

Message Security Using Rivest-Shamir-Adleman Cryptography and Least Significant Bit Steganography with Video Platform

Widad Muhammad¹, Danang Haryo Sulaksono², Siti Agustini^{3*}

^{1,2,3}Institut Teknologi Adhi Tama Surabaya, Surabaya, 60117, Indonesia

¹muh.widad@gmail.com; ²danang_h_s@itats.ac.id; ³sitiagustini@itats.ac.id*

*Corresponding Author

ABSTRACT

All over the world, information technology has developed into a critical communication medium. One of them is digital messaging. We can connect and share information in real-time using digital messages. Without us knowing it, advances in message delivery are not only followed by kindness. Message security threats are also growing. Many unauthorized parties try to intercept critical information sent for the benefit of certain parties. As a countermeasure, various message security techniques exist to protect the messages we send. One of them is cryptography and steganography. Cryptography is useful for converting our messages into coded text so that unauthorized parties cannot read them. Meanwhile, steganography is useful for hiding our encrypted messages into several media, such as videos. This research will convert messages into ciphertext using the Rivest-Shamir-Adleman method and then insert them into video media using the Least Significant Bit method. There are four types of messages tested with different sizes. All messages will be encrypted and embedding using the Python programming language. Then the video will be tested using the MSE, PSNR, and Histogram methods. So, we get a value that shows which message gets the best results. So that the message sent is more guaranteed authenticity and reduces the possibility of message leakage.

Keywords : Cryptography, Steganography, Least Significant Bit (LSB), Rivest-Shamir-Adleman (RSA), Video.

Article History

Received : October, 18th 2020

Revised : November, 29th 2020

Accepted : November, 29th 2020

I. INTRODUCTION

A message is a form of communication between humans indirectly. Messages can be in the form of text, pictures, sound, or video. Messages can be sent via several media. Message security is essential to maintain the privacy of the message owner. Many methods can be used to protect messages. Cryptography is a method used to secure data. In general, cryptography is the science and art of keeping information confidential. Cryptography uses encryption algorithms, some of which have been classified as symmetric cryptography and asymmetric cryptography [1].

In previous research, secret message hiding was encrypted using the RSA (Rivest Shamir Adleman) method and inserted into an image file using the EOF (End of File) method with a 24-bit image at the last position of each image bit. This process produces an image file that does not significantly change the quality of the original image. This method's weakness is that there is a difference in the resulting image's bottom line compared to the original image. The larger the actual image size, the smaller the lines [1]. Authors of [2] have proposed RSA usage to encrypt messages and XOR to hide ciphertext into the image. And Author of [6] describes research on the implementation of LSB steganography on video media.

This research emphasizes the implementation of Rivest-Shamir-Adleman (RSA) cryptography and Least Significant Bit (LSB) steganography into video media. The coded message is directly inserted into the video media by replacing each pixel's last bit in the frame. The quality of the video that has been inserted is similar to normal video. The difference between the previously inserted video, and what was already inserted was only 1 bit different. The color on each pixel is 1 level lighter or darker, the message will not appear while the video is playing. The number of messages that will be implemented is four types with different sizes. Messages are encrypted and embedded into video media using the Python programming language. The stego video results will later be tested using the MSE, PSNR, and Histogram methods so that you can see which stego video is considered the best based on the 3 test methods. If sending messages via the internet uses the RSA cryptographic method and LSB steganography to video, the message integrity is better maintained. This has an impact on reducing the possibility of message leakage by unauthorized parties.

II. METHOD

This system overview will explain how the system works to facilitate implementation. Input, process, and output data from the implementation of the RSA and LSB algorithms can be seen in Figure 1. This research uses a video from YouTube for the experiments.

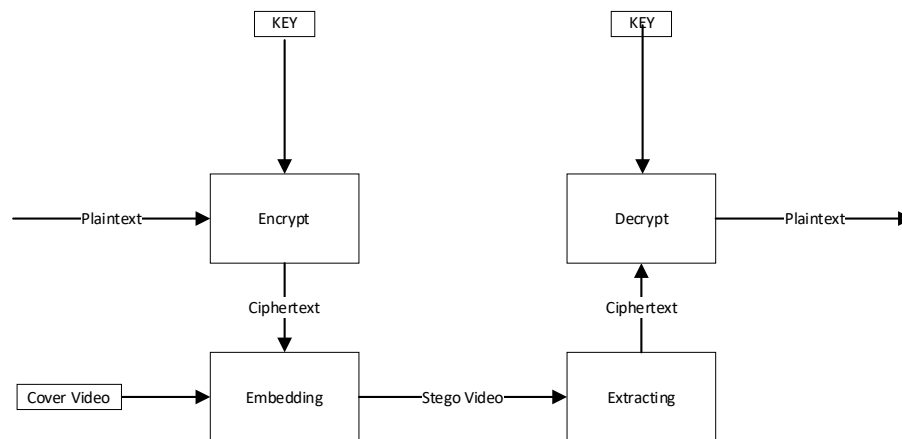


Fig. 1. General description

A. Cryptography

Cryptography is a method of encoding messages so that messages from the sender cannot be read by other parties except the receiver (receiver) safely. There are two ways to execute it, namely, Encryption and Decryption, as seen in figure 2. Encryption or enciphering is a method of encoding plaintext into ciphertext. At the same time, Decryption is the opposite of Encryption. Decryption or deciphering is a method that functions to convert ciphertext into plaintext.

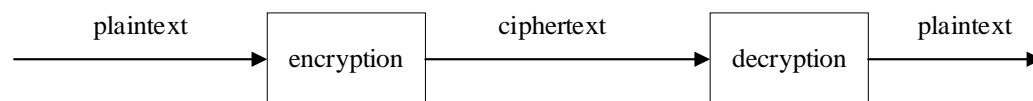


Fig. 2. Encryption and Decryption

Cryptography, apart from maintaining the confidentiality of messages, is also used to solve security problems, which include the following two things:

1) User Authentication

This point has to do with the authenticity of the sender. This problem can be expressed as a question:
"Is it true that the message received is really from the original sender?"

2) Message Authentication

This point has to do with the integrity of the message (data integrity). This problem can be expressed as a question:
"Is the message received is not subject to modification?"

3) Nonrepudiation

The sender cannot escape if he is the one who sent the message [3].

Symmetric cryptography is often called classical cryptography because the method of encryption and description uses the same key. Figure 3 shows how symmetric cryptography works.

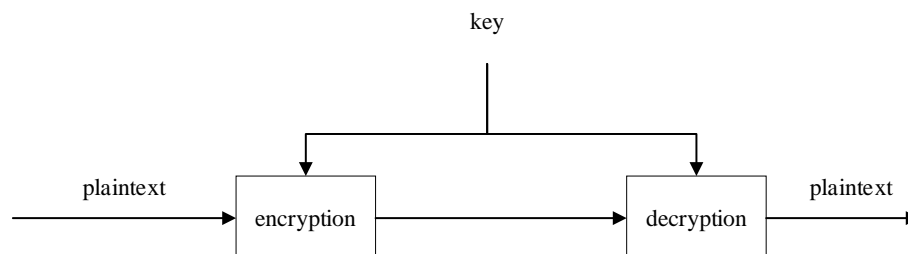


Fig. 3. Symmetric Algorithm

Asymmetric cryptography is often referred to as public-key cryptography. The keys used for Encryption and Decryption are different. In cryptography, the key asymmetry is divided into 2 (two) parts:

1) A public key is a key that can and can be known by everyone.

2) A private key is a key that only the recipient can know and is confidential [4].

Figure 4 shows that the public key is used to encrypt, and the private key is used to the decryption process.

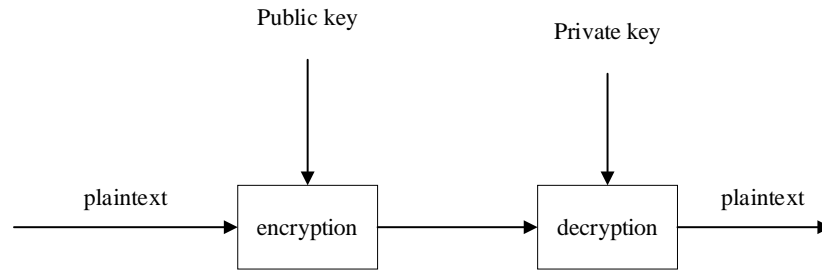


Fig. 4. Asymmetric Algorithm

B. Rivest-Shamir-Adleman (RSA)

Rivest Shamir Adleman, commonly abbreviated as RSA, is a modern cryptographic method created by three researchers from the Massachusetts Institute of Technology (MIT) in 1976. RSA is said to be difficult to solve due to factoring large numbers into its prime factors. Regarding factoring: Factor n , where n is the product of two or more prime numbers. There are several properties used in the RSA algorithm [3]. The following properties in Algorithm 1 will be used:

Algorithm 1 : Rivest-Shamir-Adleman (RSA)

- 1) p and q are the prime numbers. p and q must not be $p = q$ because the value of $n = p^2$.
 - 2) $n = p * q$
 - 3) $\phi(n) = (p - 1)(q - 1)$
 - 4) $1 < e < \phi(n)$. Prove it by calculating $\text{GCD}(e, \phi(n)) = 1$
 - 5) $d = \frac{1+k\phi(n)}{e}$
 - 6) $m = \text{plaintext}$
 - 7) $c = \text{ciphertext}$
 - 8) Encryption : $c_i = m_i^e \bmod n$
 - 9) Decryption : $m_i = c_i^d \bmod n$
-

C. Steganography

Steganography is a security method that functions as the insertion of secret messages into a medium (text, images, sound, or video) so that the message cannot be identified by humans normally. Steganography requires two ingredients for its implementation. Namely, the container media and data that will be hidden [3]. Figure 5 is an example of a cover image that is used to hide data. Figure 6 is the result after the secret information is inserted into Figure 5. At a glance, there is no difference between figure 5 and figure 5. It means the cover image can hide the secret information properly.



Fig. 5. Cover Image



Fig. 6. Stego image

Steganography recognizes several essential terms in its operation. The following terms will be used:

1. Message, message/data that you want to hide.
2. Cover-Object, the original container media. It can be in the form of pictures, sounds, videos, etc.
3. Stego-Object, media that has a message inserted into it.
4. Embedding (Encrypt). The process of inserting a message into the cover object.
5. Extracting (Decrypt). The process of retrieving messages from stage-objects [5].

D. Least Significant Bit (LSB)

Steganography has several data hiding techniques. The simplest method is the LSB (Least Significant Bit) method. In the bit

array, there is a byte (1 byte = 8 bits). There are two types of bit classification, namely, the most significant bit (MSB) and the least significant bit (LSB) [3]. Figure 7 shows the differences between LSB and MSB. MSB is located at the first bit of sequence bit. Meanwhile, LSB is located at the end of the sequence bit.

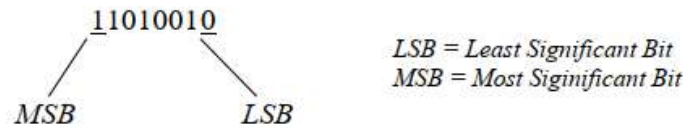


Fig. 7. Difference between MSB and LSB

E. Digital Video

Video is a live image produced by recording people and objects using a camera's help and has a second-dimensional function formed from recording in a place. Generally, videos are divided into two types, namely:

- 1) Analog is a video captured (recording) of the camera lens to an object vertically and horizontally.
- 2) Digital is a video that represents a matrix in which each element represents an absolute intensity value.

Digital video is created from a series of frames. A series of frames is played on the screen at a specific rate, according to the specified frame rate (in frames per second / FPS). When the FPS is high enough, the human eye cannot capture the image or frame in isolation but rather capture it as a continuous series of frames (video) [6].

F. Python

Guido Van Rossum developed the Python programming language from CWI, Amsterdam in 1990, to continue developing the ABC programming language. CWI released its final version in version 1.2. In 1995, Guido moved to CNRI while developing Python. The latest version released at that time was 1.6. In 2000, Guido and the core Python developers moved again to a commercial company called BeOpen.com and formed Be Open Python Labs. The company released its new version in 2.0. When the version was removed, Guido and several Python Labs members moved to Digital Creations. Python Software Foundation is a non-profit organization established for holders of Python intellectual property since version 2.1 so that other chemical companies do not own it. For now, the python distribution is stable version 3.9. [7].

G. Mean Square Error (MSE)

MSE is a widely accepted method of control and quality measurement. This MSE can be calculated utilizing an example object and then compared with the original object. The level of inequality between the sample object and the original can be seen. The MSE can be obtained by equation (1).

$$MSE = \frac{1}{MN} \sum_{y=1}^M \sum_{x=1}^N [I(x, y) - I'(x, y)]^2 \quad (1)$$

Where M, N are the dimensions of the image. I (x, y) is the pixel value of the original image. Meanwhile, I '(x, y) is the pixel value of the stego image [8].

H. Peak Signal Noise Ratio (PSNR)

PSNR is the maximum value between the measured signal and the amount of noise that affects the signal. The unit of measure for the PSNR is decibels (dB). The function of the PSNR is to compare the quality of the cover image and the stego image [9]. Calculate PSNR using equation 2.

$$PSNR = 10 \log \frac{MAX_i^2}{MSE} \quad (2)$$

I. Histogram

The histogram is a graph depicting the distribution of pixel intensity usage. The histogram also shows that the image can be said to be dark or light. The meaning of dark here is the number of graphs that stand out on the left, while the light is the number of graphs on the right. This test analysis will compare the histogram cover image with the stego image [10].

J. Flowchart Applications

This system analysis describes the flow of the system work process to facilitate implementation: input, process, and output data in Figure 7 and Figure 8.

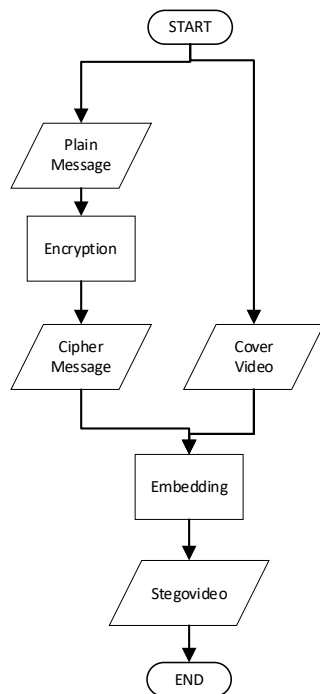


Fig. 8. Sender Flowchart

Based on the sender side system flowchart in Figure 8, it can be explained as follows:

1. Start the program from the sender side.
2. The sender enters the message into the program. The message input is a text message.
3. The next process is the encryption process. Each character is converted using the RSA cryptographic method so that the sender gets the ciphertext results in a set of numbers.
4. The sender gets the encrypted text message. Messages appear automatically on the console terminal in the form of numbers.
5. The sender enters the desired video file name. Make sure the video has been included in a directory with the program running.
6. The process of inserting the ciphertext into the video begins using the LSB method. The message is divided into several parts and then inserted into several video frames until the last part of the message.
7. The sender gets the result of the embedding, which becomes a stego video file in the same directory as the program.
8. Sender side processing is complete.

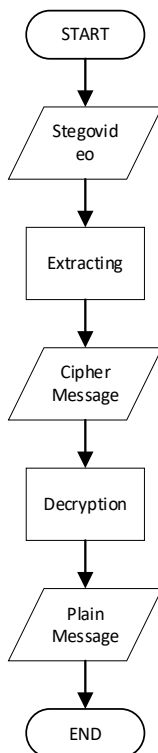


Fig. 9. Receiver Flowchart

Based on the receiver side system flowchart in Figure 9 it can be explained as follows:

1. Start programming from the receiving end.
2. The recipient enters the name of the stego video file obtained from the sender. Make sure the files are included in the same directory as the program.
3. The next process is the process of retrieving hidden messages from each frame using the LSB method. The message obtained will be in the form of ciphertext.
4. The recipient gets the message extracted in the form of an encrypted message on the console terminal.
5. Then the ciphertext will be decrypted using the RSA method so that it gets the plaintext. The plaintext will appear in the console terminal.
6. The recipient gets a decrypted message in the form of the original message from the sender.
7. Receiving side processing is complete.

III. RESULT AND DISCUSSION

The following are the stages of a manual count simulation using the help of the Microsoft Office Excel 2019 application and an embedding simulation:

1. $p = 17$ dan $q = 23$
2. $n = p * q = 391$
3. $\phi(n) = (p - 1)(q - 1) = 352$

4. $1 < e < \phi(n)$. We choose $e = 3$. Prove it by counting $\text{GCD}(e, \phi(n)) = 1$ as seen as table I.

TABLE I
THE PROOF TABLE E = 3

352	=	117	×	3	+	1
3	=	3	×	1	+	0

5. Calculate the value of d . The value of k starts from 1 to $k \times n$. So that the value of d is found to be an integer. Table II shows that d is obtained when $k=2$. It can be concluded that the value of $d = 235$

TABLE II
CALCULATION TABLE D

k	$\phi(N)$	e	d
1	352	3	117,6667
2	352	3	235

6. The encryption key (3, 391) and decryption key (235, 391)
 7. Plaintext = "Disqualified as a human because"
 8. Encryption : $C_i = m_i^e \bmod n$
 Ciphertext = 68, 265, 276, 107, 77, 79, 301, 265, 34, 265, 16, 213, 315, 79, 276, 315, 79, 315, 348, 77, 37, 79, 36, 315, 55, 16, 228, 79, 77, 276, 16, 315
 9. Decryption : $m_i = c_i^d \bmod n$
 Plaintext = 'D', 'i', 's', 'q', 'u', 'a', 'l', 'i', 'f', 'i', 'e', 'd', ' ', 'a', 's', ' ', 'a', ' ', 'h', 'u', 'm', 'a', 'n', ' ', 'b', 'e', 'c', 'a', 'u', 's', 'e', ' '

The MSE test was carried out on a video cover with a resolution of $640 * 360$ px, has a framerate of 25fps, and is about 10 seconds long. Each message is calculated in search of the MSE value. Where the smaller the MSE value, the better. The frames taken for the test sample in each video are frame 0 and frame 1. Table III explains that the average MSE value of each video. From this table, it can also be concluded that the average MSE value is 0.0355, with the lowest MSE value of 0.0066 on the video output4.mov. In Figure 10, you can see the graph of the MSE results were. The more significant the size of the message inserted, the higher the MSE value. In message four, there is a significant decrease in value, touching the value of 0.0066, and in message 3 there is an increase in the value of 0.0534. It is different from the previous MSE test, where the smaller the MSE value, is better. Meanwhile, the higher the PSNR value is better. Table IV explains that the average PSNR value for each video. From this table, it can also be concluded that the average PSNR value is 146.8078, with the highest PSNR value of 161.0298 on the video output4.mov. In Figure 11, you can see a graph of the PSNR results where the smaller the message size is inserted, the higher the PSNR value. In message three, there was a significant decrease in value, touching the value of 140.1306, and in message 4 there was an increase in the value of 161.0298. This is because the fewer messages that are inserted into the video, the fewer changes that occur in the video. In this test, the histogram method is used, which is a technique of taking the difference in the frequency of color usage between the cover image and the stego image. The results of histogram analysis can be seen in Table V and Table VI below.

TABLE III
MSE RESULTS TABLE

No	Cover Video		Message Type (Bytes)	Stego Video			MSE
	File Name	Frame Total		Stego Name	File Size (KB)	Total Frame	
1	sample.mp4	251	Message 1 (309)	output1.mov	73455	251	0,05015
2		251	Message 2 (198)	Output2.mov	73454	251	0,03185
3		251	Message 3 (343)	output3.mov	73455	251	0,0534
4		251	Message 4 (32)	output4.mov	73452	251	0,0066
Total MSE							0,0355

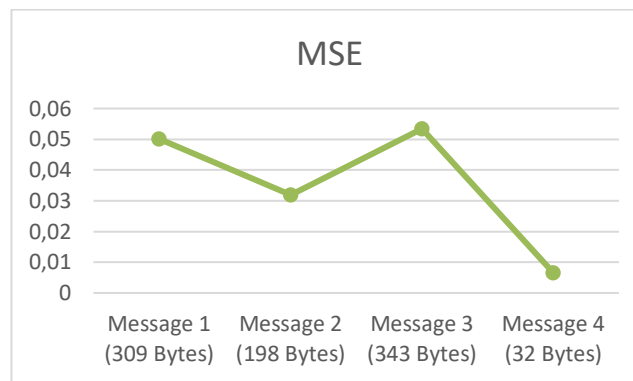


Fig. 10. MSE Result Graph

TABLE IV
PSNR RESULTS TABLE

No	Cover Video		Message Type (Bytes)	Stego Video			PSNR
	File Name	Frame Total		Stego Name	File Size (KB)	Total Frame	
1	sample.mp4	251	Message 1 (309)	output1.mov	73455	251	140,7604
2		251	Message 2 (198)	Output2.mov	73454	251	145,3103
3		251	Message 3 (343)	output3.mov	73455	251	140,1306
4		251	Message 4 (32)	output4.mov	73452	251	161,0298
						Total MSE	146,8078

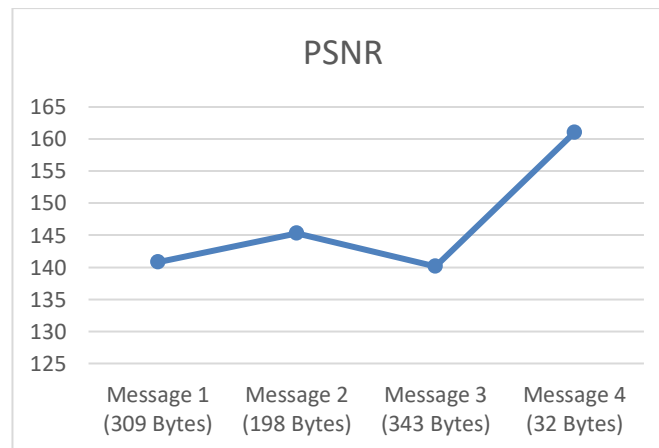


Fig. 11. PSNR Result Graph

TABLE V
COVER VIDEO HISTOGRAM RESULT TABLE


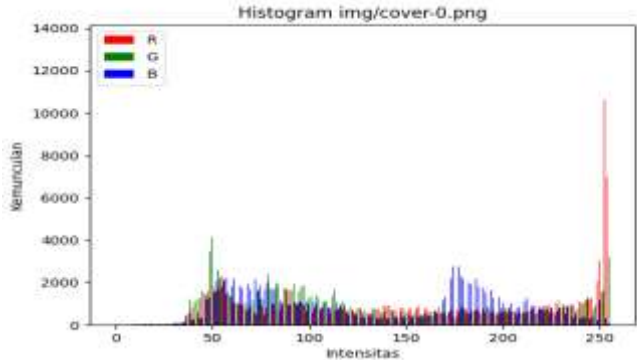

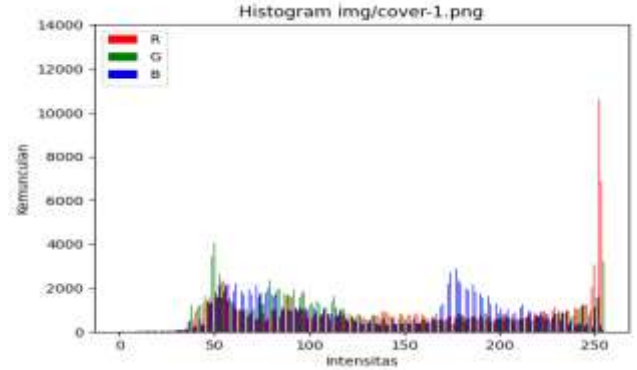
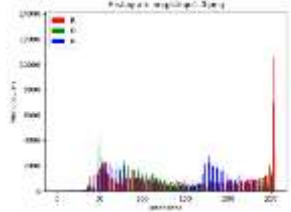
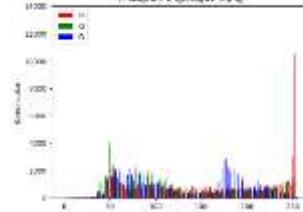
No	Image	Histogram
1	 Fig. 11a. <i>cover-0.png</i>	 Fig. 11b. histogram <i>cover-0.png</i>
2	 Fig. 12a. <i>cover-1.png</i>	 Fig. 12b. histogram <i>cover-1.png</i>

Table V explains the frequency of RGB colors from each frame 0 and 1 on the video cover. At first glance, the use of color in the two frames above is similar because they are located next to each other. In comparison, the table of the results of the use of color on stego video for each type of message can be seen in Table VI.

TABLE VI
COVER VIDEO HISTOGRAM RESULT TABLE

No	Stego Video Name	Histogram	
		Frame 0	Frame 1
1	<i>output1.mov</i>	 Fig. 13a. frame 0 stego 1	 Fig. 13b. frame 1 stego 1

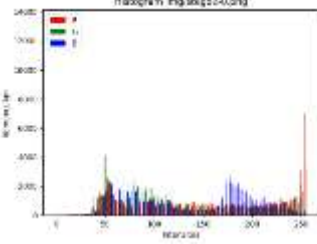
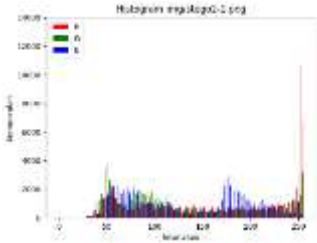
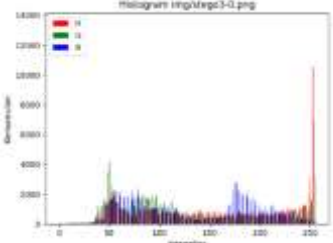
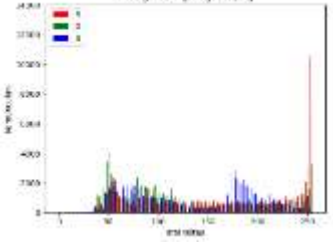
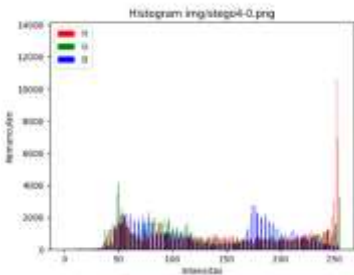
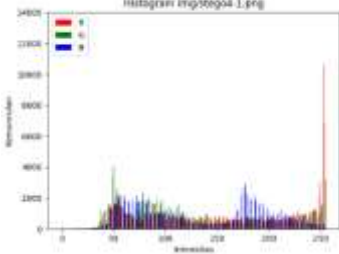
No	Stego Video Name	Histogram	
		Frame 0	Frame 1
2	<i>output2.mov</i>	 Fig. 14a. frame 0 stego 2	 Fig. 14b. frame 1 stego 2
3	<i>output3.mov</i>	 Fig. 15a. frame 0 stego 3	 Fig. 15b. frame 1 stego 3
4	<i>output4.mov</i>	 Fig. 16a. frame 0 stego 4	 Fig. 16b. frame 1 stego 4

Table VI above explains the RGB color histogram's comparison for each sample frame on the four video message insertions. In the two histogram tables above, there is very little change in the graph's shape if you look at it in detail. In inserting the message, it is inserted into the bits of each pixel so that it changes slightly up and down. The larger the message size is inserted, the more changes the histogram graph will have [11].

IV. CONCLUSION

The implementation of message security can use the Rivest-Shamir-Adleman cryptographic method and Least Significant Bit steganography into video media. From MSE and PSNR, it was found that the video stego which had the other best quality was the *output4.mov* video with an MSE result of 0.0066 and a PSNR result of 161.03. Because the MSE value on the video is lower and the closest to the value of 0. In comparison, the PSNR on the video has a higher value than others. Whereas the histogram tester can be taken if the larger the message size is inserted, the more changes will be made to the histogram graph. In future studies, it is expected to use the LSB method's modification to reduce the change in the value of the bits at each pixel significantly. Trial scenarios were also carried out with different video resolutions to get better conclusions than this research.

REFERENCES

- [1] Rachmawati, Dian, Amalia, Amalia & Elviwani. "Combination of Rivest-Shamir-Adleman Algorithm and End of File Method for Data Security". Indonesia: IOP Access, 2018, pp. 1
- [2] Agustini, S., Kurniawan, M., "Peningkatan Keamanan Teks Menggunakan Kriptografi Dan Steganografi", SCAN, Vol. XIV, No.3, 2019.
- [3] Munir, Rinaldi. (2006). Kriptografi. Bandung: Informatika.
- [4] Putri, Geby G., Styorini, Wiwin & Rahayani, R. Dian. "Analisis Kriptografi Simetris AES dan Kriptografi Asimetris RSA pada Enkripsi". Indonesia: Jurnal Informatika Mulawarman, 2018, pp. 198.

- [5] Yatini, Indra & Nurwiyati, F. Wiwiek. "Algoritma Least Significant Bit Untuk Analisis Steganografi". Indonesia: Seminar Nasional Informatika, 2015, pp. 695.
- [6] Anti, Ulan A., Krisdalaksana, Awang H. & Khairina, Dyna M. "Steganografi Pada Video Menggunakan Metode Least Significant Bit (LSB) Dan End Of File (EOF)". Indonesia: Jurnal Informatika Mulawarman, 2017, pp. 107-108.
- [7] Insanudin, E. "Implementation of Python Source Code Comparison Results with Java using Bubble Sort Method". Indonesia: IOP Access, 2018, pp. 4.
- [8] Yunus, Mahmuddin & Harjoko, Agus. "Penyembunyian Data pada File Video Menggunakan Metode LSB dan DCT". Indonesia: Universitas Gadjah Mad., 2014, pp. 85-86.
- [9] Male, Ghazali Moenandar, Wirawan & Setijadi, Eko. "Analisa Kualitas Citra Pada Steganografi Untuk Aplikasi e-Government". Indonesia: Digilab Institut Teknologi Sepuluh November, 2012, pp. 4.
- [10] Murinto, Putra, Willy Permana & Handyaningsih. "Analisis Perbandingan Histogram Equalization Dan Model Logarithmic Image Processing (LIP) Untuk Image Enhancement". Indonesia: Jurnal Informatika, 2008, pp. 204.
- [11] Hasan, Nur Fitriyaningsih, Dengan, Christin Nandari & Ariyus, Dony. Analisis Histogram Steganografi Least Significant Bit Pada Citra Grayscale. Indonesia: Digital Zone., 2020, pp. 26-27.