

EV 3

by Diperiksa Oleh Lppm Universitas Potensi Utama

Submission date: 17-Jun-2020 04:07PM (UTC+0700)

Submission ID: 1345297177

File name: EV_3.pdf (304.36K)

Word count: 3800

Character count: 16647

2 The Application of RSA and LSB in Securing Message on Images

Bob Subhan Riza¹, M. Y. Mashor², Edy Victor Haryanto S³

Universitas Potensi Utama^{1,3}, University Malaysia Perlis²

E-mail : bob.potensi@gmail.com, edyvictor@gmail.com

AJRI
ADI JOURNAL ON RECENT INNOVATION



Author Notification
31 August 2019
Final Revised
01 September 2019
Published
03 September 2019

2 cite this document :

Riza, B., Mashor, M., & Haryanto, E. (2019). APPLICATION OF RSA AND LSB IN SECURITY OF MESSAGES ON IMAGERY. *ADI Journal on Recent Innovation (AJRI)*, 1(1), 20-32. <https://doi.org/https://doi.org/10.34306/ajri.v1i1.10>

DOI:

<https://doi.org/10.34306/ajri.v1i1.10>

Abstract

In this study is to discuss cryptography and steganography where the function is to insert a message or text into an image with JPG extension, the text to be inserted into the image has been encrypted using the RSA method so that the file is safer to be inserted into images, messages that are inserted into a blue image, this application aims to secure a message that you want to save, this application is made made using Android Studio and can be run on a mobile phone.

Keywords : Security, Cryptography, Steganography, RSA, LSB

I. INTRODUCTION

Currently the development of technology has been very good and very fast and almost human activity is inseparable from the role of technology both for daily work at home or work in the office so that humans really need technology. Therefore the security of the technology used must also be increased, especially in terms of data that utilizes technology so that the data that we store in these technological devices can be protected and safe.

Cryptography is a method for securing data where the data that is secured is converted into a form that has no meaning so that others cannot find out the data. The RSA algorithm is one part of cryptography where RSA uses very difficult calculations to produce encrypted ciphertexts from secured data. It takes a long time to find out the meaning of the original data from a ciphertext from RSA encryption algorithm without knowing the correct key pair. Therefore the RSA algorithm is still the safest cryptographic algorithm to date.

Steganography is the development of cryptographic methods in which steganography hides data into the cover object so that its existence is not known by others. LSB (Least Significant Bit) is one of the steganographic methods that insert data on the last bits of the cover object so that changes before and after the insertion process are not visible to the human senses. This is because the change in value of each pixel on the cover object only increases 1 or decreases by 1 value.

The combination of cryptography and steganography has been done in previous studies, this shows that the combination has a high level of security so that the development of methods and algorithms is also widely carried out. The combination of RSA and LSB is considered very appropriate because RSA cryptography is the safest algorithm to date and LSB steganography is needed to hide the ciphertext results from RSA cryptography into images so that its presence is unknown to others. [3]

Barkah in his research designed an application in the field of steganography in securing images using the F5 method. [4]

II. METHODE

2.1. Cryptography

Cryptography (cryptograph⁸) comes from Greek: "cryptos" means "secret" writing "Cryptography is defined as the science and art of maintaining the confidentiality of messages by encoding them into a form⁷ that cannot be understood anymore. Cryptography has two concepts The main thing is encryption and decryption Encryption is the process of encoding plaintext⁶ to ciphertext, whereas decryption is the process of returning ciphertext to the original plaintext Encryption and decryption require keys as parameters used for transformation.

Cryptography is divided into 2 (two), namely:

1. Classic cryptography (character mode):
 - Cipher Substitution
 - Transposition Cipher
2. Modern cryptography (bit / binary mode):
 - Symmetry key ciphers: stream ciphers, block ciphers
 - Public key cryptography [1.]

2.2. RSA

In 1977, Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman formulated a practical algorithm that implements a public-key cryptog⁴raphic system called the RSA cryptographic system. ⁴he pair of keys used in both processes are the public key (e, n) as the encryption key and the private key as the decryption key where e, d and n are positive integers. Th⁴ RSA algorithm is a block cipher algorithm (an algorithm that works per block of data) which groups plaintext into blocks before encryption is made to ciphertext [2]

III. RESULT AND DISCUSSION

In the RSA cryptographic process, input the message as a plaintext is ASCII characters that is 255 characters. The results obtained from the encryption process are ciphertext in the form of decimal values generated from the calculation of the RSA algorithm encryption formula. After the ciphertext is obtained it is converted into a binary form which will then be inserted in each pixel cover object in the form of an RGB image, in this case a blue pixel. Image image used in the format *.jpg with image

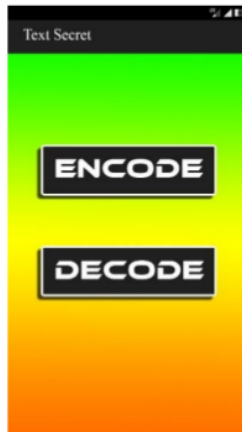


Figure 1. Main Menu Layout

In the main menu layout of the application, the user can choose a menu on the application for further processing by the system. There are 2 main menus, namely the encode menu which will encrypt the message which is then inserted into the image image, and the decode menu which will extract the message from the image that produces the ciphertext and then decrypt it using the RSA algorithm that produces a plaintext in the form of the original message. The following encoding menu layout can be seen in Figure 2 :



Figure 2. Layout Menu Encode

In the encode menu the user can encrypt and insert messages where as an example of this research, the message used is "Great Indonesia". The first step the user must first select the cover object that will be used by clicking on the Image button, the system will open the system gallery that will display images across all user devices. After the cover object is selected, then the next generation of key pairs by entering two primes that are not the same value, in this study used the values of 43 and 47, the following is a manual key generation calculation :

- a. Search value n :
- $$\begin{aligned} n &= p \times q \\ &= 43 \times 47 \\ &= 2021 \end{aligned}$$
- b. Search value ϕn :
- $$\begin{aligned} \phi n &= (p-1) \times (q-1) \\ &= (43-1) \times (47-1) \\ &= (42) \times (46) \\ &= 1932 \end{aligned}$$
- c. Search value e :
- $$e = 2$$
- While $\phi n \bmod e \neq 0$
- $$e = e + 1$$
- End While
- Proses 1 :
- $$\phi n \bmod e = 1932 \bmod 3 = 0$$
- $$e = 3 + 1$$
- $$e = 4$$
- Proses 2 :
- $$\phi n \bmod e = 1932 \bmod 4 = 0$$
- $$e = 4 + 1$$
- $$e = 5$$
- Proses 3 :
- $$\phi n \bmod e = 1932 \bmod 5 = 2$$
- $$e = 5$$
- d. Search value d :
- $$\begin{aligned} U1 &= 1 \\ U2 &= 0 \\ U3 &= \phi n \\ V1 &= 0 \\ V2 &= 1 \\ V3 &= e \\ \text{While } V3 &= 0 \\ Q &= \text{Int}(U3 / V3) \\ N1 &= U1 - (Q \times V1) \\ N2 &= U2 - (Q \times V2) \\ N3 &= U3 - (Q \times V3) \\ U1 &= V1 \\ U2 &= V2 \\ U3 &= V3 \\ V1 &= N1 \\ V2 &= N2 \\ V3 &= N3 \\ \text{End While} \end{aligned}$$

Process 1:

$$\begin{aligned} Q &= \text{Int}(U3 / V3) \\ &= \text{Int}(1932 / 5) \\ &= 386 \\ N1 &= U1 - (Q \times V1) \\ &= 1 - (386 \times 0) \end{aligned}$$

$$\begin{aligned}
 &= 1 \\
 N2 &= U2 - (Q \times V2) \\
 &= 0 - (386 \times 1) \\
 &= -386 \\
 N3 &= U3 - (Q \times V3) \\
 &= 1932 - (386 \times 5) \\
 &= 1932 - 1930 \\
 &= 2 \\
 U1 &= 0 \\
 U2 &= 1 \\
 U3 &= 5 \\
 V1 &= 1 \\
 V2 &= -386 \\
 V3 &= 2
 \end{aligned}$$

Process 2:

$$\begin{aligned}
 Q &= \text{Int}(U3 / V3) \\
 &= \text{Int}(5 / 2) \\
 &= 2 \\
 N1 &= U1 - (Q \times V1) \\
 &= 0 - (2 \times 1) \\
 &= -2 \\
 N2 &= U2 - (Q \times V2) \\
 &= 1 - (2 \times -386) \\
 &= 773 \\
 N3 &= U3 - (Q \times V3) \\
 &= 5 - (2 \times 2) \\
 &= 5 - 4 \\
 &= 1 \\
 U1 &= 1 \\
 U2 &= -386 \\
 U3 &= 2 \\
 V1 &= -2 \\
 V2 &= 773 \\
 V3 &= 1
 \end{aligned}$$

Process 3:

$$\begin{aligned}
 Q &= \text{Int}(U3 / V3) \\
 &= \text{Int}(2 / 1) \\
 &= 2 \\
 N1 &= U1 - (Q \times V1) \\
 &= 1 - (-2 \times -3) \\
 &= 5 \\
 N2 &= U2 - (Q \times V2) \\
 &= -386 - (2 \times 773) \\
 &= -1932 \\
 N3 &= U3 - (Q \times V3) \\
 &= 2 - (2 \times 1) \\
 &= 0 \\
 U1 &= -2 \\
 U2 &= 773 \\
 U3 &= 2 \\
 V1 &= 5 \\
 V2 &= -1932 \\
 V3 &= 0
 \end{aligned}$$

After the calculation process, the RSA algorithm's key pair is obtained with the value. Then the encryption process is then performed by changing the original message into the ciphertext with manual calculations as follows :

$C_i = P_i \text{ mod } n$
 C_i = Cipherteks
 P_i = Plainteks
 e = Nilai kunci e
 n = Nilai kunci n

Enkripsi Pertama :

$I = 73$
 $C_i = P_i \text{ mod } n$
 $= 735 \text{ mod } 2021$
 $= 2073071593 \text{ mod } 2021$
 $= 528$

Enkripsi Kedua :

$n = 110$
 $C_i = P_i \text{ mod } n$
 $= 1105 \text{ mod } 2021$
 $= 16105100000 \text{ mod } 2021$
 $= 1604$

Enkripsi Ketiga :

$d = 100$
 $C_i = P_i \text{ mod } n$
 $= 1005 \text{ mod } 2021$
 $= 10000000000 \text{ mod } 2021$
 $= 1055$

Enkripsi Keempat :

$o = 111$
 $C_i = P_i \text{ mod } n$
 $= 1115 \text{ mod } 2021$
 $= 16850581551 \text{ mod } 2021$
 $= 927$

Enkripsi Kelima :

$n = 110$
 $C_i = P_i \text{ mod } n$
 $= 1105 \text{ mod } 2021$
 $= 16105100000 \text{ mod } 2021$
 $= 1604$

Enkripsi Keenam :

$e = 101$
 $C_i = P_i \text{ mod } n$
 $= 1015 \text{ mod } 2021$
 $= 10510100501 \text{ mod } 2021$
 $= 1156$

Enkripsi Ketujuh :

$s = 115$
 $C_i = P_i \text{ mod } n$
 $= 1155 \text{ mod } 2021$
 $= 20113571875 \text{ mod } 2021$
 $= 1869$

Enkripsi Kedelapan :

$i = 105$

$$\begin{aligned}C_i &= \text{Pie mod } n \\&= 1055 \text{ mod } 2021 \\&= 12762815625 \text{ mod } 2021 \\&= 546\end{aligned}$$

Enkripsi Kesembilan :

$$\begin{aligned}a &= 97 \\C_i &= \text{Pie mod } n \\&= 975 \text{ mod } 2021 \\&= 8587340257 \text{ mod } 2021 \\&= 102\end{aligned}$$

Enkripsi Kesepuluh :

$$\begin{aligned}[\text{space}] &= 32 \\C_i &= \text{Pie mod } n \\&= 325 \text{ mod } 2021 \\&= 33554432 \text{ mod } 2021 \\&= 1790\end{aligned}$$

Enkripsi Kesebelas :

$$\begin{aligned}H &= 72 \\C_i &= \text{Pie mod } n \\&= 725 \text{ mod } 2021 \\&= 1934917632 \text{ mod } 2021 \\&= 106\end{aligned}$$

Enkripsi Ketigabelas :

$$\begin{aligned}e &= 101 \\C_i &= \text{Pie mod } n \\&= 1015 \text{ mod } 2021 \\&= 10510100501 \text{ mod } 2021 \\&= 1156\end{aligned}$$

Enkripsi Keempatbelas :

$$\begin{aligned}b &= 98 \\C_i &= \text{Pie mod } n \\&= 985 \text{ mod } 2021 \\&= 9039207968 \text{ mod } 2021 \\&= 507\end{aligned}$$

Enkripsi Kelimabelas :

$$\begin{aligned}a &= 97 \\C_i &= \text{Pie mod } n \\&= 975 \text{ mod } 2021 \\&= 8587340257 \text{ mod } 2021 \\&= 102\end{aligned}$$

Enkripsi Keenambelas :

$$\begin{aligned}t &= 116 \\C_i &= \text{Pie mod } n \\&= 1165 \text{ mod } 2021 \\&= 21003416576 \text{ mod } 2021 \\&= 270\end{aligned}$$

The encoding process is continued by inserting the RSA encryption algorithm results into the blue pixel cover object, where the encrypted cipher text is added with the character "|" as a delimiter between words to facilitate the process of decryption later. The ciphertext after adding

a delimiter is "5 2 8 1 2 4 1 6 0 4 1 2 4 1 0 5 5 1 2 4 9 2 7 1 2 4 1 6 0 4 1 2 4 1 1 5 6 1 2 4 1 8 6 9 1 2 4 5 4 6 1 2 4 1 0 2 1 2 4 1 7 9 0 1 2 4 1 0 6 1 2 4 1 1 5 6 1 2 4 5 0 7 1 2 3 1 0 2 1 2 3 2 7 0" the system will change the cover object to RGB where the system will use the pixel blue value as a medium to insert the ciphertext. Here is an example of insertion in a blue pixel : After the calculation process, the RSA algorithm's key pair is obtained with the value. Then the encryption process is then performed by changing the original message into the ciphertext with manual calculations as follows:

4	3	4	4	9	1	1	1
4	9	8	7	8	9	9	7
6	6	1	1	3	3	3	2
6	7	3	3	8	9	9	2
6	6	6	4	4	4	5	5
6	8	5	9	8	8	0	0
9	6	6	1	1	2	2	2
0	7	6	3	3	1	1	3
9	1	1	1	1	3	3	3
0	2	2	2	1	0	0	1
9	1	1	1	2	2	2	5
1	2	3	3	2	2	3	0

Table 3. Pieces of RGB pixel image value

After getting the cover object's RGB value, the system will separate it into 3 parts and only use one part, the blue pixel segment. The Following results are the separation of image pixel value :

4	4	9	1
8	7	8	9
1	1	3	3
3	3	8	9
6	4	4	4
5	9	8	8
6	1	1	2
6	3	3	1
1	1	1	3
2	2	1	0
1	1	2	2
3	3	2	2

Table 4. Pieces of blue pixel image

The result of the separation of image pixels is then converted into a binary form to do the insertion process in the image. The conversion results can be seen in the following table :

0011000	0010111	0110001	0001001
0	1	0	1
0000110	0000110	0011100	0011100
1	1	0	1
0011010	0011000	0011000	0011000
1	1	0	0
0011011	0000110	0000110	0010000
0	1	1	1
0000110	0000110	0000101	0001100
0	0	1	0

0000110 1	0000110 1	0001001 0	0001001 0
--------------	--------------	--------------	--------------

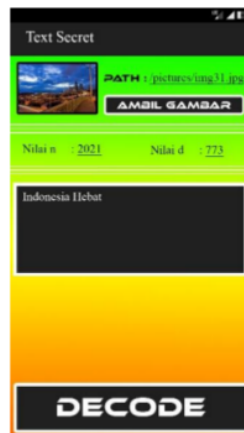
Table 5. Conversion of image pixel values

The next process is to change the message binary form, for example the message is "RSA" and converted to binary to "010100100101001101000001". The following is the result of inserting the message into the cover object.

0011000 0	0010111 1	0110001 0	0001001 1
0000110 0	0000110 0	0011100 1	0011100 0
0011010 0	0011000 1	0011000 0	0011000 1
0011011 0	0000110 0	0000110 1	0010000 1
0000110 0	0000110 1	0000101 0	0001100 0
0000110 0	0000110 0	0001001 0	0001001 1

Table 6. Results for inserting messages in blue pixels

After the process is complete, the system will store the cover object in storage and the system will display a dialog that "The message was successful in Encode"



Gambar 3. Layout Menu Decode

In the decode menu layout, the system will extract the cover object that has been inserted image by changing the image into the RGB form. The image is converted into the RGB form and then the value of each pixel is taken, the following is the value of the pixel image:

4	3	4	4	9	1	1	1
4	9	8	7	8	9	9	7
6	6	1	1	5	5	3	2
6	7	2	2	7	6	9	2

6	6	5	4	4	4	5	5
6	8	2	9	8	9	0	0
9	6	5	1	1	3	2	2
0	7	4	2	3	3	1	3
9	1	1	1	1	2	3	3
0	2	2	3	0	4	0	1
9	1	1	1	1	1	2	5
1	2	2	2	8	9	3	0

Table 7. RGB image values

After getting the RGB image value, the system will take the blue pixel value that has been pasted in the encoding process, the blue pixel value can be seen as follows :

4	4	9	1
8	7	8	9
1	1	5	5
2	2	7	6
5	4	4	4
2	9	8	9
5	1	1	3
4	2	3	3
1	1	1	2
2	3	0	4
1	1	1	1
2	2	8	9

Table 8. Blue pixel values

The blue pixel value is then converted to a binary number so that the last bits of the pixel can be taken as a secret message. Here are the results of the conversion of blue pixel values into binary forms:

0011000 0	0010111 1	0110001 0	00010011
0000110 0	0000110 0	0011100 1	00111000
0011010 0	0011000 1	0011000 0	00110001
0011011 0	0000110 0	0000110 1	00100001
0000110 0	0000110 1	0000101 0	00011000
0000110 0	0000110 0	0001001 0	00010011

Table 9. Results of blue pixel conversio

After being converted, every last bit of pixel is taken and put together to form a message Following is the process of taking the last bit of pixel:

0011000 <u>0</u>	0010111 <u>1</u>	0110001 <u>0</u>	00010011 <u>1</u>
---------------------	---------------------	---------------------	----------------------

0000110 <u>0</u>	0000110 <u>0</u>	0011100 <u>1</u>	0011100 <u>0</u>
0011010 <u>0</u>	0011000 <u>1</u>	0011000 <u>0</u>	0011000 <u>1</u>
0011011 <u>0</u>	0000110 <u>0</u>	0000110 <u>1</u>	0010000 <u>1</u>
0000110 <u>0</u>	0000110 <u>1</u>	0000101 <u>0</u>	0001100 <u>0</u>
0000110 <u>0</u>	0000110 <u>0</u>	0001001 <u>0</u>	0001001 <u>1</u>

Table 10. The process of extracting pixel bits

Then the binary message will be obtained as follows "010100100101001101000001" which if converted will produce the message "RSA". Furthermore, the system will perform the decryption process by changing the encrypted ciphertext into a plaintext again by using the key pairs and values which were first raised in the previous encryption process. For example there is a ciphertext which is "5 2 8 1 2 4 1 6 0 4 1 2 4 1 0 5 5 1 2 4 9 2 7 1 2 4 1 6 0 4 1 2 4 1 1 5 6 1 2 4 1 8 6 9 1 2 4 5 4 6 1 2 4 1 0 2 1 2 4 1 7 9 0 1 2 4 1 0 6 1 2 4 1 1 5 6 1 2 4 5 0 7 1 2 4 1 0 2 1 2 4 2 7 0 ", The decryption process will be carried out with keys n and d, namely 2021 and 773. First the ciphertext is separated from the character marker" | "to get the results of the character, then after the separation, the ciphertext "

$P_i = C_{id} \bmod n$
 P_i = Plainteks
 C_i = Cipherteks
 d = Nilai d
 n = Nilai n

Then the result of the calculation is that P_i , which is an ASCII decimal value, is converted to ASCII characters so that the plaintext can be read again.

Dekripsi Pertama :

$C_i = 528$
 $P_i = C_{id} \bmod n$
 $= 528773 \bmod 2021$
 $= 3,9447739231444509389840801901531e+2104 \bmod 2021$
 $= 73 = I$

Dekripsi Kedua :

$C_i = 1604$
 $P_i = C_{id} \bmod n$
 $= 1604773 \bmod 2021$
 $= 4,1973320925821901822743614620265e+2477 \bmod 2021$
 $= 110 = n$

Dekripsi Ketiga :

$C_i = 1055$
 $P_i = C_{id} \bmod n$
 $= 1055773 \bmod 2021$
 $= 9,422177834300404556852404327723e+2336 \bmod 2021$
 $= 100 = d$

Dekripsi Keempat :

$C_i = 927$
 $P_i = C_{id} \bmod n$
 $= 927773 \bmod 2021$

$$= 3,5697228047878728461382284266697e+2293 \bmod 2021$$
$$= 111 = o$$

Dekripsi Kelima :

$$Ci = 1604$$
$$Pi = Cid \bmod n$$
$$= 1604773 \bmod 2021$$
$$= 4,1973320925821901822743614620265e+2477 \bmod 2021$$
$$= 110 = n$$

Dekripsi Keenam :

$$Ci = 1156$$
$$Pi = Cid \bmod n$$
$$= 1156773 \bmod 2021$$
$$= 4,6388010556934147413077372755344e+2367 \bmod 2021$$
$$= 101 = e$$

Dekripsi Ketujuh :

$$Ci = 1869$$
$$Pi = Cid \bmod n$$
$$= 1869773 \bmod 2021$$
$$= 8,9947679931588830717146802446832e+2528 \bmod 2021$$
$$= 115 = s$$

Dekripsi Kedelapan :

$$Ci = 546$$
$$Pi = Cid \bmod n$$
$$= 546773 \bmod 2021$$
$$= 7,078036710012699499463325593806e+2115 \bmod 2021$$
$$= 105 = i$$

Dekripsi Kesembilan :

$$Ci = 102$$
$$Pi = Cid \bmod n$$
$$= 102773 \bmod 2021$$
$$= 4,4456244464174792567180261019521e+1552 \bmod 2021$$
$$= 97 = a$$

Dekripsi Kesepuluh :

$$Ci = 1790$$
$$Pi = Cid \bmod n$$
$$= 1790773 \bmod 2021$$
$$= 7,1731821553698319681336017738364e+1341 \bmod 2021$$
$$32 = [\text{space}]$$

Dekripsi Kesebelas :

$$Ci = 106$$
$$Pi = Cid \bmod n$$
$$= 106773 \bmod 2021$$
$$= 3,6427876007855229645751636756921e+1565 \bmod 2021$$
$$= 72 = H$$

Dekripsi Keduabelas :

$$Ci = 1156$$
$$Pi = Cid \bmod n$$
$$= 1156773 \bmod 2021$$

$$= 4,6388010556934147413077372755344e+2367 \bmod 2021$$
$$= 101 = e$$

Dekripsi Ketigabelas :

$$C_i = 507$$
$$P_i = C_i \bmod n$$
$$= 507773 \bmod 2021$$
$$= 9,3573433346090832334487836909454e+2090 \bmod 2021$$
$$= 98 = b$$

Dekripsi Keempatbelas :

$$C_i = 102$$
$$P_i = C_i \bmod n$$
$$= 102773 \bmod 2021$$
$$= 4,4456244464174792567180261019521e+1552 \bmod 2021$$
$$= 97 = a$$

Dekripsi Kelimabelas :

$$C_i = 270$$
$$P_i = C_i \bmod n$$
$$= 270773 \bmod 2021$$
$$= 2,7809276801454902544239942650756e+1879 \bmod 2021$$
$$= 116 = t$$

The results of the decryption process above produced a text which means "Great Indonesia". From the results of manual calculations and those produced by the system, there are no significant differences. Therefore RSA and LSB cryptography on the red pixel was successfully processed by the system.

V. ACKNOWLEDGMENT

From the encryption, decryption, embedded, and extraction processes that are running well, the researchers concluded that:

1. The combination of cryptographic RSA algorithm techniques and LSB method steganography techniques is able to maintain the confidentiality of messages because the presence of messages is difficult to know thanks to the LSB method that inserts messages into image images. And also the difficulty of knowing the original message without having an RSA key pair.
2. Insertion of messages successfully on image images with *.jpg format and at 3888x2592 resolution.

VI. ADVICE

In this research, it still has shortcomings under weaknesses that can be developed for subsequent research. Suggestions for future research are as follows:

1. Does not show RSA encryption results in the form of ciphertext making it difficult to compare calculations manually.
2. Researchers can develop programs using Android bass programming
3. Prime numbers still need to be entered manually, in the next research, random primers can be added.

2
REFERENCES

- [1] Rakhmat, B., & Fairuzabadi, M. (2010). Steganografi Menggunakan Metode Least Significant Bit Dengan Kombinasi Algoritma Kriptografi Vigenère Dan Rc4. *Jurnal Dinamika Informatika*, 5(2), 1-17.
- [2] Alvianto, A. R., & Darmaji, D. (2015). Pengaman Pengiriman Pesan Via SMS dengan Algoritma RSA Berbasis Android. *Jurnal Sains dan Seni ITS*, 4(1), A1-A6.
- [3] Arif, M. H., & Fanani, A. Z. (2016). Kriptografi Hill Cipher dan Least Significant Bit untuk Keamanan Pesan pada Citra. *CSRID (Computer Science Research and Its Development Journal)*, 8(1), 60-72.
- [4] Akbar, M. B., & Haryanto, E. V. (2018). Aplikasi Steganografi dengan Menggunakan Metode F5. *JUSITI: Jurnal Sistem Informasi dan Teknologi Informasi*, 4(2), 165-176.
- [5] Rambe, M. R., Haryanto, E. V., & Setiawan, A. (2018). Aplikasi Pengamanan Data dan Disisipkan Pada Gambar dengan Algoritma RSA Dan Modified LSB Berbasis Android. *Konferensi Nasional Sistem Informasi (KNSI) 2018*.
- [6] Haryanto, E. V., & Utama, U. P. (2012). Jaringan Komputer. Penerbit Andi.
- [7] Haryanto, E. V. (2015). Penerapan Metode Adaptif Dalam Penyembunyian Pesan Pada Citra. *Proceedings Konferensi Nasional Sistem dan Informatika*
- [8] Haryanto, E. V. (2015). Perbandingan Metode Robinson 5 Level Dan Prewit Dalam Mendeteksi Tepi Citra Digital. *Proceedings Konferensi Nasional Sistem dan Informatika (KNS&I)*. (KNS&I).

ORIGINALITY REPORT

19%

SIMILARITY INDEX

15%

INTERNET SOURCES

3%

PUBLICATIONS

13%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to Udayana University

Student Paper

7%

2

adi-journal.org

Internet Source

6%

3

www.scilit.net

Internet Source

2%

4

Sayuti Rahman, Indah Triana, Sumi Khairani, Amru Yasir, Siti Sundari. "RSA KEY DEVELOPMENT USING FINGERPRINT IMAGE ON TEXT MESSAGE", Journal of Physics: Conference Series, 2017

Publication

1%

5

209.61.191.11

Internet Source

1%

6

Submitted to President University

Student Paper

1%

7

Nur Atikah, Mutia Rizky Ashila, De Rosal Ignatius Moses Setiadi, Eko Hari Rachmawanto, Christy Atika Sari. "AES-RC4 Encryption

1%

Technique to Improve File Security", 2019
Fourth International Conference on Informatics
and Computing (ICIC), 2019

Publication

8

Tonni Limbong, Janner Simarmata, ARS
Tambunan, Parulian Siagian et al. "The
implementation of computer based instruction
model on Gost Algorithm Cryptography
Learning", IOP Conference Series: Materials
Science and Engineering, 2018

1%

Publication

Exclude quotes Off

Exclude bibliography Off

Exclude matches < 1%