doi: 10.1109/ECACE.2019.8679151.

[14] R. Kumar and M. P. S. Bhatia, "A Systematic Review of the Security in Cloud Computing: Data Integrity, Confidentiality and Availability," 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON), 2020, pp. 334-337, doi: 10.1109/GUCON48875.2020.9231255.

[15] K. Rani and R. K. Sagar, "Enhanced data storage security in cloud environment using encryption, compression and splitting technique," 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), 2017, pp. 1-5, doi: 10.1109/TEL-NET.2017.8343557.

[16] S. Pramanik, D. Samanta, S. Dutta, R. Ghosh, M. Ghonge and D. Pandey, "Steganography using Improved LSB Approach and Asymmetric Cryptography," 2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI), 2020, pp. 1-5, doi: 10.1109/ICATMRI51801.2020.9398408.