Performance analysis of 256-bit aes encryption algorithm on android smartphone

by Samsuryadi Samsuryadi

Submission date: 13-Apr-2023 03:58PM (UTC+0700)
Submission ID: 2063331365
File name: ARTIKEL_-_Performance_analysis_of_256-bit_aes_encryption.pdf (609.3K)
Word count: 3239
Character count: 17205

Journal of Physics: Conference Series

Performance Analysis of 256-bit AES Encryption Algorithm on Android Smartphone

Performance Analysis of 256-bit AES Encryption Algorithm on Android Smartphone

Nur Rachmat[1], Samsuryadi*[2]

[1]Student in Master of Informatics, Universitas Sriwijaya, STMIK Global Informatika MDP, Palembang
[2]Lecturer, Faculty of Computer Science, Universitas Sriwijaya, Palembang

E-mail: rachmat.nur91@mdp.ac.id[1], syamsuryadi@unsri.ac.id[2]

Abstract. Encryption Techniques are one of the most important and very useful things to secure and protect the messages against such threads. To develop a highly secure android application, the application developer needs to take consideration various things furthermore the security after performance. Because of Android smartphone have limited resources, therefore effort to improve the performance of applications for the Android platform need to be done. The questions about which algorithms are performing better on Android are frequently arise in various developer forum such us Stack Overflow and Quora. To answer the questions this paper will show performance comparison of the most common safe encryption algorithm specifically Rijndael, Serpent and Twofish in order to establish which one of these is the most optimum to be implemented in Android smartphone. The test is carried out over devices which measure the performance and computational cost from the CPU and memory consumptions of the device and execution time when they run each algorithm on 256-bit key length. The test result shows that the Serpent has better encryption and decryption performance than Rijndael and Twofish. Serpent has the faster time for encryption and decryption. While Serpent and Twofish surpass Rijndael on efficiency use of memory for encryption and decryption. However, the most efficient percentage of CPU usage is done by Rijndael.

1. Introduction

Internet usage is currently increasing rapidly. This causes an increase in the need for data and information security whose distribution is needed at all times. Encryption techniques are one of the most important and very useful things to secure messages. One of the most widely used encryption techniques is [1]. The Advanced Encryption Standard (AES) is a Federal Information Processing Standard (FIPS) which has been declared after the competition standard encryption algorithm held by National Standards and Technology in 2011.

To develop a high secure android application, the application developer needs to take consideration various things furthermore the security after performance. Cryptographic algorithm such as AES should be injected on Android application particularly message-based application to secure its content. Because of Android smartphone have limited resources, therefore effort to improve the performance of applications for the Android platform need to be done. To answer this question, it is necessary to do a performance comparison between some general cryptographic algorithms that are of good speed and are popularly used.

In [2] study performance comparison the Rijndael, Serpent, and Twofish algorithms which compare benchmarks (MB/s), time (ms), memory usage (KB), and percentage of processor usage (%) on Android

smartphone. The results of this test show the performance of the Twofish algorithm is superior to other algorithms based on benchmark testing. While the results of time testing show Serpent faster than other algorithms. The Rijndael algorithm is superior in testing lower CPU usage. Whereas the results of memory usage testing show that the performance of the Twofish algorithm is superior. While the [3] only compares time (ms), memory usage (KB), and the percentage of processor usage (%). However, this study concludes that the Serpent algorithm has the best performance in the process of encryption and decryption on smartphones compared to Twofish and Rijndael algorithms. This means that implements on smartphones the performance of the Rijndael algorithm is not necessarily better than other AES candidate algorithms. Unfortunately study [2,3] conducted performance comparison of Rijndael, Serpent, and Twofish algorithms on 128 bit key length.

Based on the literature study that has been carried out, this study continues [2,3] who did a comparison of the performance applied to smartphones on 128 bit key, but this study will compare the three major AES candidate algorithms, specially Rijndael, Serpant and Twofish on 256 bit key length. This research is important to find out which algorithm has the best performance in its application to smartphones, so that it can help application developers in designing applications with high security and efficient resources.

2. Literature Review

2.1. Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES) is a Federal Information Processing Standard (FIPS) which was declared after the competition for encryption algorithms held by National Standards and Technology in 2011. AES is a very high security algorithm [1]. Some algorithms were selected as candidates in the top 5 in a row after Rijndael are Serpent, Twofish, RC6, and MARS algorithms [4]. AES is proven immune to conventional attacks (linear and differential attacks) that use statistics to crack passwords. AES has the expected cipher properties, which are: resistant to known password analysis, flexible to use in various hardware and software, good for hash functions, suitable for devices that require fast key agility, and suitable for stream ciphers.

2.1.1. Rijndael

Rijndael is a symmetric key algorithm that was selected as AES. This algorithm supports cryptography with a key length of 128 bits up to 256 bits with step 32 bits. The block size can be chosen independently and each block is encrypted in a certain number of turns. Rijndael are grouped according to the length of key. Each group has a different number of rounds for each stage [2].

Table 1. AES Groups

AES Group	Key Length	Number of Rounds
AES-128	128 bits	10
AES-192	192 bits	12
AES-256	256 bits	14

The general stages of Rijndael are as follows.

- AddRoundKey. This process is an initial round that performs XOR between the plaintext and the chipper key.
- 9, 11 or 13 round includes SubBytes, ShiftRows, MixColumns, AddRoundKey.
- The last process is the last round or the final round (making 10, 12 or 14 rounds in total) includes SubBytes, ShiftRows, and AddRoundKey

2.1.2. Serpent

Serpent Algorithm is a chipper block algorithm with 32 rounds of substitution permutation (SP) network that operates on four 32-bit words, which means the block size is 128 bits. For internal computing, all values are represented in little-indian, where the first word is least-significant word, and the last word is the most-significant word. Externally, each block is written as plain hexadecimal 128 bits [2,3].

Serpent encrypts plaintext (P) 128 bits to 128-bit ciphertext (C) in 32 rounds with control of 33 128-bit K0, ..., K32 sub keys. The key length is the same as other AES candidates, namely 128, 192 and 256 bits [3].

2.1.3. Twofish

Twofish is one of the AES candidates. Unlike the Rijndael which grouped according to the key length and number of turns for each key length, Twofish is done with 16 rounds [2]. Twofish is a 128-bit block cipher that accepts keys with a flexible length of up to 256-bit, similar to Rijndael and Serpent. The cipher of the Twofish algorithm uses quite a number of methods in its implementation. The methods include Feistel Network, S-Box, MDS Matrix, Pseudo-Hadamard transformation, whitening, and key schedule.

2.2. Literature Study

Study that analyzes several algorithms in cryptography are [2,3,5–7]. Study [2] and [3] perform a performance comparison of the rijndael, serpent, and twofish algorithms. Study [5] compared the performance of several symmetric key algorithms, such as DES, 3DES, RC4, blowfish and AES (Rijndael). Study [2] compares benchmarks (MB/s), time (ms), memory usage (KB), and percentage of processor usage (%) on mobile devices. While study [3] compares time (ms), memory usage (KB), and percentage of processor usage (%). Even as [5] uses encryption time, decryption time, throughput, CPU process time, and memory utilization for performance comparison parameters.Whilst study [7] combines the RSA algorithm with the Diffie-Hellman algorithm to compare their performance with AES (Rijndael), DES, 3DES, RC2, and Blowfish algorithms. The comparison parameters in this study are CPU time, memory, and battery power. The latest study take performance comparisons of AES and blowfish algorithms, unfortunately the measured parameter is only time (ms).

The results of the study [5] showed that the AES and Blowfish performed best, however AES was superior to Blowfish in more efficient memory usage. But evidently the results of research [6] show that Blowfish has better speed performance than AES based on data length and key length. Then Madal showed that the combination of RSA algorithm and Diffie-Hellman algorithm has superior performance compared to AES (Rijndael), DES, 3DES, RC2, and Blowfish.

The test results on study [2] show that the performance of the Twofish algorithm is superior to other algorithms based on benchmark tests. While time testing results show Serpent faster than other algorithms. The Rijndael algorithm which is the standard AES algorithm is superior in testing the percentage of lower CPU usage. Whereas the results of memory usage tests show that the performance of the Twofish algorithm is superior. At the same time results on study [3] concluded that the serpent algorithm has the best performance in the process of encryption and decryption on smartphones compared to Twofish and Rijndael. Unfortunately, this research has only been done on 128-bit key size.

Tests on mobile devices have been carried out by [2,3,5,6,8]. Study [3,5,8] developed a SMS application for Android device to test the algorithms. While [2,6] build an Android application to test the algorithm for text data only. There is no study that build a chatting application for testing purpose, hence this study will build a chat application on Android device to test the performance of Rijndael, Serpent and Twofish algorithm on 256-bit key size.

3. Design and Test

3.1. Performance Analysis

This study conducted a series of tests to analyze the performance of the Rijndael, Twofish and Serpent algorithm. The performance parameters analyzed in this study are as follows:

- Time (ms)
- Memory usage (kb)
- CPU usage (%)

The results of this test will also compared over the latest study result [3] on difference key size (128 bit).

3.2. Test Scenario

Each algorithm will be carried out the process of testing encryption and decryption by performing several test scenarios. The testing conducted in this study uses the following test scenarios:

- Testing 1: Each device performs 20 times encryption, each device and each decryption process is done with the Rijndael, Serpent and Twofish on 256-bit key length.
- Testing 2: Each device performs 20 times decryption, each device and each decryption process is done with the Rijndael, Serpent and Twofish on 256-bit key length.

On testing process, the abnormal values may appear due to the Android process running in the background, a leaning or abnormal value will be ignored and deleted to be replaced with a value the new results from another test. Testing for abnormal value replacement is done outside the scope of 20 tests. According to Jakob Nielsen, quantitative testing with 20 tests has a 90% confidence level [9], therefore in this test each test will be repeated 20 times to reduce the margin of error [6].

3.3. Test Implementation

To test the scenario this study builds an application on Android smartphone called Crypto Chat. Crypto Chat developed and compiled in Android Studio. The compiled application should be installed on Android device to running the tests. Figure 2 shows some screenshots of the application.

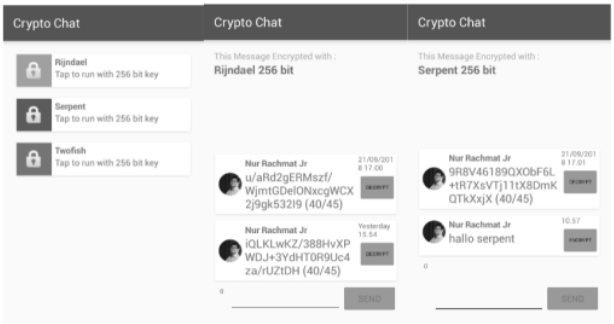

Figure 1. Crypto Chat application screenshots

This application utilizes Firebase Real-time Database technology so that the development process can be done quickly because the developer does not need a lot of effort for backend process. Real-time Database is a database system that uses real-time processing to handle workloads whose status keeps changing. Real-time Databases different with traditional databases that contain persistent data. For example, the stock market changes very quick and dynamically. Real-time processing means that transactions are processed quick enough so that the results can be returned and processed immediately [10,11].

Table 2. Devices Used in Experiments

Device	Specification	Operating System
Samsung Galaxy Grand Duos	• Dual-core 1.2 GHz Cortex-A9 • 1 GB RAM • Internal Storage 8 GB	Android 4.2.2
LG Magna	• Quad-core 1.3 GHz Cortex-A7 • 1 Gb RAM • Internal Storage 8 GB	Android 5.0.1
Xiaomi Redmi Note 5	• Octa-core 1.8 GHz Kryo 260 • 4 GB RAM • Internal Storage 64 GB	Android 7.1.2

When the test is run, the test data will be sent and stored in the Firebase Real-time Database as well as the sender, encrypted message and time. The test data were stored in analysis child, messages data in

messages child while user's data stored in user child. The devices and operating systems used to conduct experiments in this study are as follows.

4. Result and Discussion

After several test conducted over three devices on two testing scenarios, we get 180 datasets to be analyzed. There are three parameters obtained which will be analyzed such as time, memory and CPU usage. Table 3 shows the data from Firebase Real-time Database of the tests that was conducted on 256-bit Rijndael, Serpent and Twofish algorithm.

Table 3. Test Result

Test	Rijndael			Serpent			Twofish		
	Time (ms)	Memory (KB)	CPU (%)	Time (ms)	Memory (KB)	CPU (%)	Time (ms)	Memory (KB)	CPU (%)
1	1.9348	0.4336	0.0101	1.4206	0.4307	0.0146	1.7815	0.4309	0.0146
2	1.9412	0.4316	0.0133	1.4282	0.4271	0.0152	1.7732	0.4265	0.0147
3	2.0154	0.4342	0.0152	1.4241	0.4312	0.0152	1.8431	0.4315	0.0153

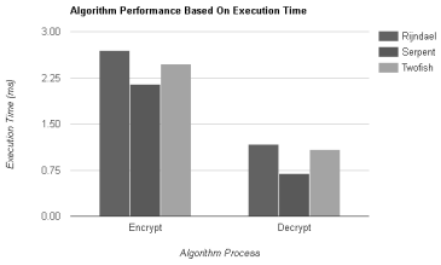

Figure 2. Execution Time Test Result

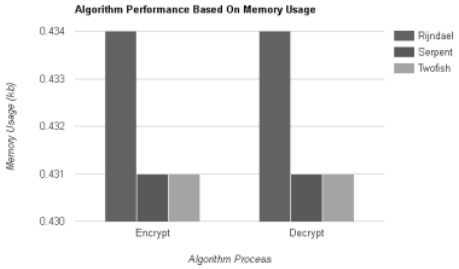

Figure 3. Memory Usage Test Result

4.1. Execution Time Result

Figure 2 shows the results of the most efficient encryption algorithm in the execution time is Serpent (2.151 ms) compared to Twofish (2.477 ms) and the Rijndael (2.696 ms). While decryption process serpent (0.69 ms) outperform Twofish (1.086 ms) and Rijndael (1.173 ms). This means that in testing the execution time of Serpent algorithm is better than other algorithms with an average usage of time 1.42 ms.

4.2. Memory Usage

The most efficient encryption and decryption process in memory usage is Serpent and Twofish (0.431 kb) outperform Rijndael with 0.434 kb memory usage. Figure 3 shows of Serpent and Twofish algorithm surpass Rijndael with 0.003 kb memory usage.

4.3. CPU Usage

For CPU usage, Rijndael outperform Serpent and Twofish algorithm. Rijndael outplay 0.009% CPU usage against Twofish (0.015%) and Serpent (0.013%) in the encryption process. While the Rijndael (0.011%) superior from Serpent (0.014%) and Twofish (0.016%) on decryption process. Serpent algorithm decryption process is able to outplay Twofish in the race for second place.

5. Conclusion

Based on the design, implementation and test results the following conclusions are obtained:

- Serpent has good performance on execution time while encryption process (2.6962 ms) and decryption process (0.6897 ms) with average execution time is 1.42 ms.
- Whilst Serpent and Twofish outperform Rijndael algorithm in memory consumption by 0.43 kb while encryption and decryption process.

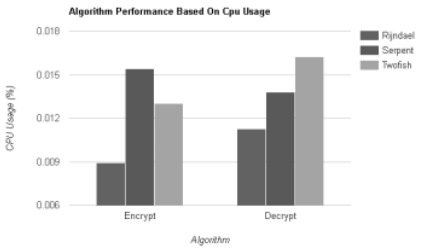

Figure 4. CPU Usage Test Result

- CPU usage percentage Rijndael was superior against Serpent and Twofish with 0.009% on encryption and 0.011% on decryption process while the average of CPU usage is just 0.013 %.

The result of this study conclude Serpent has a great performance in encryption and decryption process stack up Rijndael and Twofish on Android smartphone over 256-bit key size. This study yields similar conclusions to research [2] and [3], which means performance of the Serpent is not affected by the difference of key lengths. Serpent performance outperform Rijndael and Twofish on execution time and memory consumptions. Therefore, Android application developers not to be hesitate to choose a larger key size to make the safer system.

Several things can be done for further research by add few more devices as additional experiments. Subsequent research can also test from the point of view of developing applications used in experiments, such as applications developed natively android and ios, and also hybrid.

References

[1] Kumar P dan Rana S B 2016 Development of modified AES algorithm for data security *Opt. - Int. J. Light Electron Opt.* **127** 2341–5

[2] Montoya B. A O, Munoz G. M A dan Kofuji S T 2013 Performance analysis of encryption algorithms on mobile devices *2013 47th International Carnahan Conference on Security Technology (ICCST)* (IEEE) hal 1–6

[3] Farisi A 2018 Analisis Kinerja Algoritma Kriptografi Kandidat Advanced Encryption Standard (AES) pada Smartphone *J. Tek. Inform. DAN Sist. Inf.* **4** 199–208

[4] Bernstein D J 2015 Crypto competitions: AES: the Advanced Encryption Standard

[5] Karale S N, Pendke K dan Dahiwale P 2015 The survey of various techniques & algorithms for SMS security *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)* (IEEE) hal 1–6

[6] Rahman M T, Pinandito A dan Pramukantoro E S 2017 Perbandingan Performansi Algoritme Kriptografi Advanced Encryption Standard (AES) dan Blowfish pada Text di Platform Android **1** 1551–9

[7] Mandal B K, Bhattacharyya D dan Bandyopadhyay S K 2013 Designing and performance analysis of a proposed symmetric cryptography algorithm *Proceedings - 2013 International Conference on Communication Systems and Network Technologies, CSNT 2013* (IEEE) hal 453–61

[8] Azaim M H, Sudiharto D W dan Jadied E M 2016 Design and implementation of encrypted SMS on Android smartphone combining ECDSA - ECDH and AES *2016 Asia Pacific Conference on Multimedia and Broadcasting (APMediaCast)* (IEEE) hal 18–23

[9] Jakob Nielsen 2006 Quantitative Studies: How Many Users to Test?

[10] Nurmaini S, Malik R F, Stiawan D, Firdaus, Saparudin dan Tutuko B 2017 Information Framework of Pervasive Real Time Monitoring System: Case of Peat Land Forest Fires and Air Quality in South Sumatera, Indonesia *IOP Conf. Ser. Mater. Sci. Eng.* **190** 012029

[11] Rachmat N, Octaria O, Tarigan D M dan Samsuryadi 2016 Sistem Pemanggilan Antrian Menggunakan Websocket *Annu. Res. Semin.* **2** 445–8

Performance analysis of 256-bit aes encryption algorithm on android smartphone

ORIGINALITY REPORT

90%	81%	89%	15%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Nur Rachmat, Samsuryadi. "Performance Analysis of 256-bit AES Encryption Algorithm on Android Smartphone", Journal of Physics: Conference Series, 2019 Publication	76%
2	www.researchgate.net Internet Source	6%
3	Submitted to Sriwijaya University Student Paper	5%
4	J Bousquet, A Seubert, R Henry. "New finite element neutron kinetics coupled code system FENNECS/ATHLET for safety assessment of (very) Small and Micro Reactors", Journal of Physics: Conference Series, 2020 Publication	1%
5	Rosana Medeiros Moreira, Elcio Cruz de Oliveira. "Proposition of a new approach for calculating the efficiency of domestic gas	1%

cooking appliances", Journal of Physics: Conference Series, 2021
Publication

6	V I Romanenko, S P Nikonov, V G Zimin, Y Perin, R Henry, K Velkov. "SKETCH-N/ATHLET coupled calculations using the boundary conditions plugin", Journal of Physics: Conference Series, 2020 Publication	1 %
7	arhutek.blogspot.com Internet Source	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		