Jurnal SimanteC
Vol. 14, No. 1 December 2025
P-ISSN : 2088-2130
E-ISSN : 2502-4884

Implementation of Least Significant Bit Steganography with Caesar Cipher Layout Dvorak web-based

[1]Ridhwan Nadif Kurnia, [2]Al Diras Pradiptha, [3]Muhammad Fajar Jati Permana,[4]Kimberly Alfa Di Pradja,[5]Rizki Afrizal,[6]Deden Pradeka,[7]Zahra Khaerunnisa
[1,2,3,4,5,6,7]Computer Engineering, Indonesia University of Education
Dr. Setiabudhi Street No. 229, Bandung
e-mail: [1]ridhwannadif752@gmail.com, [2]aldiraspra@upi.edu, [3]muhammad.fajarjati@upi.edu, [4]kimberlyalfa@upi.edu, [5]rizkiafrzl02@upi.edu, [6]dedenpradeka@upi.edu, [7]zahrakhae@upi.edu

Abstract

Secure data communication is an essential requirement in the digital age, especially in the exchange of confidential information. One of the main problems is the potential for eavesdropping by unauthorised parties who can access open messages. To overcome this, this research aims to develop a website-based system that combines the Least Significant Bit (LSB) method steganography technique and the Caesar cipher method cryptography with the Dvorak keyboard layout to increase data security. The research method used is prototype method with the stages of concept formulation, prototype design, and continuous evaluation. The system was tested by inserting the word 'TEKKOM' into a digital image and measuring the quality using Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). The test results show an MSE value of 0.0001466049 and PSNR of 86.47 dB, which means that the visual quality of the image does not decrease significantly and the message is successfully inserted and extracted without errors. The Dvorak keyboard layout adds a layer of security through uncommon input patterns, making it difficult for outsiders to analyse. Thus, the system is able to provide a secure, efficient, and practical solution for web-based information hiding without the need for a database and is still user-friendly.

Keywords: *Dvorak, Steganography, Caesar Cipher, Least Significant Bit, Website*

1 INTRODUCTION

Advancements in information technology, particularly in the field of digital image processing, have become a central focus across various disciplines due to their potential in everyday life. An image is a representation of a two-dimensional object in the form of a collection of colored pixels, widely used in fields such as art, human vision, astronomy, and engineering [1]. As the need to securely identify and transmit data through digital images increases, challenges also arise in maintaining the confidentiality of information from unauthorized access.

Data security is a process that combines regulations and technology to protect information from damage, modification, and unauthorized distribution [2]. This effort includes both technical and procedural mechanisms to prevent hacking by unauthorized parties [3]. Data protection is also essential for maintaining the integrity and trust in the processes of data collection, processing, and storage [4]. Information security remains a critical aspect that must not be overlooked, even in the absence of incidents or breaches. The primary objective of information security is to safeguard technical aspects such as devices, networks, and all data processing facilities from both direct and indirect threats [5]. Without additional protection, data becomes vulnerable to unauthorized attacks, such as in the case of data theft [6]. One unique security approach is embedding data into media such as images, music, or videos so that the message remains unnoticed by attackers [7].

Steganography is a technique used to conceal information within digital media such as images, audio, or text [8]. This technique enables the insertion of confidential data without causing noticeable visual changes to the media [9]. Digital images are often chosen as the carrier medium due to their widespread use in online communication, making them less suspicious to attackers [10]. This process effectively protects information, as steganography conceals the presence of data from parties attempting to decipher the message—particularly if they do not possess the required key [11].

The Least Significant Bit (LSB) technique is one of the most popular steganography methods. This technique embeds a message into a digital image by replacing the least significant bit of the pixel value [12]. Since this bit has minimal impact on the pixel's color, the changes are imperceptible to the human eye. The method replaces the last bit of the steganographic media with bits from the message [13], making LSB highly effective for covert communication without significantly compromising image quality.

In addition, cryptography also plays a vital role in data security systems. The Caesar cipher is one of the classic cryptographic methods classified as a symmetric algorithm, where the encryption and decryption processes use the same key. This technique involves shifting characters based on a specific key value to obscure the message. The encryption process transforms the original message (plaintext) into an encrypted form (ciphertext), while the decryption process returns it to its original form using the same key. This flow is illustrated in Figure 1, which shows the stages of data transmission from the user: starting with inputting the plaintext message, then encrypting it into ciphertext using a key, and finally sending it to the receiver, who uses the same key to decrypt it back into a readable message.

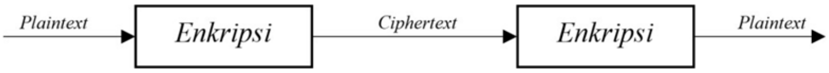


Figure 1. Cryptographic Mechanism

The combination of the Caesar cipher method with LSB results in a mechanism for embedding encrypted messages into digital images. This process produces a *stego image* that is visually identical to the original image. Figure 2 illustrates the steganographic encoding process in more detail. In the figure, the system receives input in the form of a digital image and an already encrypted secret message. The image and message are then processed by the LSB algorithm, in which each bit of the message is embedded into the least significant bit of the image's pixels. This process generates a new image (the stego image) that appears identical to the original image but contains hidden data.

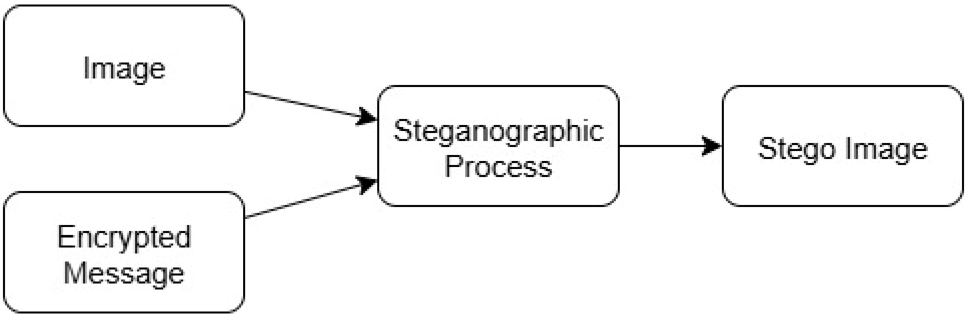


Figure 2. Steganographic Encoding Process

Conversely, Figure 3 illustrates the decoding or message extraction process from the stego image. The image containing the hidden message is uploaded to the system, then the LSB algorithm reads the least significant bits of each pixel to reconstruct the encrypted message. After that, the extracted message is decrypted using the Caesar cipher and the corresponding key. This process is the reverse of encoding, and the final result is the original message, which can be read again by the user. With high accuracy and no damage to the image, this process ensures the integrity of the information is preserved.

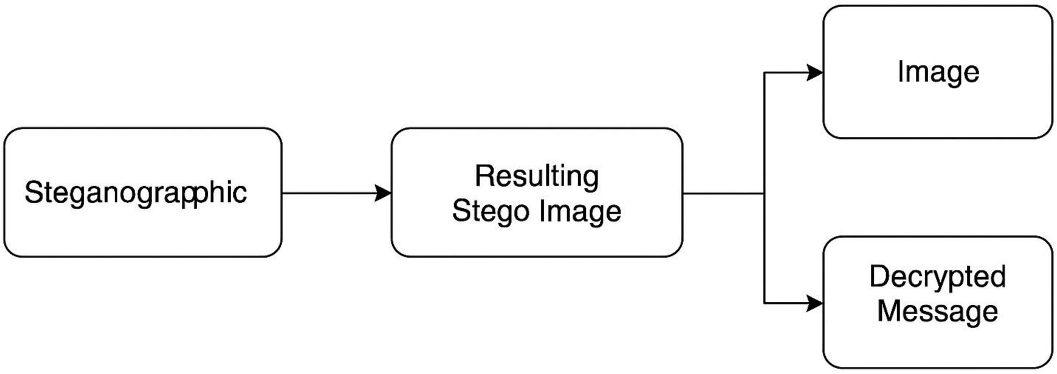


Figure 3. Steganographic Decoding Process

The addition of the Dvorak keyboard layout to this system enhances security by producing an unconventional input pattern that differs from QWERTY. This pattern makes it more difficult for external parties to analyze user input based on common typing patterns. Within the developed system, the use of Dvorak serves not only as an input tool but also as a source of input pattern variation in the encryption process, making it harder to predict. As a contribution to digital media-based data security studies, this system integrates all three elements into an interactive website platform without using a database for storage. This approach not only improves efficiency and ease of use, but also introduces a *novelty* by applying an alternative keyboard layout as an added layer of security in the message hiding process. Thus, the system provides an innovative solution for securing digital communications that is both user-friendly and robust in terms of security.

2 LITERATURE REVIEW

The use of web technologies in securing data through steganographic techniques has become a focus in various studies over recent years. One study developed a dynamic website creation system equipped with a feature to embed messages into images using the Least Significant Bit (LSB) method [14]. This study emphasized efficiency and speed, where the process of generating a web page with a hidden message could be completed in under five minutes. The system was designed for confidential communication with a user-friendly interface, making it ideal for military contexts or organizations with high confidentiality needs [15].

Another study demonstrated an innovative approach by integrating a Generative Adversarial Network (GAN) algorithm to strengthen steganography techniques against steganalysis attacks [16]. GAN works by generating fake images that closely resemble original images but have hidden messages embedded in them. This technique utilizes the training of two neural networks—generator and discriminator—to increase image realism and reduce the chances of message detection. Testing showed that GAN-generated images have high resistance to detection attempts, even when the image metadata is altered or recompressed, making it one of the most powerful approaches for protecting visual messages.

From the cryptography perspective, several studies have adopted simple encryption methods such as the Caesar cipher to hide text in digital media. One study embedded messages into video files instead of images, showing that steganographic techniques can be applied across different media types [17]. While this method successfully concealed data, it also caused a significant increase in video file size. The study further explained that character-shift-based encryption methods are easy to implement but vulnerable to attacks if used without additional protection layers. In several cases, such techniques are only suitable for low-risk environments and are not intended for large-scale production.

Although previous research has shown significant progress in combining web technology, steganography, and cryptography, most of the approaches remain fragmented and do not integrate multiple layers of security simultaneously. The majority of studies focus either solely on message embedding using LSB or only on basic cryptographic techniques. None of the previous approaches explicitly combine LSB, Caesar cipher encryption, and the Dvorak keyboard layout into a complete web-based system.

The novelty of this system lies in the integration of three approaches: (1) message hiding using LSB, (2) message content protection through Caesar cipher encryption, and (3) input pattern randomization based on the Dvorak keyboard layout. The use of Dvorak in particular is a unique approach not previously applied in earlier studies. This layout generates typing patterns that differ from QWERTY, complicating input pattern detection in forensic analysis or behavior-based attacks. Furthermore, the absence of a database for storage makes the system lighter and more secure from database exploitation.

With this multi-layered approach, the system developed in this study not only integrates existing techniques but also expands the security dimensions of confidential communication in a more complex and resilient manner. Compared to previous studies, this system offers a unique combination of methods, message protection effectiveness, and user accessibility through a web platform.

3 RESEARCH METHOD

This study adopts the Prototype Method approach, a software engineering methodology based on iterative development, which allows system creation through a cyclical process: building an initial prototype, testing it directly, and then conducting evaluations based on the obtained results. According to the method proposed in [18], the prototype approach is ideal when system requirements are not yet fully defined at the outset, as it enables gradual exploration alongside users until a stable final system is achieved. It emphasizes direct interaction between users and the system, as well as continuous refinement toward the final version.

In this research, the prototype method was chosen because the design of a steganography system integrated with cryptography and the Dvorak keyboard layout is relatively unexplored. Therefore, exploration and user feedback are crucial for refining both the interface and the algorithmic functionality.

3.1. Research Stages Diagram

The stages of the prototype method in this study consist of: concept formulation, prototype design, and system evaluation. These stages are visualized in Figure 4, which illustrates the logical flow and iterative relationships between each development phase.

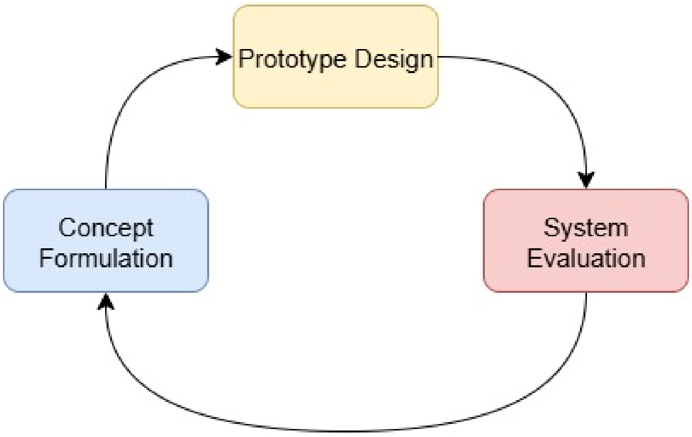


Figure 4. Flowchart of the Research Stages Using the Prototype Method

In Figure 4, it is shown that each stage is not linear in one direction but supports iteration. For example, after the prototype is tested, the process returns to the design phase to improve weaknesses based on user evaluation. This approach ensures that the system continues to evolve until it meets the desired criteria of functionality, security, and usability.

3.2. Concept Formulation

The concept formulation began with identifying the system requirements to be developed, which include: embedding secret messages into digital images using the LSB method, protecting messages with Caesar cipher encryption, and enhancing security by utilizing the Dvorak keyboard layout as the basis for character shifting. The system is also designed to operate without a database, using only local files to avoid potential data leakage.

These requirements led to the idea that the system must be capable of processing images directly, embedding messages without degrading visual quality, and being accessible to general users through a lightweight and responsive web interface. All these needs serve as the foundation for determining the structure of the prototype to be developed.

3.3. Prototype Design

The prototype was designed to implement the integration of the Caesar cipher and LSB algorithms in a web environment using Python Streamlit. The user interface is developed with a minimalist yet functional design, allowing users to upload images, enter secret messages, set encryption keys, and display both the resulting stego image and the extracted message.

Figure 5 illustrates a basic simulation of character shifting in the Caesar cipher algorithm, where each letter in the original message is shifted according to the encryption key value. Although this illustration refers to the standard alphabetical order, the same principle is applied in this system with adjustments based on the letter sequence of the Dvorak keyboard layout. In the actual implementation, the Dvorak sequence is used to replace the QWERTY alphabet in order to increase the complexity of the encryption pattern.

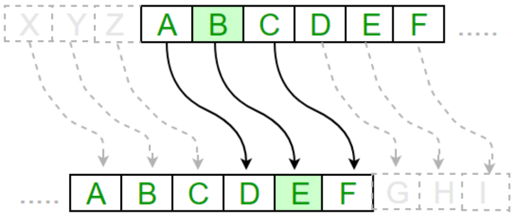


Figure 5. Simulation of Character Shifting in Caesar Cipher [19]

Next, Figure 6 shows the layout of the Dvorak keyboard, which is used as the basis for the algorithm. This layout explains why the encryption output pattern differs from that of the standard Caesar cipher algorithm.

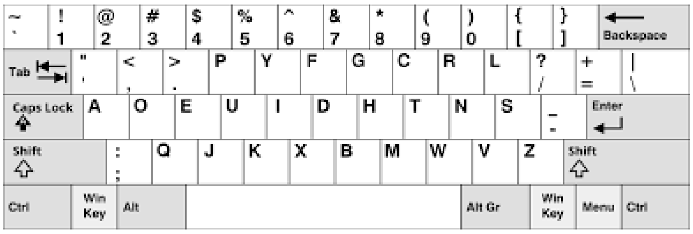

~ ` ! 1 @ 2 # 3 $ 4 % 5 ^ 6 & 7 * 8 (9) 0 { [}] Backspace
Tab " ' < , > . P Y F G C R L ? / + = | \
Caps Lock A O E U I D H T N S _ - Enter
Shift : ; Q J K X B M W V Z Shift
Ctrl Win Key Alt Alt Gr Win Key Menu Ctrl

Figure 6. Dvorak Keyboard Layout Used in the Encryption System

For the message embedding process into the image, the Least Significant Bit (LSB) method is used. This process takes each bit of the message and replaces the 0th bit of each pixel in an 8-bit grayscale image. This technique leverages the fact that changes in the least significant bit are imperceptible to the human eye. Each message bit is directly embedded into the binary value of the pixel's grayscale color. For example, if a pixel value is 10101010 and the message bit is 1, then the pixel will be changed to 10101011 [20].

The implementation of this process is carried out in a structured manner in the system, following the sequence:

image input → message encryption → binary conversion → embedding into pixels → stego output.

To visualize how the system works, two sequence diagrams were created to illustrate the interactions between the user and the system during the encryption and decryption processes. Figure 7 shows the encryption process. In this figure, the user first uploads an image, types the message, and enters the encryption key. The system then encrypts the message using a Dvorak-based Caesar cipher, and embeds the encrypted result into the image using the LSB algorithm. The final result is displayed to the user as a stego image.

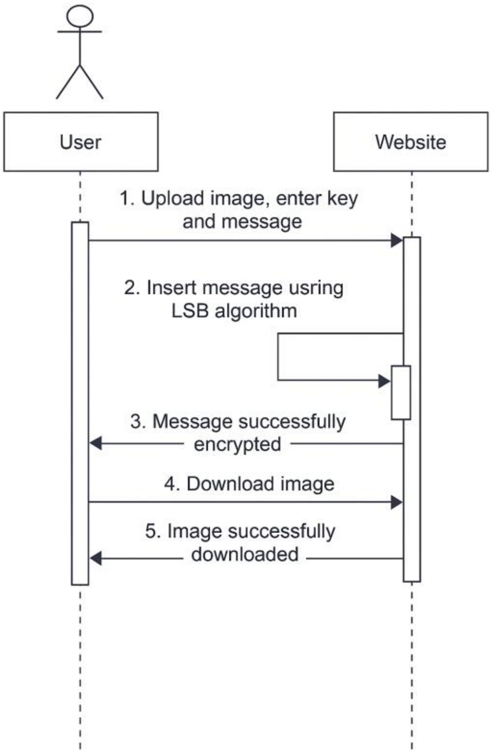

User
Website
1. Upload image, enter key and message
2. Insert message usring LSB algorithm
3. Message successfully encrypted
4. Download image
5. Image successfully downloaded

Figure 7. Sequence Diagram of the Encryption Process

Conversely, Figure 8 illustrates the decryption process. The stego image is uploaded, then the system extracts the hidden bits, reconstructs the encrypted message, and decrypts it using the provided key. This process results in the original message, which is then displayed to the user.

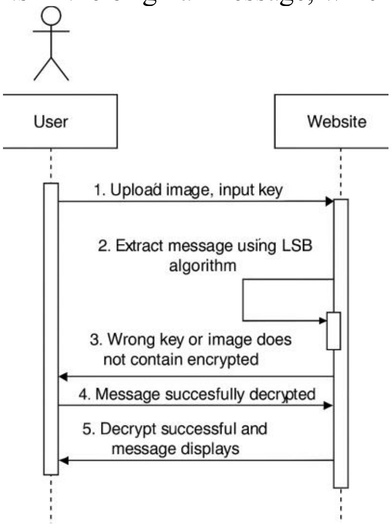

User
Website
1. Upload image, input key
2. Extract message using LSB algorithm
3. Wrong key or image does not contain encrypted
4. Message succesfully decrypted
5. Decrypt successful and message displays

Figure 8. Sequence Diagram of the Decryption Process

3.4. Prototype Evaluation

The evaluation was conducted through functional testing, image quality testing, and user perception testing.

1. **Functionality Testing**

All features were tested, starting from image upload, message input, key input, encryption, embedding, stego image saving, extraction, and decryption. Each function was tested using various input scenarios, including short and long messages, as well as different types of grayscale images.

2. **Image Quality Testing**

This test was conducted to determine the extent to which data embedding affects the visual quality of the image. Two quantitative metrics were used:

a. MSE (Mean Squared Error), calculated using the formula:

$$MSE = \frac{1}{mn}\sum_{i=1}^{m} \sum_{j=1}^{n} [I(i,j) - K(i,j)]^2 \quad (1)$$

Explanation of variables Equation (1):

m = number of rows (image height)

n = number of columns (image width)

$I(i,j)$ = pixel intensity at position (i,j) in the original image

$K(i,j)$ = pixel intensity at position (i,j) in the stego image

This formula represents the mathematical equation for calculating the average difference in pixel values between the original image and the stego image. A small MSE value indicates that the modification is nearly imperceptible.

b. PSNR (Peak Signal-to-Noise Ratio) is calculated using the formula:

$$PSNR = 10 . log_{10}(\frac{MAX^2}{MSE}) \quad (2)$$

Explanation Equation (2):

MAX = the maximum possible pixel intensity value (usually 255 for 8-bit images)

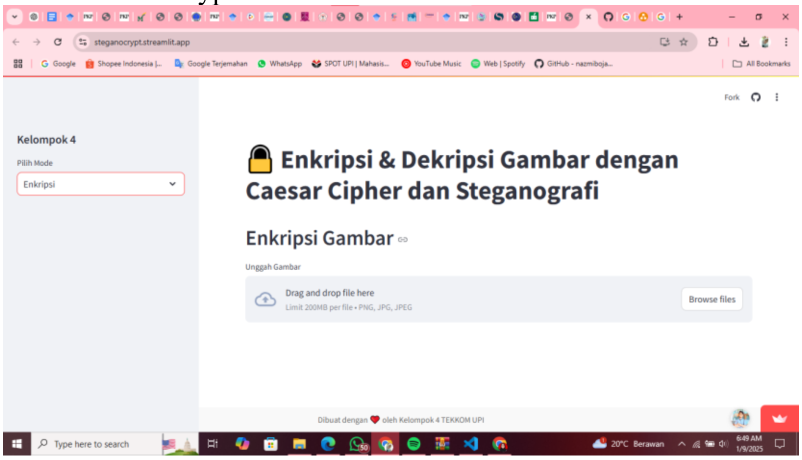
This formula represents the calculation of PSNR, which expresses image quality on a logarithmic scale. A PSNR value above 50 dB is considered excellent in terms of human visual perception.

4 RESULTS AND DISCUSSION

This research successfully produced a website-based system that integrates the Least Significant Bit (LSB) algorithm for steganography and a Dvorak keyboard layout-based Caesar cipher for message encryption. The system is designed to allow users to upload digital images, embed encrypted secret messages, and later extract those messages—all through a user-friendly interface. The use of the Dvorak layout serves not only as an alternative input method but also as a mechanism to enhance security both cognitively and technically.

4.1. System Design and Website Functionality

The developed website interface was designed to be user-friendly, enabling general users to operate the system without needing technical knowledge in steganography or cryptography. At the initial stage, users are directed to upload an image that will be used as the medium for message embedding. This process is shown in Figure 9, which displays the webpage when a user uploads an image and accesses the encryption menu.



Kelompok 4

Pilih Mode: Enkripsi

Enkripsi & Dekripsi Gambar dengan Caesar Cipher dan Steganografi

Enkripsi Gambar

Unggah Gambar

Drag and drop file here

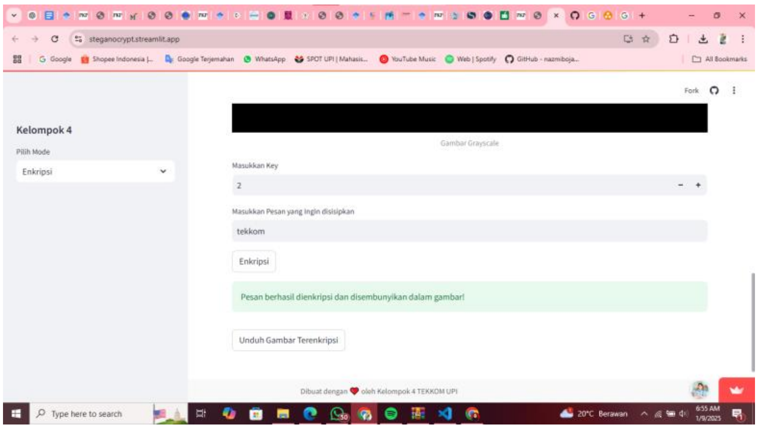
Limit 200MB per file • PNG, JPG, JPEG

Dibuat dengan ❤ oleh Kelompok 4 TEKKOM UPI

Figure 9. Encryption Process Interface

After the image is successfully uploaded, the user is prompted to enter the secret message to be embedded along with a numerical encryption key. The system then encrypts the message using the Caesar cipher algorithm based on the Dvorak letter sequence, and embeds it into the image using the LSB method.

This input stage is visualized in Figure 10, which displays the input form for the message and the encryption key.



Kelompok 4

Pilih Mode: Enkripsi

Gambar Grayscale

Masukkan Key: 2

Masukkan Pesan yang ingin disisipkan: tekkom

Enkripsi

Pesan berhasil dienkripsi dan disembunyikan dalam gambar!

Unduh Gambar Terenkripsi

Dibuat dengan ❤ oleh Kelompok 4 TEKKOM UPI

Figure 10. Input Interface for Key and Secret Message

The system also provides a feature to download the encrypted image. Visually, the resulting image shows no noticeable difference compared to the original image; however, it contains a hidden secret message embedded digitally. Figure 11 shows an example of the original image before the embedding process, an displays the encrypdted image, which appears visually identical to the original.

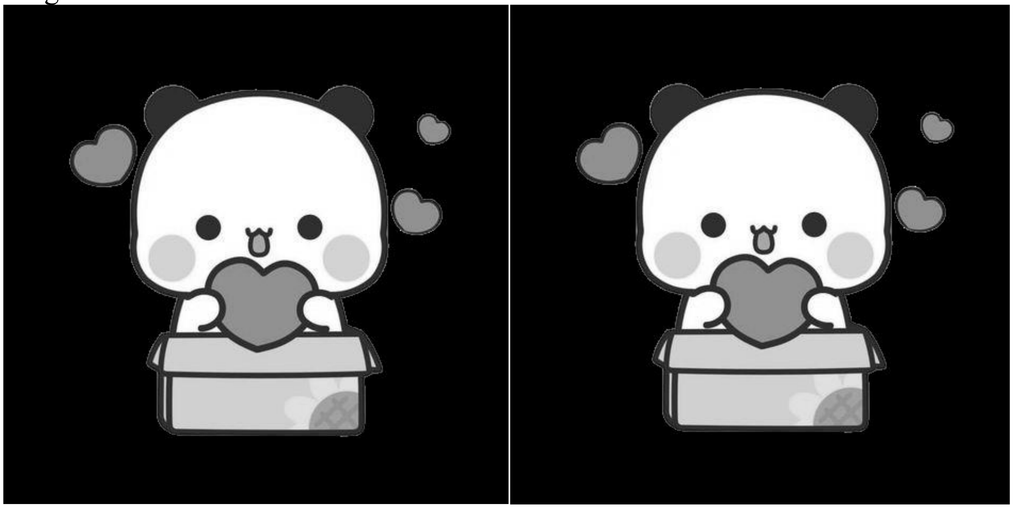
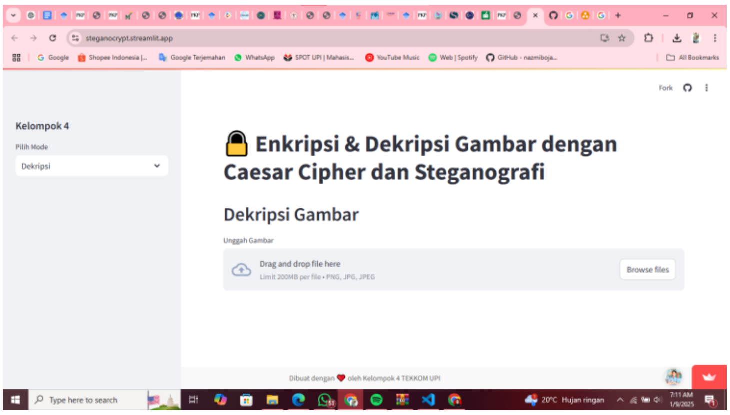


Figure 11. (a) Uploaded Image, (b) Encrypted Image

Both images are visually almost indistinguishable, demonstrating the effectiveness of the LSB method in embedding information without significantly degrading visual quality.

4.2. Decryption Process and Key Validation

The decryption process in the system follows a flow similar to the encryption process. The user uploads the stego image, enters the encryption key, and the system then extracts and decrypts the hidden message. Figure 12 shows the interface when the user accesses the decryption page and uploads the image.



Kelompok 4

Pilih Mode

Dekripsi

Enkripsi & Dekripsi Gambar dengan Caesar Cipher dan Steganografi

Dekripsi Gambar

Unggah Gambar

Drag and drop file here

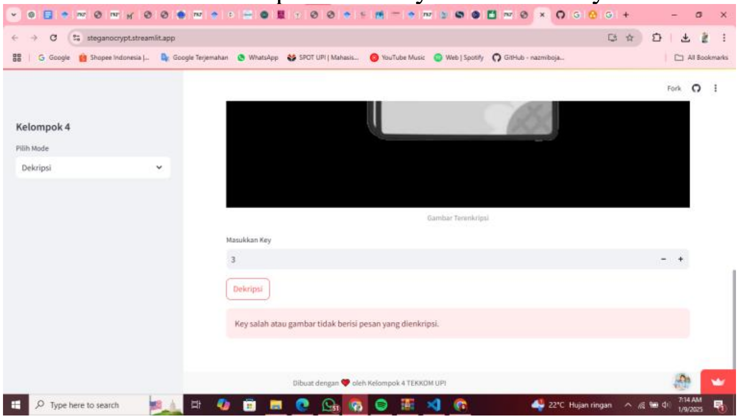
Limit 200MB per file • PNG, JPG, JPEG

Browse files

Dibuat dengan ❤ oleh Kelompok 4 TEKKOM UPI

Figure 12. Display of the decryption process

After the user enters the key, the system performs validation. If the entered key is incorrect, the system will automatically reject the decryption process and display an error message, as shown in Figure 13. This feature is essential as part of the system's security validation.



Kelompok 4

Pilih Mode

Dekripsi

Gambar Terenkripsi

Masukkan Key

3

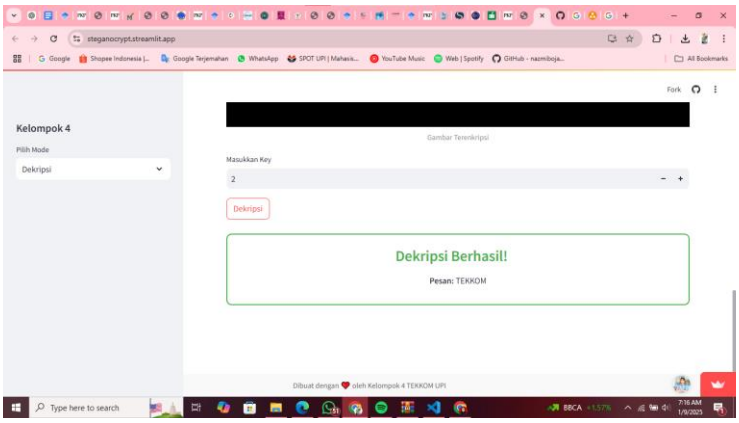
Dekripsi

Key salah atau gambar tidak berisi pesan yang dienkripsi.

Dibuat dengan ❤ oleh Kelompok 4 TEKKOM UPI

Figure 13. Warning Display When the Entered Key Is Incorrect

Conversely, if the correct key is entered, the system successfully extracts the message and displays it fully to the user, as shown in Figure 14. This success demonstrates that the encryption and decryption processes can be performed symmetrically and reliably, as long as the key used remains consistent.



Kelompok 4

Pilih Mode

Dekripsi

Gambar Terenkripsi

Masukkan Key

2

Dekripsi

Dekripsi Berhasil!

Pesan: TEKKOM

Dibuat dengan ❤ oleh Kelompok 4 TEKKOM UPI

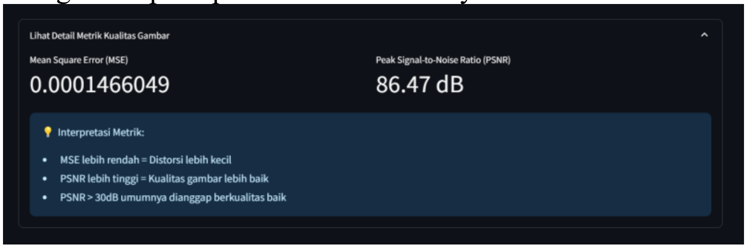
Figure 14. Display When the Entered Key Is Correct

4.3. Evaluation of Encrypted Image Quality

To ensure that the embedding process does not visually degrade image quality, an evaluation was carried out using two metrics: Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR).

Figure 15 presents the calculation results of these two metrics based on tests conducted on images with embedded messages. The resulting MSE value is very low, at 0.0001466049, while the PSNR value reaches 86.47 dB.

A PSNR value above 50 dB indicates that the image quality is very high, and any difference from the original image is imperceptible to the human eye.



Lihat Detail Metrik Kualitas Gambar

Mean Square Error (MSE)

0.0001466049

Peak Signal-to-Noise Ratio (PSNR)

86.47 dB

💡 Interpretasi Metrik:

- MSE lebih rendah = Distorsi lebih kecil
- PSNR lebih tinggi = Kualitas gambar lebih baik
- PSNR > 30dB umumnya dianggap berkualitas baik

Figure 15. Calculation of MSE and PSNR

These results indicate that the LSB method is capable of preserving the visual quality of the image extremely well, while effectively hiding the embedded data.

4.4. Uniqueness and Comparison with Previous Studies

Compared to previous studies that only used conventional LSB methods without additional elements in encryption or interface design, the system developed in this study offers uniqueness in two key aspects.

First, the system utilizes the Dvorak keyboard layout as the basis for the Caesar cipher algorithm, creating an encryption pattern that is not easily predictable based on the standard alphabetical order. This adds an extra layer of protection against input pattern analysis attacks or simple brute-force attempts.

Second, the system was developed as an interactive web-based platform without a database, meaning that all processes are performed locally without storing any sensitive data. This sets the study apart from most previous research, which generally relies on desktop-based systems or standalone applications. The web-based approach offers greater accessibility and efficiency for users.

By combining a user-friendly interface, Caesar cipher encryption based on the Dvorak layout, and a low-distortion LSB embedding technique, this study successfully designed a digital message security system that is innovative, secure, and still accessible to non-technical users.

5 CONCLUSION

Based on system testing results and the technical analysis conducted, it can be concluded that the integration of the Least Significant Bit (LSB) technique for steganography and the Caesar cipher algorithm based on the Dvorak keyboard layout within a web-based platform has proven effective in hiding secret messages within digital images. The system is capable of performing encryption and decryption processes with high accuracy, without causing significant degradation to the visual quality of the image.

Quantitative test results indicate that image quality remains well-preserved after the embedding process. The obtained MSE value of 0.0001466049 signifies that the pixel changes caused by embedding are minimal. Meanwhile, the PSNR value of 86.47 dB confirms that the quality of the encrypted image is excellent and nearly indistinguishable from the original image. The main innovation in this study lies in the use of the Dvorak keyboard layout as the basis for modifying the Caesar cipher algorithm, which introduces a non-conventional encryption pattern. This makes it more difficult for third parties—accustomed to the QWERTY layout—to analyze or break the code. This approach provides an additional layer of protection both psychologically and algorithmically against input pattern-based hacking attempts.

In addition to security, the system also prioritizes practicality. Users do not need in-depth technical knowledge to encrypt or decrypt messages. The web-based interface allows the process to be carried out directly in the browser, without requiring installation or additional configuration—making the system efficient and easily accessible to users from various backgrounds.

Overall, the system succeeds not only in terms of technical performance and security, but also in delivering a user experience that is simple yet robust. The combination of cryptographic algorithms, steganographic methods, and a web-based interface makes this system an innovative and practical solution for securing digital messages.

REFERENCES

[1] A. R. Putri, "Image processing using a webcam on a moving vehicle on the highway," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*., vol. 1, no. 1, pp. 1–6, 2016. [Online]. Available: https://doi.org/10.29100/jipi.v1i01.18.

[2] S. Siaulhak, S. Kasma, and Suparman, "File delivery system using digital image processing steganography based on laboratory matrix," *Bandwidth: J. Inform. dan Rekayasa Komputer*., vol. 1, no. 2, pp. 75–81, 2023. [Online]. Doi: https://doi.org/10.53769/bandwidth.v1i2.522

[3] A. S. Joel, F. Abdussalaam, and Y. Yunengsih, "Medical record management based on information technology in handling patient data confidentiality and security using cryptographic methods," *J. Indones. Manajemen Inform. dan Komunikasi.*, vol. 4, no. 3, pp. 837–848, 2023. Doi: https://doi.org/10.35870/jimik.v4i3.287

[4] S. D. Sari, "Privacy and data security in official statistics: challenges and solutions in protecting individual data," *Madani: J. Ilmiah Multidisiplin*., vol. 1, no. 11, pp. 700–703, 2023. [Online]. Available: https://doi.org/10.5281/zenodo.10371661

[5] A. L. Maryanto, M. N. Al Azam, and A. Nugroho, "Evaluation of information security management in technology-based startup companies using the KAMI Index.," *J. Simantec*, vol. 11, no. 1, pp. 1–12, 2022. Doi: https://doi.org/10.21107/simantec.v11i1.14099

[6] B. Anwar, N. B. Nugroho, J. Prayudha, and Azanuddin, "Implementation of the RSA algorithm for data security in savings and loan systems," *Sains dan Komputer (SAINTIKOM)*., vol. 18, no. 1, pp. 30–34, 2019. [Online]. Available: https://doi.org/10.53513/jis.v18i1.100.

[7] S. Magdy, S. Youssef, K. M. Fathalla, and S. ElShehaby, "DeepSteg: Integerating new paradigms of cascaded deep video steganography for securing digital data," Alexandria Eng. J., vol. 116, pp. 483-501, 2025, doi: https://doi.org/10.1016/j.aej.2024.12.034. https://doi.org/10.1016/j.aej.2024.12.034

[8] L. F. Adhimah, I. Nurhafiyah, A. A. Muntahar, F. Kristiaji, and D. Mustofa, "Implementation of a web-based steganography application using the LSB and BPCS algorithms," *KOMPUTA: J. Ilmiah Komputer dan Informatika*., vol. 12, no. 2, pp. 100–108, 2023. Doi: https://doi.org/10.34010/komputa.v12i2.10319

[9] D. Pradeka, A. Adiwilaga, D. A. R. Agustini, A. Suheryadi, and R. Nuriman, "Design and build an assessment platform by inserting Moodle-based cryptographic methods," *TEKNOSI.*, vol. 9, no. 3, pp. 264–270, Jan. 2024. Doi: https://doi.org/10.25077/TEKNOSI.v9i3.2023.264-270

[10] S. Solera-Cotanilla, M. Álvarez-Campana, C. Sánchez-Zas, and M. Vega-Barbas, "Proposal for a security and privacy enhancement system for private smart environments," Internet of Things, vol. 31, p. 101585, 2025, doi: https://doi.org/10.1016/j.iot.2025.101585

[11] I. U. W. Mulyono, Y. Kusumawati, and N. K. Ningrum, "Visual analysis of images resulting from the combination of steganography and cryptography based on the Least Significant Bit in cipher," *J. Masyarakat Informatika*., vol. 14, no. 1, pp. 16–28, 2023. [Online]. Doi: https://doi.org/10.14710/jmasif.14.1.51484

[12] A. Supardi, A. A. Alkodri, and B. Isnanto, "Steganography technique for hiding secret text messages in digital images using the Least Significant Bit method," *J. Sisfotek Global*., vol. 11, no. 1, pp. 70–74, 2021. [Online]. Doi: https://doi.org/10.38101/sisfotek.v11i1.351

[13] A. R. Mido and E. I. H. Ujianto, "Analysis of the impact of images on the combination of RSA cryptography and LSB steganography," *JTIIK*., vol. 9, no. 2, pp. 279–286, 2022. Doi: http://dx.doi.org/10.25126/jtiik.201743299.

[14] A. M. Ramadhani and T. Hassanuddin, “Modification of Least Significant Bits in images for data hiding in steganography,” *IJODAS.*, vol. 2, no. 2, pp. 91–102, 2021. Doi: https://doi.org/10.25126/jtiik.201743299

[15] A. Alenizi, M. S. Mohammadi, A. A. Al-Hajji, and A. S. Ansari, “A Review of Image Steganography Based on Multiple Hashing Algorithm,” Comput. Mater. Contin., vol. 80, no. 2, pp. 2463-2494, 2024, Doi: https://doi.org/10.32604/cmc.2024.051826.

[16] Y. D. Cahyono, A. Suryanta, and I. P. Wardani, “Implementation of AJAX for a Dynamic Interface in a Website Builder with Hidden Message Embedding Capability,” *JATI.*, vol. 9, no. 1, pp. 9–14, 2023. Doi: https://doi.org/10.36040/jati.v9i1.12129

[17] G. M. Fahmi, K. N. Isanini, and D. Suhartono, “Image Steganography Implementation Using the Generative Adversarial Network (GAN) Algorithm,” *SINTECH J.*, vol. 6, no. 1, pp. 47–57, 2023. Doi: https://doi.org/10.31598/sintechjournal.v6i1.1258

[18] G. Zhao, Y. Huang, B. Jia, J. Ji, and P. Cheng, “A prototype low-pressure assisted microwave plasma ionization mass spectrometry for on-line monitoring of organic and inorganic hazardous compounds simultaneously: Design and feasibility validation,” Anal. Chim. Acta, vol. 1354, p. 344004, 2025, doi: https://doi.org/10.1016/j.aca.2025.344004

[19] GeeksforGeeks, “Caesar cipher in cryptography,” *GeeksforGeeks*., [Online]. Available: https://www.geeksforgeeks.org/caesar-cipher-in-cryptography/. [Accessed: 9-Jan-2025].

[20] B. Purnama and A. H. H. Rohayani, “A New Modified Caesar Cipher Cryptography Method with LegibleCiphertext From a Message to Be Encrypted,” Procedia Comput. Sci., vol. 59, pp. 195-204, 2015, Doi: https://doi.org/10.1016/j.procs.2015.07.552