Journal of Computer Science

Original Research Paper

Efficient Text Data Hiding Technique Using Cryptography and Steganography

Nor Fazlida Mohd Sani and Mohamad Adreen Nujjaid

Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang, Selangor, Malaysia

Article history
Received: 18-07-2024
Revised: 03-09-2024
Accepted: 10-09-2024

Corresponding Author:
Nor Fazlida Mohd Sani
Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang, Selangor, Malaysia
Email: fazlida@upm.edu.my

Abstract: Steganography serves as a technique for covertly embedding confidential information within other data, ensuring that both the existence and content of this hidden data remain undisclosed to unauthorized parties. Its applications span diverse domains, including military operations, online payment systems, and the Internet of Things (IoT). Steganography proves highly adaptable in concealing data, as information with an unknown presence escapes detection by potential attackers. Nonetheless, recent developments in steganalysis and image processing techniques have made it simpler for adversaries to reveal secret information. In response to this challenge, researchers have suggested various countermeasures, such as integrating steganography with encryption. This study employs steganography utilizing the Least Significant Bit (LSB) approach, which uses the Advanced Encryption Standard (AES) for symmetric encryption, and the Rivest-Shamir-Adleman (RSA) method for asymmetric key encryption. The Diffie-Hellman key exchange method improves the procedure as a whole. The effectiveness of these methods in protecting and hiding confidential information is then evaluated, along with how quickly the data is hidden. According to the results, using Diffie-Hellman key exchange in conjunction with AES performs better than using RSA for the encryption technique in terms of data hiding capabilities and speed.

Keywords: Cryptography, Steganography, Key Exchange Algorithm

Introduction

Ensuring secure data transmission holds significant importance in the contemporary landscape. The development of the Internet has made it easier to exchange different types of data, such as text, photos, audio, and video. Data confidentiality and privacy are pivotal considerations in the realm of data transmission (Santoso *et al*., 2018). Several types of data possess varying degrees of confidentiality, requiring the implementation of appropriate data security measures like encryption and data hiding for protection. Steganography is an approach that falls under the category of data-hiding techniques (Sahu and Sahu, 2020). The process of hiding data, including text, pictures, audio, and video, inside another set of data-known as a cover image is known as steganography (Bandekar and Suguna, 2018). Typically, this method comprises three key elements: The data itself, the carrier of the data, and a secret key. Unlike encryption, which does not hide the presence of processed data, steganography focuses on concealing both the meaning and existence of the data (Hasan *et al*., 2021). This characteristic makes steganography particularly attractive for scenarios where both aspects of data significance and presence need to be kept concealed.

Steganography finds diverse applications across military, banking, Internet of Things (IoT), and e-commerce domains. Xin *et al*. (2018) suggest utilizing audio steganography for confidential communication, especially in military or government contexts. Liashenko *et al*. (2018) conducted an investigation into the potential applications of steganography for remote biometric authentication, evaluating different network-based steganography algorithms. Douglas *et al*. (2018) carried out a more thorough investigation of steganography applications in biometric data. Additionally, Khari *et al*. (2020) implement steganography in the Internet of Things using Matrix XOR encoding. On the darker side, IEEE Computer Society's Computer Magazine (2016) notes the use of steganography in crafting malware like Zbot, Adgholas, and Cerber. These diverse applications underscore steganography's widespread use in both data security and security attacks.

The process employed to identify the presence of steganography is referred to as steganalysis and technological advancements have enhanced the tools for steganalysis (Hasan *et al*., 2021). In this study, the LSB method is employed for steganography and it is coupled with encryption algorithms to prevent exposure by steganalysis tools. The Least Significant Bit (LSB) approach is the most basic kind of steganographic technique, where the least significant bit of the image is replaced with the bits of the hidden data. The selected encryption algorithm, RSA, has been in use since its debut as a public key cryptography technique in 1977 and is a lightweight approach appropriate for text encryption. The three main components of the RSA algorithm are message encryption, message decryption, and key generation. Furthermore, the Diffie-Hellman algorithm is used in conjunction with the AES encryption system for the key exchange procedure. All of these techniques are well-known and regarded as sufficiently safe for text data encryption.

Confidentiality of data is a vital component of security. The most often used type of data is text. Secret data such as passwords or hidden messages require suitable data security techniques to ensure that the data is protected. The data and its presence are hidden from anybody but the sender and the recipient through the use of steganography.

There are many advantages of using steganography. By keeping the message and its presence hidden, this approach adds another layer of security to the data. However, using the steganography technique alone may be disadvantageous to the data owner. The advancement of steganalysis tools has caused traditional steganography techniques to be less secure. The primary considerations for this technique's text data security are the quality of the cover image to hide it from an attacker and the speed at which the data is hidden to guarantee that the text data is secured from transmission to the recipient.

Al Saffar (2019) study suggested combining RSA and steganography to secure text data with cryptography and steganography. In Al Saffar's research, no detailed quantitative performance measurement has been done. While the implementation was successful, Al Saffar (2019) suggested that a different algorithm could be created to improve the cover image's text data concealment mechanism and data hiding speed for the proposed technique.

Related Works

Data Security

The abundance of data transmitted daily has invoked security concerns by data owners. Tawalbeh and Saldamli (2021) have reviewed the data privacy issues in big data and proposed methods to countermeasure this issue. In IoT, system components such as networks, sensors, and back-end systems are listed as the main security concern factors (Suresh Babu *et al*., 2018). Some other domains where data security concern is addressed are e-business, healthcare, banking, social networking and governance (Rath and Kumar, 2021).

Based on the research conducted by Sahu and Sahu (2020), there are two main data security techniques, which are cryptography and data hiding techniques. Two main methods that are classified as data-hiding techniques are watermarking and steganography. Hussain *et al*. (2018) stated that steganography allows both the existence of communication and the data to be concealed while maintaining the complexity of the method.

There are two types of steganography, which are text and image (Ali Khodher and Aldeen Khairi, 2020). Image steganography methods are spatial domain, spread spectrum, and distortion technique. In the spatial domain, methods like Pixel Value Differencing (PVD), Least Significant Bit (LSB), and replacement are employed. In terms of performance, LSB offers high capacity, medium robustness, and high precision, according to a comparison of different approaches in the same research.

Steganography in Data Security

There are several types of research related to steganography. Ramya *et al*. (2018) have studied LSB and DWT-based steganography for images and audio. Mukherjee *et al*. (2018) developed an image steganography technique based on Mid Position Value (MPV). Arnold's transformation is used to jumble the cover image before the concealed message is embedded. In another research, the LSB method is combined with the image compression technique that is conducted after hiding the secret message. The data recipient will then decompress and then decrypt this secret message (Pandey *et al*., 2021).

The majority of recent research focuses on combining steganography with encryption approaches. This is done in an effort to make the suggested procedure more difficult. Modified RSA has been implemented in steganography by Majumder and Rahman (2019). Some of the disadvantages of this technology are that it requires high-resolution photos, can only be used with grayscale images, and requires a transmission protocol for both the original image and the image carrying secret data.

Another research conducted has combined the usage of the Dual RSA and Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT) steganography method (Vinothkanna, 2019). This combination is called Cryptography Incorporated Steganography (CICS). RSA algorithm is used to encrypt secret text before hiding the data inside a .bmp image file (Nassif Jassim *et al*., 2019).

This technique was also developed by Al Saffar (2019) to hide the concealed plain text in the cover image. In this research, Al Saffar has suggested that implementing the Diffie-Hellman algorithm with the existing technique may improve the data hiding complexity. The research listed above has proved that the RSA algorithm is a suitable combination with steganography.

It is observed that some steganography combinations with encryption algorithms are using the Advanced Encryption Standard (AES) algorithm. According to reports, AES performs better than RSA in terms of complexity and security (Hindi *et al*., 2019). The LSB approach and AES algorithm serve as the foundation for the picture watermarking method known as mark to lock (Cirineo *et al*., 2017). An extra image is hidden inside the cover image in this study. The average file size increase for the majority of the resultant images is 56.208%.

Siddalingesh and Manjunatha (2019) have used steganography with AES to hide text and graphics in audio. The DCT coefficient is used to carry out the procedures. The LSB method and AES were then combined by Abood and Taha (2019). This study shows good Peak Signal Noise Ratio (PSNR) values for the.png file. The original cover image and the histograms generated for the steganographic image are almost the same.

Fundamental Concepts

In addition to studying past research papers on the related topics, the fundamental concepts that are to be applied to this research are also studied. Firstly, the concept of RSA which is used in the anchor paper by Al Saffar (2019) is investigated. According to Wahab *et al*. (2021), RSA is a symmetric key encryption scheme that depends on the rigidity of the analysis of many compounds and a complex number for a given odd integer. Two integers make up the RSA public key. RSA is a well-liked encryption technique with several applications (Al Saffar, 2019).

The cover image is created using LSB, a simple data-hiding technique that involves successively inserting the bits of the secret message into the least significant bit of each byte of pixels (Mahdi *et al*., 2019). There are two types of LSB methods, which are LSB replacement (LSBR) and LSB matching (LSBM) (Fateh *et al*., 2021).

The improved approach in this study entails building upon the current LSB data hiding methodology by combining the AES encryption algorithm with Diffie-Hellman key exchange. The Diffie-Hellman key exchange method involves the sender and the recipient sharing a public key. This public key is then used to calculate a secret key that may be used to both encrypt and decrypt the secret message. Modulus and inverse calculations are part of this process (Gupta and Subba Reddy, 2022).

The symmetric cryptography block cipher approach known as Advanced Encryption Standard (AES) generates round keys from the cipher key using the AES key schedule mechanism. Bit-shifting, substitution, and XOR operations are a few of the operations carried out in AES (Ripon, 2021).

Based on recent trends, steganography is usually paired with encryption algorithms to ensure that the attackers are unable to gain meaning from the data if it is obtained during transmission. AES performs better in terms of time, encryption outcomes, and image quality. However, RSA is less complex and is more suitable for a simple form of data such as text. It is observed that there is a possibility to further improve the complexity of encryption to steganography so that it provides the critical factors needed to be secure in text such as confidentiality and integrity.

Materials and Methods

As highlighted by a review of the literature, the research approach employed here focuses on exploring and applying previously established findings. This research will apply and improve the method proposed by Al Saffar (2019). For asymmetric encryption, the RSA encryption algorithm is used with the LSB steganography technique. Also, the LSB steganography method combined with the AES encryption algorithm and Diffie-Hellman key generation algorithm for symmetric encryption. The basic flowchart for the proposed method process by Al Saffar (2019) is shown in Fig. (1).

The secret text message will be encrypted using RSA. Figure (2) illustrates the processes for the RSA encryption technique.

Then, the cover image is interlaced with the encrypted text using the LSB technique. LSB is the process of swapping out the image's least significant bit with the concealed data bits. A transmission of the LSB output will be sent to the receiver.

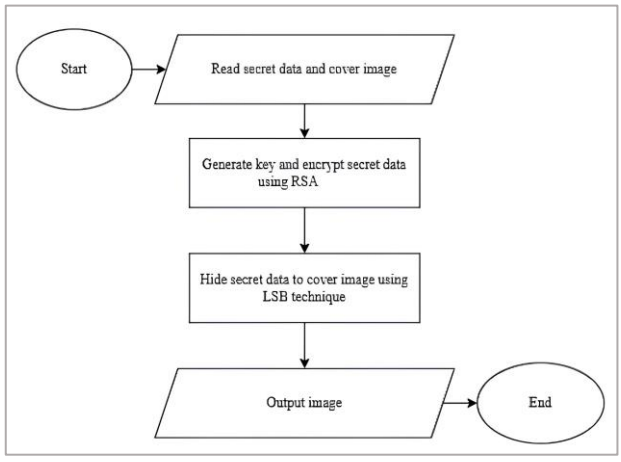

Fig. 1: Asymmetric algorithm flowchart for data hiding

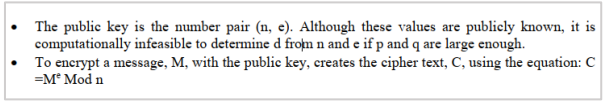
- The public key is the number pair (n, e). Although these values are publicly known, it is computationally infeasible to determine d from n and e if p and q are large enough.
- To encrypt a message, M, with the public key, creates the cipher text, C, using the equation: $C = M^e \text{ Mod } n$

Fig. 2: RSA message encryption process

Data extraction begins as soon as the recipient receives the steganography image. The flowchart of the conducted process is illustrated below in Fig. (3).

RSA Encryption Algorithm

The RSA encryption algorithm is taken from the complete MATLAB library that has also been used by the researchers. Some improvisation is done in order to fit this research's purposes.

Key Generation Using Asymmetric Key Algorithm

Two types of keys in RSA: Private key and public key. According to Rivest *et al.* (1978), Each station selects two large primes at random and on its own. p and q number and multiplies them to produce $n = pq$. This is the modulus used in the arithmetic calculations of the RSA algorithm. The algorithm for key generation is given as:

1. Select p and q (both prime numbers)
2. Calculate $n = pq$
3. Calculate $z = \varphi(n) = (p-1)(p-1)$
4. Select integer e, $\gcd(\varphi(n), e) = 1$; $1<e<\varphi(n)$
5. Calculate d, $e = 1 \bmod \varphi(n)$

RSA Image Encryption Scheme

The RSA image encryption processes are applied from the MATLAB library module. The flow of the encryption algorithm can be described as in RSA encryption is done with the public key to generate the cipher text. The algorithm for encryption is given as:

$$1.\ C = Pe\ mod\ n$$

where, C is ciphertext, P is plaintext.

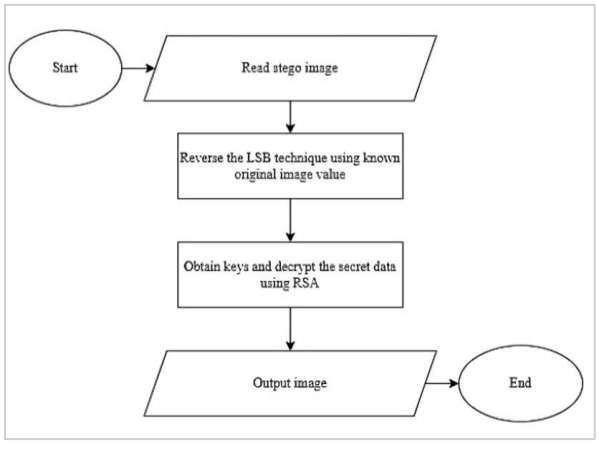

Fig. 3: Data extraction flowchart for the asymmetric algorithm

Only square photos could be used in the original study by Al Saffar (2019). This encryption algorithm can also be used to encrypt photos with aspect ratios other than 1:1 for this study. However, there is a 512×512-pixel limit on image resolution.

RSA Image Decryption Scheme

The greyscale image is the output of the original image. The image decryption process is reversed from the encryption process. Since the main usage of this algorithm is text data only, a greyscale image is sufficient and can help reduce storage capacity.

In RSA, decryption is done with the private key to get the plain text. The algorithm for decryption is given as:

$$1.\ P = Cd\ mod\ n$$

where, P is plaintext, *and* C is ciphertext.

The combination of the LSB steganography method with the AES encryption algorithm and Diffie-Hellman key generation algorithm for symmetric encryption. The basic flowchart for the new proposed process is shown in Fig. 4.

Diffie-Hellman Key Exchange Algorithm Process

After the Diffie-Hellman algorithm completes the key creation process, the encrypted text is interlaced with the cover picture using the LSB technique. The practice of replacing the image's least significant bit with the hidden data bits is known as LSB. The output of LSB will be given to the recipient. Once the recipient has the steganography image, the same data extraction process is carried out. The process flowchart that was used is displayed in Fig. 5.

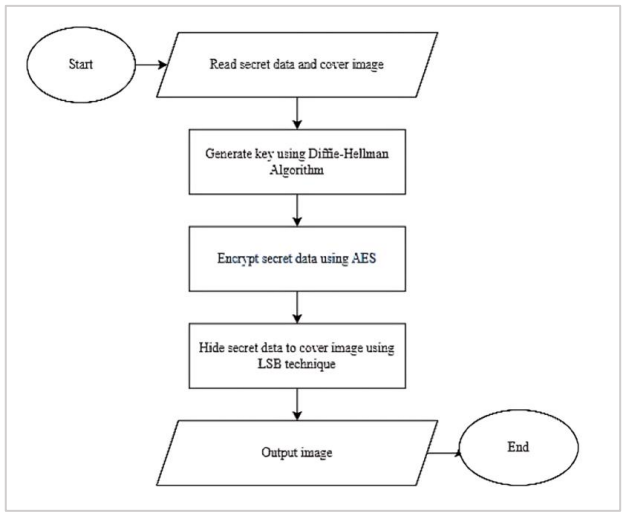

Fig. 4: Symmetric algorithm flowchart for data hiding

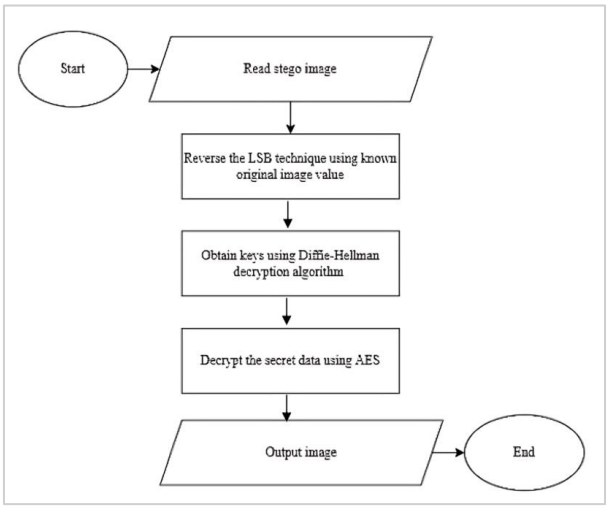

Fig. 5: Data extraction flowchart for a symmetric algorithm

AES Encryption Algorithm

The AES encryption algorithm is taken from the MATLAB library that is provided by the researchers. Some changes are made in order to fit this research's purposes.

Key Generation Using Symmetric Key Algorithm

The encryption algorithm generates a secure key exchange over a public channel using the Diffie-Hellman key generation algorithm. This algorithm is taken from the MATLAB library and some improvements are done to fit in with this research.

The algorithm for Diffie-Hellman key generation is given as:

1. Sender and receiver generate prime number p, q as its primitive root
2. Sender and receiver choose private key 'a' and 'b'
3. Sender's public key $A = q^a \bmod p$
4. Receiver's public key $B = q^b \bmod p$
5. Sender and receiver exchange public key
6. Sender calculates $B^a \bmod p = q^{ba} \bmod p = S$
7. The receiver calculates $A^b \bmod p = q^{ba} \bmod p = S$
8. The sender and receiver get 'S' as a shared secret key

AES Image Encryption and Decryption Scheme

The AES image encryption and decryption processes are applied from the MATLAB library provided.

The AES encryption algorithm is given in Fig. 6.

The input state array is XORed before any round-based processing for encryption can start. Each round consists of the following four steps for encryption: Substitute bytes, shift rows, mix columns and add a round key. Lastly, XOR is the output of the previous three steps.

The AES decryption algorithm is given as.

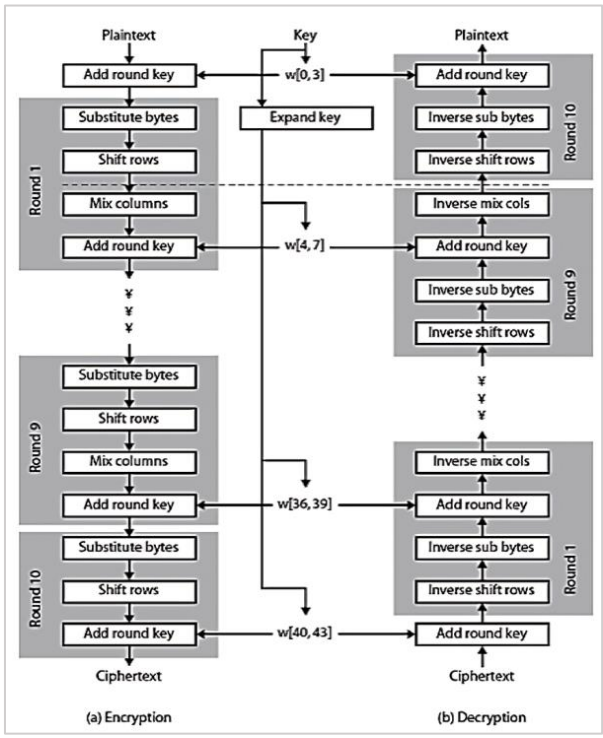

Fig. 6: Encryption and decryption flowchart for AES algorithm

XOR is the ciphertext state array. Each round consists of the following four steps for decryption: Inverse shift rows, inverse substitute bytes, add round key, and inverse mix columns. Lastly, we XOR the output of the previous two steps.

The overall process of encryption and decryption of the AES algorithm was summarized by Abdullah (2017) (Fig. 6).

Image sizes for this study are restricted to square images, yet this encryption technique can also encrypt images with aspect ratios other than 1:1. Nevertheless, the maximum pixel resolution for the image is 512×512.

Results and Discussion

Image Steganography

The performance of hiding text secrets using steganography by implementing different types of key generations and encryptions before using LSB to interlace the cover image and secret data. The efficiency of these approaches is compared using data concealing speed, Mean Square Error (MSE), Peak signal-to-noise ratio (PSNR), and histogram analysis.

Histogram Analysis

Steganography histogram analysis is used to show the differences between the stego and cover images. Unlike histogram analysis in picture encryption, the histogram plot for the original image and the stego image in

steganography must be as similar as possible. Attackers or data interceptors might not be able to spot the image's concealed message if the histogram plots appear to be comparable. A 512×512 image of Lenna is utilized in this study. Before using any further methods, the image is transformed to greyscale. The two axes of the histogram plot are the pixel intensity value on the x-axis and the pixel count on the y-axis.

The histogram graphs demonstrate that the values for the stego image and cover image for both techniques that is LSB and RSA (Fig. 7) and LSB, AES, Diffie-Hellman (Fig. 8) do not differ much. Since the cover and stego image cannot be visually distinguished, both techniques are able to hide the secret content.

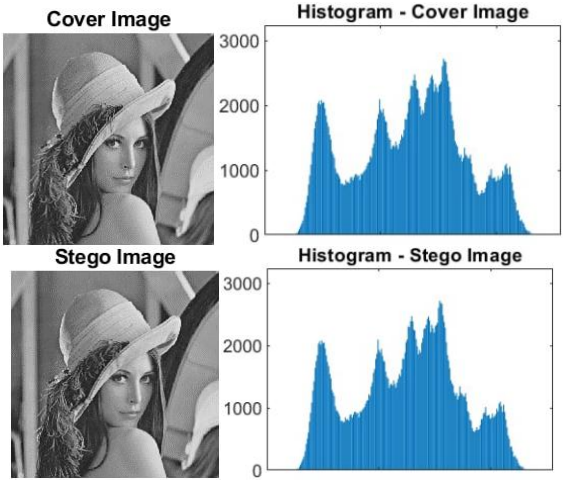

Fig. 7: Histogram analysis for stego and cover images utilizing LSB and the RSA encryption technique

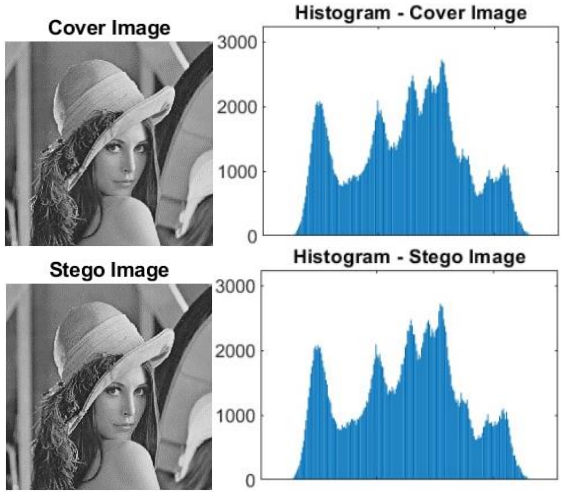

Fig. 8: Histogram analysis for stego and cover images using LSB, AES encryption, and the Diffie-Hellman key exchange technique

Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE)

A technique to assess the quality of the image recovered from the steganography image is the *PSNR* value. The output image quality increases with the *PSNR* value. The formula below can be used to determine the *PSNR* value:

$$PSNR = 10\,\mathrm{Log}_{10}(R^2 \div MSE)$$

MSE is the image's mean-square-error value and *R* is the greatest fluctuation in the input image data.

The difference between the stego image (which has a secret message embedded) and the original image is indicated by the *MSE* value. The smaller the difference between the original and stego images, the lower the *MSE* value. The following formula can be used to determine *MSE*:

$$MSE = \frac{1}{MXN}\sum_{i=1}^{M}\sum_{j=1}^{N}[\,C(i,j) - S(i,j)\,]^{2}$$

M and *N* are the rows and columns of the cover image respectively and *C* (*i*, *j*) and *S* (*i*, *j*) mean the pixel value at position (*i*, *j*) in the cover image and the corresponding stego image, respectively.

Both encryption methods have high PSNR values (Table 1), indicating that the quality of the generated image is good. There is only a 0.03% difference in the PSNR value calculated from the two algorithms shown above. MSE values are directly correlated with PSNR values for the given algorithm comparison. When AES, Diffie-Hellman, and LSB techniques are combined, the PSNR value is higher and the MSE is lower than when RSA and LSB techniques are combined.

Data Hiding Speed

Processing time is crucial for real-time steganography. A successful steganography algorithm must not only perform well in protecting the secrecy of the data but also maintain a good processing speed. The MATLAB code is run on a machine with Intel Core i7 8th Gen processor and 8GB RAM.

Table 1: Comparison of PSNR and MSE values for different steganography methods

Algorithm	Image	MSE	PSNR (%)
LSB steganography + RSA encryption algorithm	Lenna png	0.0006 1289	81.35
AES encryption algorithm + Diffie-Hellman key exchange algorithm + LSB steganography	Lenna png	0.0006 0145	81.38

Table (2) shows the time taken for the combination of AES, Diffie-Hellman key exchange algorithm, and LSB technique is lesser than the time taken for the combination of RSA and LSB technique by 0.1175 sec. This shows that the former technique is more efficient in terms of speed. The clearer comparison of all the results for both algorithms is illustrated in the graphs below (Figs 9-11).

Table 2: Time taken for different steganography techniques

Algorithm	Image	Time taken (s)
LSB steganography + RSA encryption algorithm	Lenna png	0.3335
AES encryption algorithm + Diffie-Hellman key exchange Algorithm + LSB steganography	Lenna png	0.2160

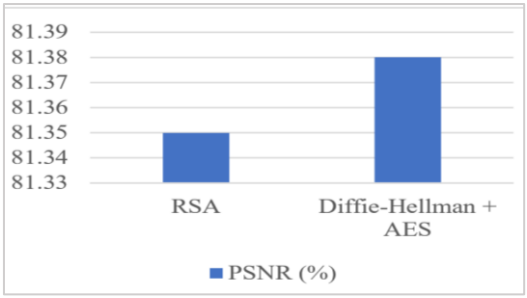

Fig. 9: PSNR value comparison

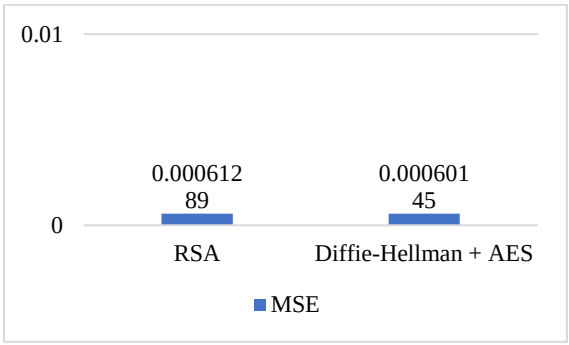

Fig. 10: MSE value comparison

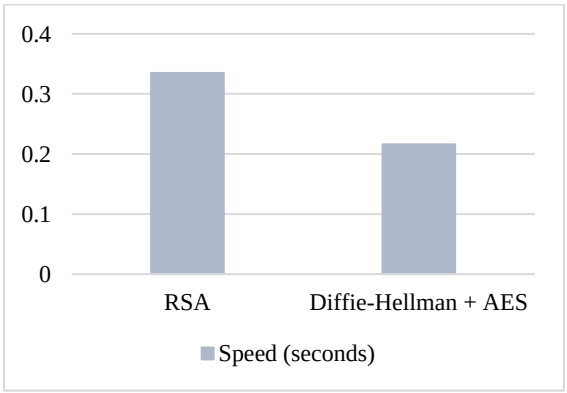

Fig. 11: Speed comparison

Expected Level of Security

In a brute force attack on steganographic techniques, every potential stego key is methodically tested until the right one is found, exploiting the vast number of potential predefined patterns from diverse sources such as web images, mobile devices, and computers. These attacks are further complicated by the need to understand the size of the hidden message, which is related to the secret pattern used. The secret text is embedded randomly based on color distribution and pixel matching, making retrieval challenging. This study introduces an improvement method to steganography that builds on the Least Significant Bit (LSB) technique, incorporating AES encryption and Diffie-Hellman key exchange algorithms to conceal text within an image. By using pixel selection randomization and a mask as an encryption key and bit-shifting tool, the suggested approach overcomes the drawbacks of conventional LSB approaches. This approach not only enhances security by making it harder to detect and extract hidden text but also reduces visual distortions. Future improvements will concentrate on improving the user interface to make it more user-friendly and adaptable, testing and strengthening resistance against cryptanalysis and steganalysis attacks using machine learning and artificial intelligence, and optimizing the encryption and compression algorithms to handle larger messages more effectively. By using a secret key generated from the image and breaking the image up into smaller blocks, the technique successfully overcomes the constraints of conventional LSB, thereby adding layers of security and reducing visual impact. Future work should explore optimizing mask dimensions, employing more complex encoding methods, and developing a more user-friendly interface with advanced control options.

Conclusion

In conclusion, a combination of encryption algorithms and steganography can help improve the performance of existing steganography techniques. Diffie-Hellman key exchange algorithm is employed in this study to increase the difficulty of the encryption method's implementation. Based on the results, it also can be concluded that the combination of AES, Diffie-Hellman, and LSB techniques is better than the combination of LSB and RSA in terms of PSNR, MSE value, and processing speed. The PSNR value for RSA is 81.35% while the PSNR value for the combination of Diffie-Hellman and AES is 81.38%. The MSE value for RSA is 0.00061289 while the MSE value for the combination of AES and Diffie-Hellman is 0.00060145. Steganography using RSA encryption data hiding speed is 0.3335 sec, which is longer than the combination of Diffie-Hellman and AES which takes only 0.2160 sec. The findings of this study imply that the speed and quality of the cover picture can be enhanced by combining the LSB data hiding technique with Diffie-

Hellman key exchange and AES encryption. Further advancements are expected to strengthen the proposed method's resistance against cryptanalysis and steganalysis by enhanced encryption and compression methods, as well as the application of machine learning and artificial intelligence integration. Additionally, the user interface will be improved for increased customization and usability. Future work will focus on optimizing mask dimensions, implementing faster, lightweight encryption algorithms, and applying compression to manage larger messages. This may be applied or implemented for secure communications, confidential data storage, digital watermarking, or subject to various applications.

Acknowledgment

The author feels grateful to the reviewers for their valuable suggestions and comments toward improving the quality of the paper and would like to thank the editors of the journal from the core of our hearts. Special thanks from authors for publication financial support from the Faculty of Computer Science and Information Technology, Universiti Putra Malaysia.

Funding Information

This research received no financial support from public or private sources.

Author's Contributions

Nor Fazlida Mohd Sani: Supervised study contributed to the written of the manuscript, provided essential guidance and oversight throughout the manuscript.

Mohamad Adreen Nujjaid: Designed the algorithm of the key authentication scheme, provided efficient performance of the algorithm, and prepared the preliminary and final version of this manuscript.

Ethics

This article is original and contains unpublished material. The author confirms no ethical issues are involved.

References

Al Saffar, N. F. H. (2019). Steganography Algorithm Based RSA Cryptosystem. *Journal of Engineering and Applied Sciences*, *14*(7), 2240–2243. https://doi.org/10.36478/jeasci.2019.2240.2243

Ali Khodher, M. A., & Aldeen Khairi, T. W. (2020). Review: A Comparison Steganography between Texts and Images. *Journal of Physics: Conference Series*, *1591*(1), 012024. https://doi.org/10.1088/1742-6596/1591/1/012024

Abood, M. H., & Taha, Z. K. (2019). Secure and Hidden Text Using AES Cryptography and LSB Steganography. *Journal of Engineering Science and Technology*, *14*(3), 1434–1450. https://doi.org/10.13140/RG.2.2.29786.80321

Bandekar, P. P., & Suguna, G. C. (2018). LSB Based Text and Image Steganography Using AES Algorithm. *2018 3rd International Conference on Communication and Electronics Systems (ICCES)*, 782–788. https://doi.org/10.1109/cesys.2018.8724069

Cirineo, C. C., Escaro, R. Q., Silerio, C. D. Y., Teotico, J. B. B., & Acula, D. D. (2017). MarkToLock: An Image Masking Security Application Via Insertion of Invisible Watermark using Steganography and Advanced Encryption Standard (AES) Algorithm. *2017 International Conference on Applied System Innovation (ICASI)*, 995–997. https://doi.org/10.1109/icasi.2017.7988620

Douglas, M., Bailey, K., Leeney, M., & Curran, K. (2018). An Overview of Steganography Techniques Applied to the Protection of Biometric Data. *Multimedia Tools and Applications*, *77*(13), 17333–17373. https://doi.org/10.1007/s11042-017-5308-3

Fateh, M., Rezvani, M., & Irani, Y. (2021). A New Method of Coding for Steganography Based on LSB Matching Revisited. *Security and Communication Networks*, *2021*, 1–15. https://doi.org/10.1155/2021/6610678

Gupta, C., & Subba Reddy, N. V. (2022). Enhancement of Security of Diffie-Hellman Key Exchange Protocol using RSA Cryptography. *Journal of Physics: Conference Series*, *2161*(1), 012014. https://doi.org/10.1088/1742-6596/2161/1/012014

Hasan, F.-K., Elissa, S., Hussin, J. H., Ali, E. D., & Hassan, R. (2021). JPEG Steganography: Hiding in Plain Sight. *International Journal of Forensic Sciences*, *6*(1), 1–11. https://doi.org/10.23880/ijfsc-16000223

Hindi, A. Y., Dwairi, M. O., & AlQadi, Z. A. (2019). A Novel Technique for Data Steganography. *Engineering, Technology and Applied Science Research*, *9*(6), 4942–4945. https://doi.org/10.48084/etasr.2955

Hussain, M., Wahab, A. W. A., Idris, Y. I. B., Ho, A. T. S., & Jung, K.-H. (2018). Image Steganography in Spatial Domain: A Survey. *Signal Processing: Image Communication*, *65*, 46–66. https://doi.org/10.1016/j.image.2018.03.012

Implementation of Diffie-Hellman Algorithm - GeeksforGeeks. (2024). GeeksforGeeks. https://www.geeksforgeeks.org/implementation-diffie-hellman-algorithm/

IEEE Computer Society's Computer Magazine. (2016). *Computer*, *49*(12), 5–5. https://doi.org/10.1109/mc.2016.383

Khari, M., Garg, A. K., Gandomi, A. H., Gupta, R., Patan, R., & Balusamy, B. (2020). Securing Data in Internet of Things (IoT) Using Cryptography and Steganography Techniques. *IEEE Transactions on Systems, Man and Cybernetics: Systems*, *50*(1), 73–80. https://doi.org/10.1109/tsmc.2019.2903785

Liashenko, G., Astrakhantsev, A., & Chernikova, V. (2018). Network Steganography Application for Remote Biometrie user Authentication. *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, 326–330. https://doi.org/10.1109/dessert.2018.8409153

Mahdi, M. H., Abdulrazzaq, A. A., Mohd Rahim, M. S., Taha, M. S., Khalid, H. N., & Lafta, S. A. (2019). Improvement of Image Steganography Scheme Based on LSB Value with Two Control Random Parameters and Multi-level Encryption. *IOP Conference Series: Materials Science and Engineering*, *518*(5), 052002. https://doi.org/10.1088/1757-899x/518/5/052002

Majumder, S., & Rahman, M. M. (2019). Implementation of Security Enhanced Image Steganography with the Incorporation of Modified RSA Algorithm. *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 1–5. https://doi.org/10.1109/ecace.2019.8679147

Mukherjee, S., Roy, S., & Sanyal, G. (2018). Image Steganography Using Mid Position Value Technique. *Procedia Computer Science*, *132*, 461–468. https://doi.org/10.1016/j.procs.2018.05.160

Nassif Jassim, K., Khudhur Nsaif, A., Kuder Nseaf, A., Hazidar, A. H., Priambodo, B., Naf'an, E., Masril, M., Handriani, I., & Pratama Putra, Z. (2019). Hybrid Cryptography and Steganography Method to Embed Encrypted Text Message within Image. *Journal of Physics: Conference Series*, *1339*(1), 012061. https://doi.org/10.1088/1742-6596/1339/1/012061

Pandey, D., Wairya, S., Al-Mahdawi, R. S., Najim, S. A. M., Khalaf, H. A., Al-Barzinji, S. M., & Obaid, A. J. (2021). Secret Data Transmission using Advanced Steganography and Image Compression. *International Journal of Nonlinear Analysis and Applications*, *12*, 1243–1257. https://doi.org/10.22075/ijnaa.2021.5635

Rath, D. K., & Kumar, A. (2021). Information Privacy Concern at Individual, Group, Organization and Societal Level - A Literature Review. *Vilakshan - XIMB Journal of Management*, *18*(2), 171–186. https://doi.org/10.1108/xjm-08-2020-0096

Ripon, P. (2021). PrivateDH: An Enhanced Diffie-Hellman Key-Exchange Protocol using RSA and AES Algorithm. *IACR Cryptology EPrint Archive*, 647.

Ramya, G., Janarthanan, P. P., & Mohanapriya, D. (2018). Steganography Based Data Hiding for Security Applications. *2018 International Conference on Intelligent Computing and Communication for Smart World (I2C2SW)*, 131–135. https://doi.org/10.1109/i2c2sw45816.2018.8997153

Rivest, R. L., Shamir, A., & Adleman, L. (1978). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, *21*(2), 120-126. https://doi.org/10.1145/359340.359342

Siddalingesh, B., & Manjunatha, R. H. S. (2019). Combined Audio Steganography and AES Encryption to Hide the Text and Image into Audio using DCT. *Manjunatha*, *8*(3), 1732–1738. https://doi.org/10.35940/ijrte.c4456.098319

Sahu, A. K., & Sahu, M. (2020). Digital Image Steganography and Steganalysis: A Journey of the Past Three Decades. *Open Computer Science*, *10*(1), 296–342. https://doi.org/10.1515/comp-2020-0136

Santoso, K. A., Fatmawati, & Suprajitno, H. (2018). On Max-Plus Algebra and Its Application on Image Steganography. *The Scientific World Journal*, *2018*(1), 1–9. https://doi.org/10.1155/2018/6718653

Suresh Babu, E., Bhargav Raj, V., Manogna Devi, M., & Kirthana, K. (2018). A Review on Security Issues and Challenges of IoT. *International Journal of Engineering and Technology*, *7*(32), 341–342. https://doi.org/10.14419/ijet.v7i2.32.15708

Tawalbeh, L. A., & Saldamli, G. (2021). Reconsidering Big Data Security and Privacy in Cloud and Mobile Cloud Systems. *Journal of King Saud University - Computer and Information Sciences*, *33*(7), 810–819. https://doi.org/10.1016/j.jksuci.2019.05.007

Vinothkanna, R. (2019). A Secure Steganography Creation Algorithm for Multiple File Formats. *Journal of Innovative Image Processing*, *1*(1), 20–30. https://doi.org/10.36548/jiip.2019.1.003

Wahab, O. F. A., Khalaf, A. A. M., Hussein, A. I., & Hamed, H. F. A. (2021). Hiding Data Using Efficient Combination of RSA Cryptography and Compression Steganography Techniques. *IEEE Access*, *9*, 31805–31815. https://doi.org/10.1109/access.2021.3060317

Xin, G., Liu, Y., Yang, T., & Cao, Y. (2018). An Adaptive Audio Steganography for Covert Wireless Communication. *Security and Communication Networks*, *2018*(1), 1–10. https://doi.org/10.1155/2018/7096271