IMPLEMENTATION OF RSA ENCRYPTION ALGORITHM ON INSTANT MESSAGING APPLICATION

Ardiansyah Sukma Wijaya[1], Dodon T. Nugrahadi[2], Muhammad Itqan Mazdadi[3], Andi Farmadi [4], Ahmad Rusadi[5]
[1,2,3,4,5]Computer Science Study Program FMIPA ULM
A. Yani Street Km 36 Banjarbaru, South Kalimantan
Email: ardiansyahsw@gmail.com

Abstrak

At this time the use of instant messaging applications is increasingly used compared to the use of SMS or other media because of its use which is more practical and faster. From the other side, the message information sent certainly requires confidentiality so that the message is not spread and known by others. For this reason mechanisms are needed, one of which is encryption to maintain message security. This research will implement the RSA (Rivest Shamir Adleman) encryption algorithm in the instant messaging application. This study uses a key length scenario of RSA 1024, 2048, 4096, and 6144 bits and a message length of 125, 250, 500, and 1000 characters implemented on 3 different devices. The results of testing on time and speed are the shorter the key used, the process will be shorter and faster.

Keywords : *RSA, Processing Time, Processing Speed.*

1. INTRODUCTION

The development of technology now has a very large, fast and rapid influence on the world of information and telecommunications technology. The emergence of a variety of applications provides a choice in improving the performance of a job, both desktop based, web based, and until now the emergence of new applications running on mobile on the Android platform system. Nowadays new technologies emerge where communication can be done without using a cable, such as by using Internet Media that is a client server on mobile android [2].

Currently one form of communication service that is growing rapidly is instant messaging. Instant messaging is one of the communication services that can make users to communicate in the form of short messages in real time through the internet network. Even the use of instant messaging has outperformed the use of SMS services following the current smartphone development. According to data from Informa research company, in 2012 the number of messages sent through instant messaging services reached 19 billion, exceeding the number of messages sent via SMS, which amounted to 17.6 billion[3].

This instant messaging uses the internet network, so maintaining the confidentiality of messages is important. One way to maintain data confidentiality is to use cryptography. The basic principle of cryptography is the process of encryption and decryption [4]. One cryptographic algorithm that is widely used is RSA. The RSA algorithm with a 768-bit key could be solved or factored in 2015 and those who managed to break the 768-bit key speculated that a 1024-bit RSA key

could be broken before 2020 [1]. This statement gives hope that RSA keys over 1024 bit are still being debated to be solved.

Based on this, the research will implement the RSA encryption algorithm inapplications instant messaging and simulate sending messages with different scenarios hardware (smartphone devices), variations in the number of texts or strings (message length), and variations in length key (encryption keys). and decryption). By using the test parameters used are the time and speed required during the encryption and decryption process message

2. RESEARCH METHOD

The RSA algorithm was created by 3 researchers from MIT (Massachussets Institute of Technology) in 1976. The three researchers are Ron Riverst, Adi Shamir, and Leonard Adleman. The RSA algorithm is used to generate 2 keys, namely the public key and private key. Public keys are two variable numbers (e, n) that are used to encrypt data. While the private key are two variable numbers (d, n) that are used to decrypt data [4].

The methodology and steps (procedures) used in this study are:

a. Study of Literature is to study data and writing material based on theoretical data relating to the RSA encryption algorithm
b. In the design to obtain data, scenarios used to test encryption are the use of different smartphone devices, variations in the number of texts or strings (message length) and different key lengths
c. The development of instant messaging applications in this study uses Android Studio.
d. RSA algorithm implementation in the application created
e. Testing based on a scenario that has been designed
f. Analyze test data obtained from the testing process. The analysis uses different test scenarios, namely
    1. The length of the RSA key used.
    2. The length of the input or message sent.
    3. The use of different smartphone devices

3. RESULTS AND ANALYSIS

3.1 Aplication Design

In the development and manufacturing of this instant messaging application using the Java programming language that is on the Android Studio application and using the realtime database provided by Google, which is Firebase as a place for storing data and managing accounts and user messages. Then the SQLite database is used to store message data on the user's smartphone's local storage device. This application is made in such a way that it can work like a messaging application in general that can send and receive messages from one user to another with the implementation of the RSA algorithm. Here is a display of this application :

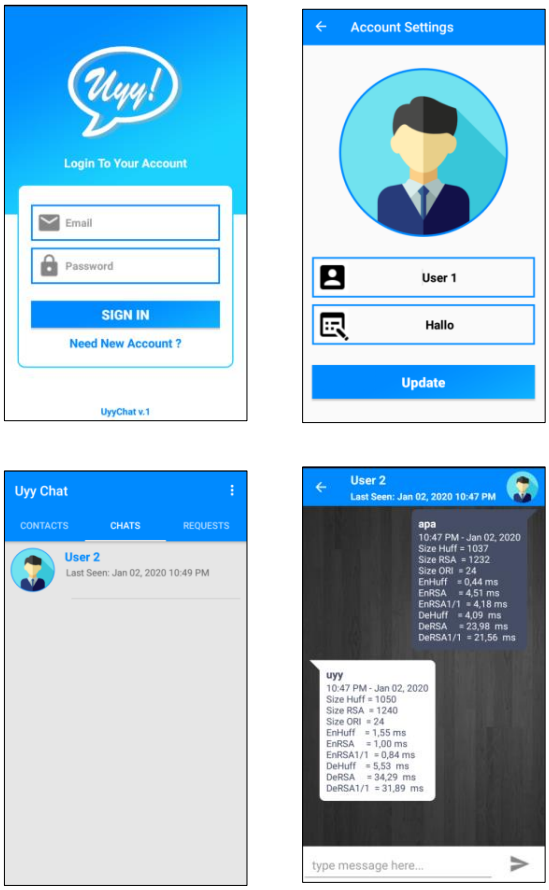

Figure 1. Interface of instant messaging application

3.2 Simulation Parameters

Table 1. Simulation Parameters

Encryption Algorithm	RSA
Key Length	1024, 2048, 4096, and 6144 bit
Message Text Length	125, 250, 500, and 1000 characters
Type of Device Smartphone	Xiaomi Redmi 4X, Xiaomi Redmi 3 Pro, and Realme 5 Pro
Test Parameter	Time and speed
Number of Trial	10 times per scenario (*average*)

3.3 Time Testing Result

Data taken is the total processing time of messages that are encrypted and decrypted. This scenario will discuss the results of time testing based on different devices. The devices used are Redmi 4X, Redmi 3 pro, and Realme 5 Pro with key lengths of 1024, 2048, 4096 and 6144 bits with message lengths of 125, 250, 500, and 1000 characters.

a. Redmi 4X

Table 2. Time test results for the Redmi 4X scenario

Key Length	Input Length	Total RSA Time

(bit)	(character)	(ms)
1024	125	37.61
	250	72.43
	500	128.76
	1000	228.52
2048	125	147.89
	250	154.53
	500	246.09
	1000	474.16
4096	125	727.26
	250	719.58
	500	705.07
	1000	1410.20
6144	125	2511.29
	250	2500.79
	500	2494.97
	1000	5018.88

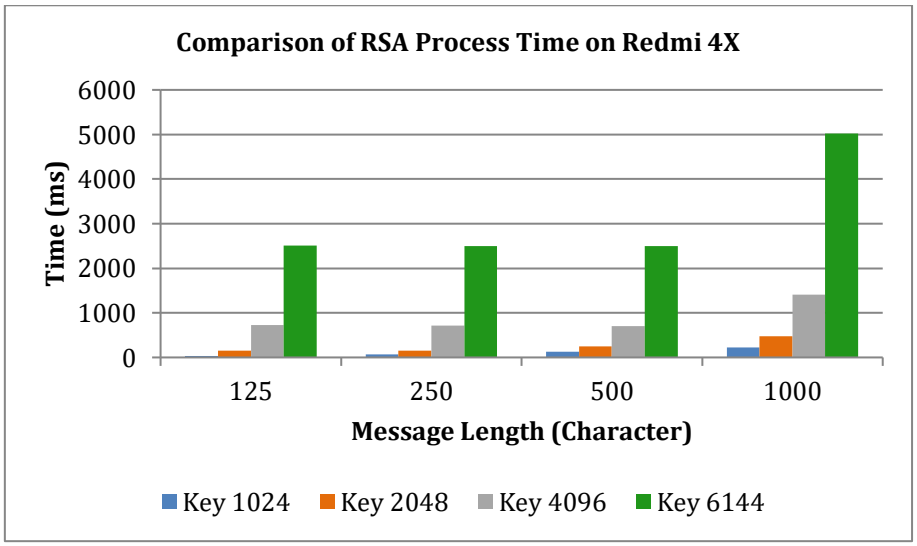

Figure 2. Process time comparison graph using a Redmi 4X device

b. Redmi 3 Pro

Table 3. Time test results for the Redme 3 Pro scenario

Key Length (bit)	Input Length (character)	Total RSA Time (ms)
1024	125	26.17
	250	46.98
	500	83.48
	1000	163.21
	125	86.50

2048	250	82.97
	500	174.07
	1000	330.89
4096	125	491.47
	250	480.21
	500	487.03
	1000	913.89
6144	125	1577.46
	250	1560.48
	500	1562.91
	1000	3077.13

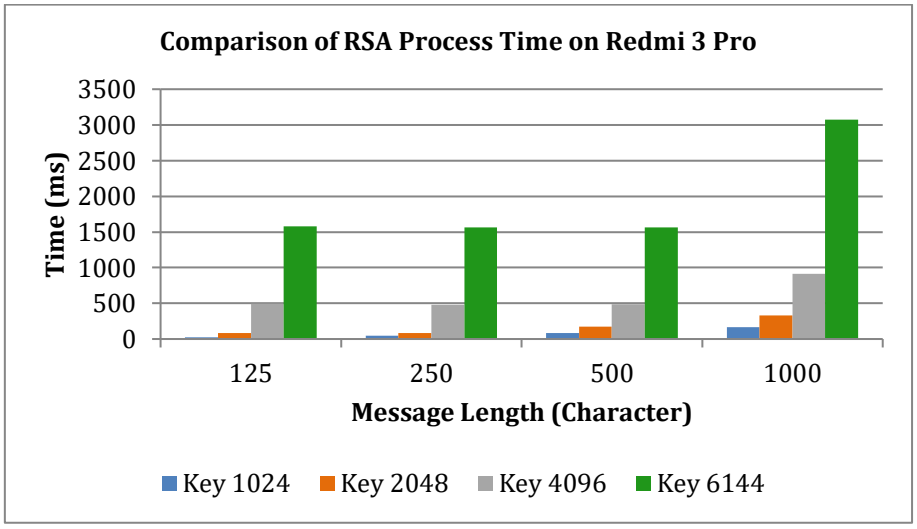

Figure 3. Process time comparison graph using a Redmi 3 Pro device

c. Realme 5 Pro

Table 4. Time test results for the Realme 5 Pro scenario

Key Length (bit)	Input Length (character)	Total RSA Time (ms)
1024	125	29.52
	250	43.77
	500	56.66
	1000	88.44
2048	125	44.88
	250	46.49
	500	71.67
	1000	130.78
4096	125	147.25
	250	161.97
	500	141.29
	1000	272.46

6144	125	395.61
	250	397.48
	500	393.74
	1000	767.96

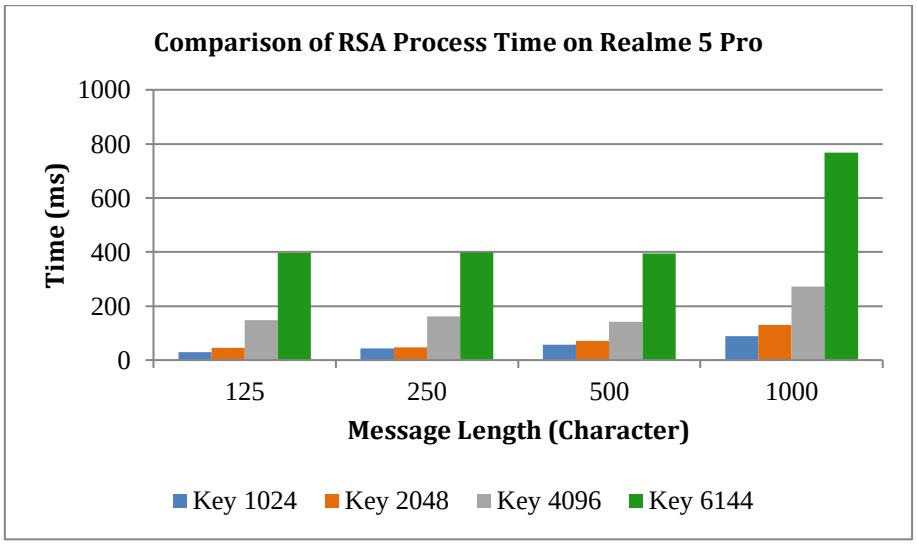

Figure 4. Process time comparison graph using a Realme 5 Pro device

And based on the key length scenario used, the comparison results are:

a. 1024 Bit Key

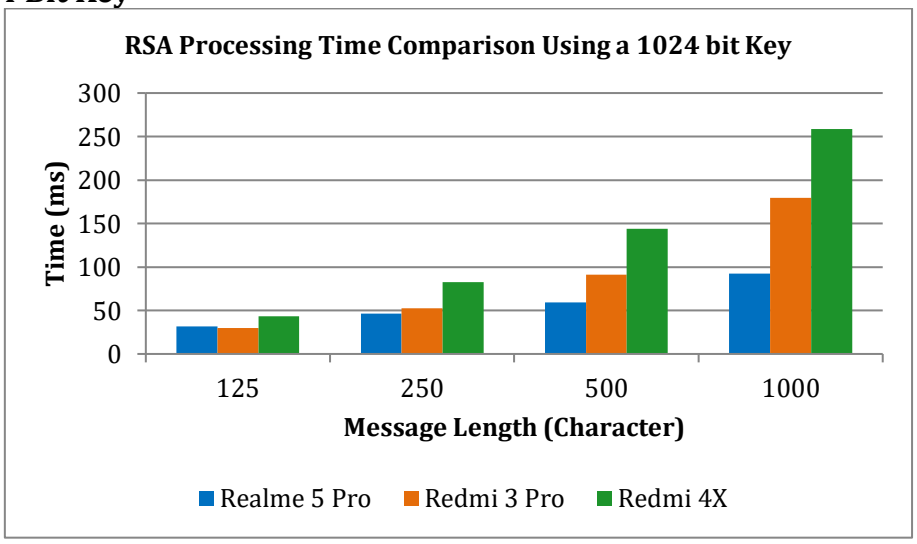

Figure 5. Process time comparison graph using a 1024 bit key

b. 2048 Bit Key

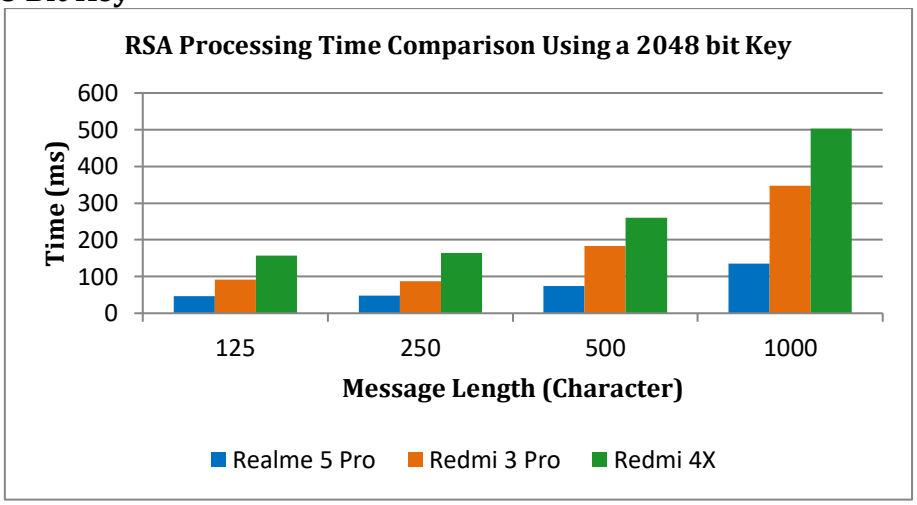

Figure 6. Process time comparison graph using a 2048 bit key

c. 4096 Bit Key

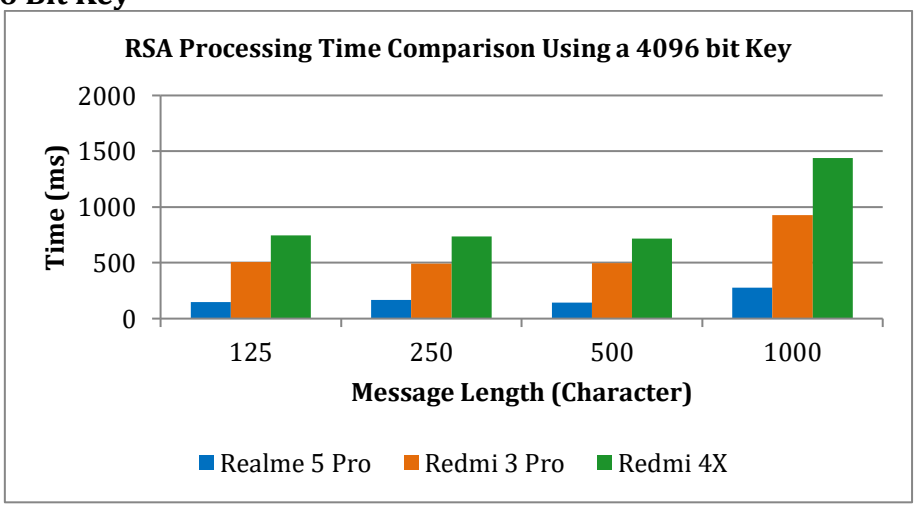

Figure 7. Process time comparison graph using a 4096 bit key

d. 6144 Bit Key

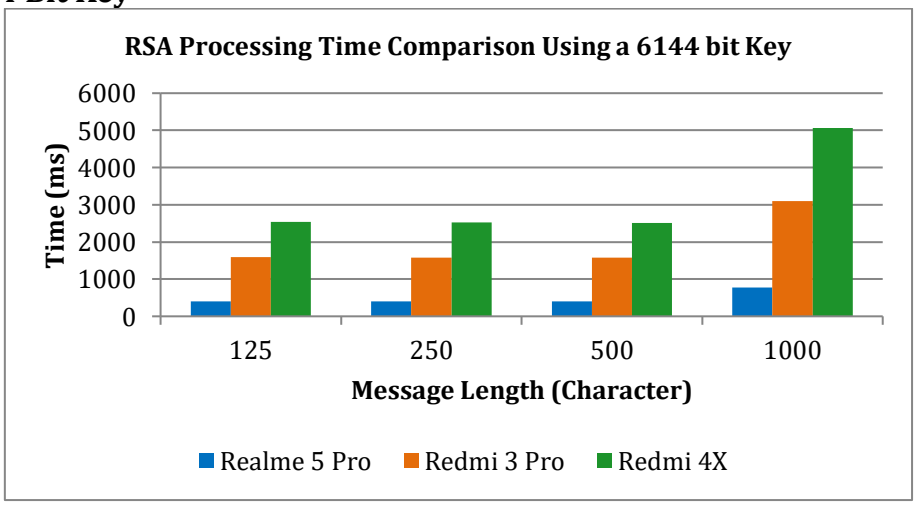

Figure 8. Process time comparison graph using a 6144 bit key

Based on the results of the test scenarios and the average calculation performed and the use of different devices, it can be concluded that the processing time required continues to increase with the increase in RSA key length and message length used. Based on the key scenario of RSA 1024 bits, RSA 2048, 4096, and 6144 bits have an increased processing time which is directly proportional to the key. With the fastest time of 88.44 ms for a maximum of 1000 characters input using a 1024 bit RSA key and Realme 5 Pro device and the longest time recorded 5018.88 ms for a maximum of 1000 characters input using a 6144 bit RSA key and a 4X Redmi device. In terms of security for 1024-bit keys it is certainly less secure if used for now, when compared to the use of larger keys such as 2048 bits it will be safer if used for the future because the greater the length of the RSA encryption the key length will be more difficult to break because in RSA encryption the longer the key used is the more difficult to break.

3.4 Speed Test Results

Speed test results are also implemented on different devices. The devices used are Redmi 4X, Redmi 3 pro, and Realme 5 Pro with key lengths of 1024, 2048, 4096, and 6144 bits with message lengths of 125, 250, 500, and 1000 characters.

a. Redmi 4X

Table 5. Speed test results for Redmi 4X scenario

Key Length (bit)	Input Length (character)	Total RSA Speed (ms/s)
1024	125	26.59
	250	27.61
	500	31.07
	1000	35.01
2048	125	6.76
	250	12.94
	500	16.25
	1000	16.87
4096	125	1.38
	250	2.78
	500	5.67
	1000	5.67
6144	125	0.40
	250	0.80
	500	1.60
	1000	1.59

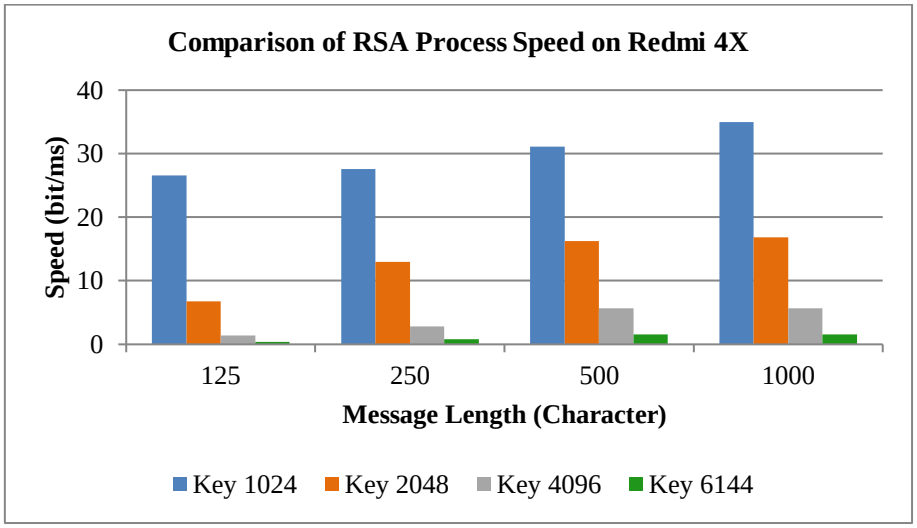

Figure 9. Process time comparison graph using a Redmi 4X device

b. Redmi 3 Pro

Table 6. Speed test results for Redmi 3 Pro scenario

Key Length (bit)	Input Length (character)	Total RSA Speed (ms/s)
1024	125	38.21
	250	42.57
	500	47.92
	1000	49.02
2048	125	11.56
	250	24.10
	500	22.98
	1000	24.18
4096	125	2.03
	250	4.16
	500	8.21
	1000	8.75
6144	125	0.63
	250	1.28
	500	2.56
	1000	2.60

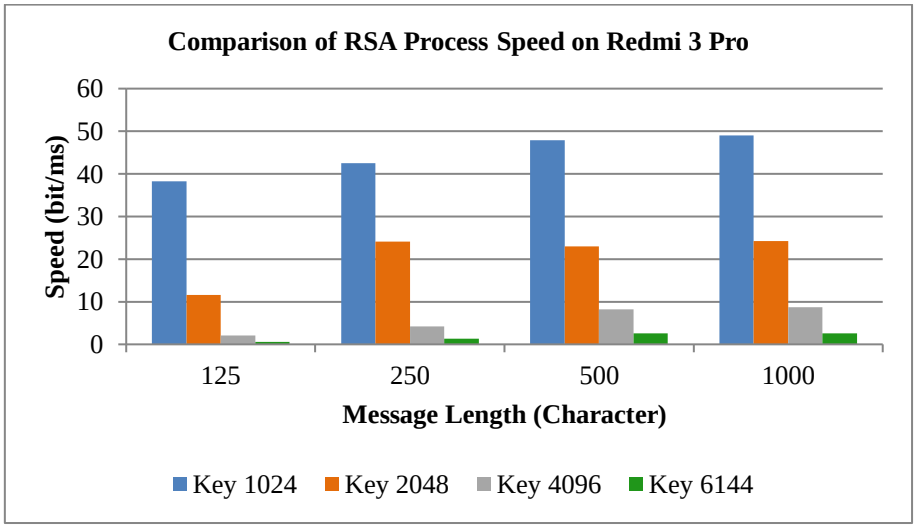

Figure 10. Process time comparison graph using a Redmi 3 Pro device

c. Realme 5 Pro

Table 7. Speed test results for Realme 5 Pro scenario

Key Length (bit)	Input Length (character)	Total RSA Speed (ms/s)
1024	125	33.88
	250	45.69
	500	70.60
	1000	90.46
2048	125	22.28
	250	43.02
	500	55.81
	1000	61.17
4096	125	6.79
	250	12.35
	500	28.31
	1000	29.36
6144	125	2.53
	250	5.03
	500	10.16
	1000	10.42

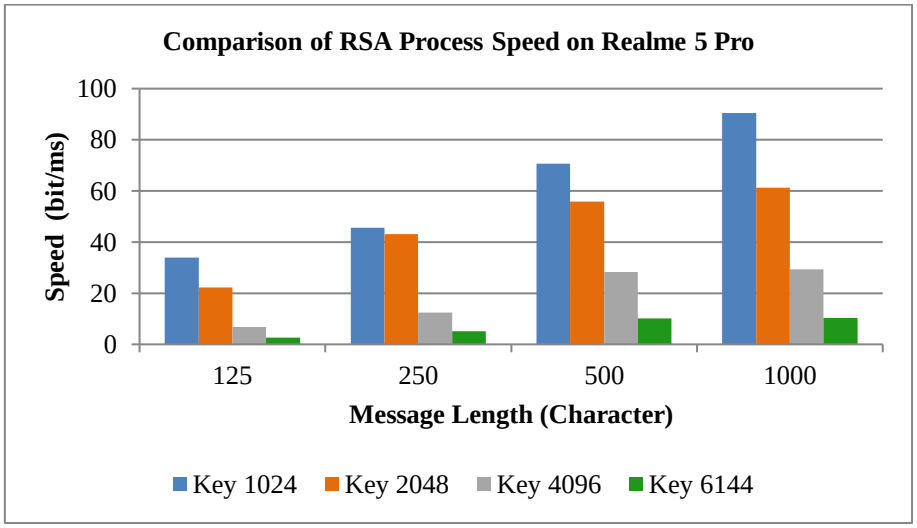

Figure 11. Process time comparison graph using a Realme 5 Pro device

Based on the test scenario and the calculation of the speed of the process carried out as well as the use of different devices the results are obtained that the required process speed continues to increase with the increasing length of the message used. However, it decreases with increasing length of the RSA key used. With the fastest speed of 90.46 bits / s with a maximum input of 1000 characters on the use of 1024-bit keys and Realme 5 Pro devices.

4. CONCLUSION

Based on the results of the study, the conclusion of the discussion is that the smaller the RSA key used, the shorter the processing time required and vice versa, the greater the RSA key used, the longer the processing time needed. However, this is inversely proportional when viewed in terms of security because the smaller the RSA key used, the security on the message will be reduced. At the process speed the greater the RSA key, the speed will also be slower and the smaller the RSA key, the speed will also be faster. Besides the device used also affects the higher the specifications of the device used, the processing time will be shorter and the process speed will also be faster.

REFERENCES

[1] Childs, Lindsay N. 2019. **Cryptology and Error Correction: An Algebraic Introduction and Real-World Application**. Department of Mathematics and Statistics University at Albany New York.

[2] Laurentinus. 2017. **Implementasi Kriptografi Dan Kompresi Sms Menggunakan Algoritma Rc6 Dan Algoritma Huffman Berbasis Android**. TI STMIK Atma Luhur.

[3] Nugroho, Andreas Dwi, Rinaldi Munir. 2015. **Comparison Of Lossless Data Compression Algorithms For Text Data**. Bandung Institute of Technology.

[4] Zainuddin, Muhammad Arif, and Danang Iskandar Mulyana. 2016. **Penerapan Algoritma RSA untuk Keamanan Pesan Instant pada Perangkat Android**. STIKOM CKI Informatic Engineering.