International Journal of Scientific Research in Science and Technology
Available online at : www.ijsrst.com
OPEN ACCESS
Print ISSN: 2395-6011 | Online ISSN: 2395-602X
doi : https://doi.org/10.32628/IJSRST251245

Digital Data Security : Integration of RSA and Pseudo-Random Prime Number Generator in Steganography Engineering

Andysah Putera Utama Siahaan*, Ade May Luky Harefa, Indra Nasution, Robin Antoni, Yohanes France Limbong, Sella Monica Br. Tarigan

*Information Technology, Pembangunan Panca Budi University, Medan, Jl. General Gatot Subroto Km. 4.5, Medan, Indonesia

ARTICLEINFO

Article History:

Accepted : 26 June 2025
Published: 01 July 2025

Publication Issue :
Volume 12, Issue 4
July-August-2025

Page Number :
29-34

ABSTRACT

The increasing reliance on digital data exchange has raised critical concerns regarding information security and confidentiality. This research proposes a secure data hiding technique by integrating the RSA algorithm and a Pseudo-Random Prime Number Generator (PRPNG) into steganography. The objective is to enhance both the encryption efficiency and the robustness of hidden data. The method involves converting secret messages into numeric form limited to digits 0-9 to optimize the Least Significant Bit (LSB) substitution process within digital images. The encryption utilizes asymmetric RSA keys generated from dynamically selected pseudo-random prime numbers, which adds a layer of complexity and security. The experiment compared conventional and optimized steganographic approaches by measuring encryption-decryption time and the imperceptibility of the modified images. Results indicate that the optimized steganography method significantly reduces encryption time while embedding a greater amount of data without affecting visual quality. Moreover, the decryption process, while slightly slower, benefits from enhanced security due to the requirement of specific private keys, correct prime pair identification, and precise padding. This integration proves to be a viable and secure approach for embedding sensitive data in digital images, contributing to improved digital data protection in various applications, especially those requiring confidentiality, such as secure communications and digital watermarking.

Keywords: Steganography, RSA Algorithm, Pseudo-Random Prime Number Generator, Data Security, Digital Image Encryption.

I. INTRODUCTION

The rapid development of the digital era has provided extraordinary convenience in communication activities and access to information [1]. Now, humans can exchange data instantly, without time and location restrictions. These digital innovations clearly bring great benefits to life, ranging from the education, health, business, to government sectors (Editor, 2024). However, behind this convenience, there is a serious threat to data security. The phenomenon of hacking, wiretapping, and theft of personal and institutional information is becoming an increasingly prevalent global problem [3]. Therefore, a solution is needed that is able to ensure the safe transmission and receipt of data with high security standards, in order to protect the integrity and confidentiality of information.

One of the technologies that has long been used to maintain data security is cryptography. Cryptography is a data encoding technique so that only parties with a certain key can read the content of the message [4]. According to historical records, cryptography has been used since the time of Ancient Greece, around 400 BC. One of the most well-known modern cryptographic methods is the RSA (Rivest-Shamir-Adleman) algorithm [5]. RSA is an asymmetric key algorithm that uses two different keys, namely a public key for encryption and a private key for decryption [6]. This characteristic makes RSA one of the most powerful algorithms in keeping data confidential because only recipients with the corresponding private key can open encrypted messages [7].

In addition to cryptography, there are also steganography techniques that have different approaches to maintaining data security [8]. Steganography is a technique of hiding messages in other objects, such as digital, audio, or video images, in such a way that the existence of the message is not detected by ordinary observers [9]. The main purpose of steganography is to disguise the message so that it is not suspicious, so that even if the file is hijacked, the content of the message is not immediately known [10]. Digital imagery is one of the most commonly used media in steganography, primarily through the Least Significant Bit (LSB) method, which inserts bits of messages into image pixels [11]. However, this method still has loopholes if the data insertion pattern is not random enough or if there is a steganalysis attack [12]. To overcome these shortcomings, an approach that combines cryptography and steganography can be used, namely by first encrypting messages using the RSA algorithm, then inserting them into digital images using steganography techniques. However, in order to make data insertion more random and less predictable, a powerful randomization system is needed, one of which is the Pseudo-Random Number Generator (PRNG). PRNG is a pseudo-random number generation algorithm based on a specific initial value or seed [13]. Despite its pseudo-nature, PRNG can generate a series of numbers that appear random and is very useful for determining the position of data insertion in an image. If PRNG is further developed to generate random prime numbers, it could be the foundation in the creation of more varied and robust RSA keys [14].

The integration of prime number-based RSA and PRNG algorithms into steganography systems offers a more solid digital data security solution. Messages that have been encrypted using RSA will be difficult to decrypt without the right private key, and the insertion of messages into the image is done with the help of PRNG so that their positions are scattered randomly and difficult to trace. With this combination, data security is not only guaranteed through encoding the content of the message, but also through the disguise of the existence of the message itself. Therefore, research on the incorporation of RSA and Pseudo-Random Prime Number Generator in steganography techniques is important to be further developed as an answer to the increasingly urgent need for digital data protection in this modern era [15].

II. METHODS

To optimize the steganography process using the LSB method, RSA algorithms and Pseudo-Random Prime-Number Generator will be used. Here is the implementation scheme.

A. Encryption Scheme

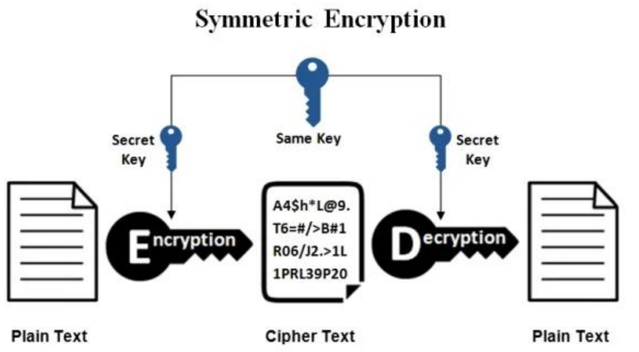


Figure 1: Encryption Scheme

1. Perform key generation by taking two random prime numbers, e.g. p and q, using a pre-optimized pseudo-random prime number.
2. Calculate the value of n with $n = p \times q$ whose value does not need to be kept secret and the value of m with $m = (p - 1)(q - 1)$ whose value is kept secret.
3. Selects a random number e that meets the condition PBB(e, m) = 1 as the public key.
4. Request the address of the image and the message to be inserted into the image.
5. Converts message input into ASCII values for each of its characters.
6. Perform the calculation for each ci-ciphertext value for the plaintext block pi converted to ASCII with the equation ci = pe (mod n) with e is the public key.
7. Unify the calculations of the ciphertext code with padding known only to the message creator. A message will be created with a much longer and more complex cryptographic result.
8. Implement a steganography scheme by inserting a message on the last bit of the image using the LSB method.
9. Save the image and the implementation of steganography is complete. Don't forget to store the public key and the private key that has been raised before.

B. Decryption Scheme

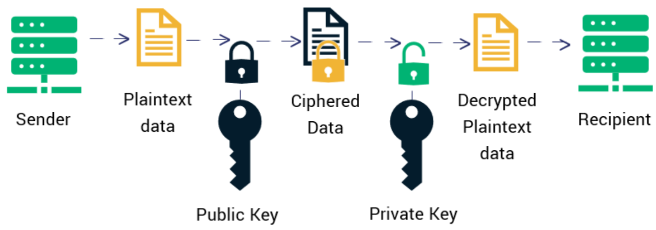


Figure 2: Decryption Scheme

1. Request the address of the image to be decrypted.
2. Decrypt from steganography to obtain a secret message. However, remember, the message you get is not the actual message because it is the result of the RSA algorithm implementation.
3. Repartition according to the partition at the time of encryption. A person who does not know encryption padding will have difficulty at this stage because the decryption result will be different if the encoded padding is different.
4. Asking for input from two values, namely the value of n and the public key e.
5. Performs a decryption key calculation d that meets the equation $ed \equiv 1 \pmod m$ with a value of m that is known only to the creator of the message and the person to whom the message is intended.
6. Performing a decryption calculation for each of the values of the plaintext block pi with the ciphertext block ci through the equation pi = cd (mod n) with e is the decryption key that was sought in the previous stage.
7. A number of characters will be obtained to unite the results of the plaintext code calculation, so the message conveyed through steganography has been successfully decrypted.

The process of inserting messages into steganography will make many changes to an image, but with much faster and less visible changes. This can happen because the character change process is carried out only in the range of 0 – 9 numbers and does not contain characters that make the LSB implementation need to change 2 – 3 bits of the image.

The decryption process will be even more time-consuming because finding a prime number pair that meets these conditions is not easy if the intended number range is above 105. This makes the decryption process very time-consuming and of course the results will not be correct if the keys and padding given and declared are not as they should be. To make matters worse, the prime numbers generated through PRNG require a very long period of time to be repeatable and different prime pairs will always appear for each input, so it will be difficult to declare a static key every time a message is inserted into an image.

III.RESULTS AND DISCUSSION

The following is a test mechanism carried out to see the effectiveness of steganography optimization results. An input image will be used as follows.

Figure 3: Image to be used

This original image has dimensions of 770 × 516 pixels and is stored as a PulauPanjangNiasUtara.jpg. In this part, tests will be carried out for two steganography methods, the LSB method and optimization. Here is a plaintext with 395 characters to be included in the image.

One of the destinations that must be visited while in North Nias is Panjang Island. This island offers extraordinary natural charm, both from its land side and under the sea. The natural beauty that is still natural makes Panjang Island one of the hidden marine tourism prima donnas in this region.

Figure 4: Plaintext

A. Ordinary steganography

This usual steganography was done without the use of the RSA algorithm and the Pseudo-Random Number Generator mechanism, here are the results:

Figure 5: Images of ordinary steganography results

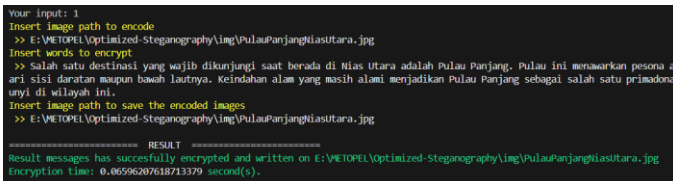
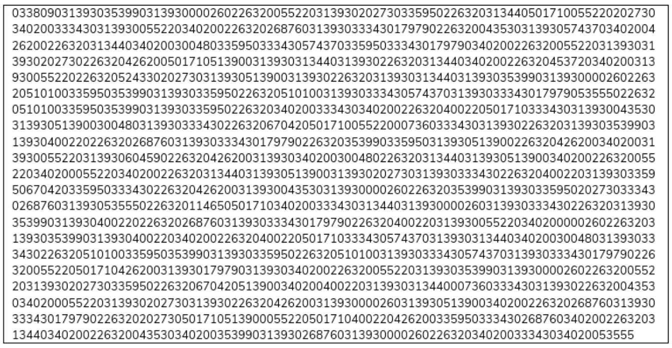


Figure 6: Encryption process 0.06596207618713379 second(s).

B. Optimized steganography

In this optimized steganography, all algorithms have been used, the following are the results.



0338090313930353990313930000260226320055220313930202730335950226320313440501710055220202730
3402003334303139300552203402002263202687603139303334301797902263200435303139305743703402004
2620022632031344034020030048033595033343057437033595033343017979034020022632005522031393031
3930202730226320426200501710513900313930313440313930226320313440340200226320453720340200313
9300552202263205243302027303139305139003139302263203139303134403139303539903139300002602263
2051010033595035399031393033595022632051010031393033343057437031393033343017979053555022632
0510100335950353990313930335950226320340200333430340200226320400220501710333430313930043530
3139305139003004803139303334302263206704205017100552200073603334303139302263203139303539903
1393040022022632026876031393033343017979022632035399033595031393051390022632042620034020031
3930055220313930604590226320426200313930340200300480226320313440313930513900340200226320055
2203402000552203402002263203134403139305139003139302027303139303334302263204002203139303359
5067042033595033343022632042620031393004353031393000026022632035399031393033595020273033343
0268760313930535550226320114650501710340200333430313440313930000260313930333430226320313930
3539903139304002202263202687603139303334301797902263204002203139300552203402000002602263203
1393035399031393040022034020022632040022050171033343057437031393031344034020030048031393033
3430226320510100335950353990313930335950226320510100313930333430574370313930333430179790226
3200552205017104262003139301797903139303402002263200552203139303539903139300002602263200552
2031393020273033595022632067042051390034020040022031393031344000736033343031393022632004353
0340200055220313930202730313930226320426200313930000260313930513900340200226320268760313930
3334301797902263202027305017105139000552205017104002204262003359503334302687603402002263203
1344034020022632004353034020035399031393026876031393000026022632034020033343034020053555

Figure 7: Encryption results with RSA method optimized with a 6-bit padding system

Figure 8: Encryption process 0.10040569305419922 second(s).

Based on the results of the tests that have been carried out, it can be seen that the optimized steganography method shows advantages in terms of encryption time, because it takes a shorter time compared to conventional steganography methods. Interestingly, in the process of inserting messages, the optimized method actually makes more changes to the image, because the number of characters inserted is more. This is possible because character changes are only made in the range of numbers 0 to 9. In contrast to regular steganography which inserts characters in the form of letters and symbols, the implementation of the Least Significant Bit (LSB) method must change 2 to 3 bits in the image, which has an impact on longer encryption times.

In general, the steganography results of both methods are still quite good and the message insertion results are still well disguised. This is due to the pixel changes that occur in the image relatively small compared to the overall dimensions of the image, so it does not have a significant visual impact.

In terms of decryption, the optimized steganography method again shows advantages, although the time required to decrypt is longer than the usual method. This is due to the need to find prime number pairs that fit certain conditions in order to unlock decryption. This process is much safer because it involves several important aspects and parameters that only the sender and receiver of the message know. These parameters include private keys for decryption, precise partition padding (where a slight error can produce different results), as well as concepts in number theory that are not commonly known to the public.

IV. CONCLUSION

Based on the results of the research that has been conducted, it can be concluded that the integration of RSA algorithms and Pseudo-Random Prime Number Generator in steganography techniques is able to improve the security and efficiency of the insertion and message retrieval process in digital images. The optimized steganography method proved to be superior in terms of encryption speed, although it resulted in more changes in the image, but it was still able to maintain good visual quality. In addition, the decryption process is more secure because it requires special parameters such as private keys, specific prime numbers, and precise padding partitions, making it difficult for unauthorized parties to access the hidden information. Thus, this method can be an effective solution to maintain the confidentiality and integrity of data in digital communications.

REFERENCES

[1]. A. M. A. Saputra, L. P. I. Kharisma, A. A. Rizal, M. I. Burhan, and N. W. Purnawati, TEKNOLOGI INFORMASI: Peranan TI dalam berbagai bidang. PT. Sonpedia Publishing Indonesia, 2023.

[2]. I. W. Redhana, Literasi Digital: Pedoman Menghadapi Society 5.0. Samudra Biru, 2024.

[3]. V. Kumalasari, “Etika Profesi, Dalam Bidang Teknologi Informasi,” Penerbit Yayasan Prima Agus Tek., pp. 1–75, 2021.

[4]. D. Darwis, W. Wamiliana, and A. Junaidi, “Proses Pengamanan Data Menggunakan Kombinasi Metode Kriptografi Data Encryption Standard dan Steganografi End Of File,” in Prosiding Seminar Nasional METODE KUANTITATIF, 2017.

[5]. H. Putri, L. Virna, T. Febrianti, and T. Sutabri, “Pengamanan Data Transmisi Aplikasi Web Menggunakan Algoritma Kriptografi RSA: Studi Kasus dan Analisis,” J. Manaj. Inform. Teknol., vol. 5, no. 1, pp. 153–170, 2025.

[6]. N. B. N. Putra, F. A. Raihana, W. M. A. Mondong, and A. R. Kardian, “Analisis Enkripsi Kriptografi Asimetris Algoritma RSA Berbasis Pemrograman Batch pada Media Flashdisk,” Jurasik (Jurnal Ris. Sist. Inf. dan Tek. Inform., vol. 8, no. 1, pp. 142–154, 2023.

[7]. N. A. Fauzi, “Analisis Pengembangan Aplikasi Menggunakan Algoritma RSA dan El-Gamal Pada Teknik Steganografi Dengan Metode Least Significant Bit (LSB),” eProceedings Eng., vol. 12, no. 2, pp. 1–12, 2025.

[8]. A. Siahaan, “High Complexity Bit-Plane Security Enchancement in BPCS Steganography,” 2016.

[9]. L. P. Malese, “Penyembunyian Pesan Rahasia Pada Citra Digital dengan Teknik Steganografi Menggunakan Metode Least Significant Bit (LSB),” J. Ilm. Wahana Pendidik., vol. 7, no. 5, pp. 343–354, 2021.

[10]. R. P. Harahap and A. H. Hasugian, “Teknik Keamanan Data Menggunakan Metode Vigenere Cipher Dan Steganografi Dalam Penyisipan Pesan Teks Pada Citra,” J. FASILKOM, vol. 13, no. 3, pp. 570–577, 2023.

[11]. G. M. Fahmi, K. N. Isnaini, and D. Suhartono, “Implementation of Steganography on Digital Image With Modified Vigenere Cipher Algorithm and Least Significant Bit (Lsb) Method,” J. Tek. Inform., vol. 4, no. 2, pp. 333–344, 2023.

[12]. A. W. Laksono, S. Suhada, and A. Zakaria, “Implementasi Metode Least Significant Bit (Lsb) Dalam Teknik Steganografi Pada Citra Digital Menggunakan Matlab,” Diffus. J. Syst. Inf. Technol., vol. 4, no. 1, 2024.

[13]. M. A. Verdiana, I. Suarjaya, and A. Wiranatha, “Implementasi Algoritma PRNG pada Aplikasi Port Knocking Sebagai Perlindungan Server,” vol, vol. 8, p. 12, 2020.

[14]. A. P. U. Siahaan, “Factorization Hack of RSA Secret Numbers,” 2016.

[15]. A. P. U. Siahaan, B. O. Elviwani, and B. Oktaviana, “Comparative analysis of rsa and elgamal cryptographic public-key algorithms,” in Proceedings of the Joint Workshop KO2PI and The 1st International Conference on Advance & Scientific Innovation, 2018, pp. 162–171.