Performance Comparison Between AES256-Blowfish and Blowfish-AES256 Combinations

Muhammad Abdul Muin
Amikom Yogyakarta University
Yogyakarta, Indonesia
muinmuhammad@gmail.com

Muhammad Abdul Muin
STMIK Bina Patria
Magelang, Indonesia
muinmuhammad@gmail.com

Arief Setyanto
Amikom Yogyakarta University
Yogyakarta, Indonesia
arief_s@amikom.ac.id

Sudarmawan
Amikom Yogyakarta University
Yogyakarta, Indonesia
sudarmawan@amikom.ac.id

Kartika Imam Santoso
STMIK Bina Patria
Magelang, Indonesia
kartikaimams@gmail.com

***Abstract*— Security level is an essential feature of a cryptographic algorithm. Performing two well known cryptographic algorithms may improve a possibility to gain higher security level. This research implements a composite cryptosystem, which consists of AES256 and Blowfish algorithms. In combining AES256 and Blowfish, two options are available. The first option executes the AES256 followed by Blowfish (AES256-Blowfish). The second option is performing Blowfish and followed by AES256 (Blowfish-AES256). Security level in this research is measured by the required time to decrypt the ciphertext. Longer decryption time lead to a longer time to perform a brute force attack to get the original text message, therefore more secure. Experiment result shows that a composite cryptosystem on AES256-Blowfish required longer decryption time compare to the composite cryptosystem in reverse order (Blowfish-AES256)**

***Keywords*— AES256, Blowfish, combination order, security**

I. INTRODUCTION

Computer system plays important role in human life at the moment. One of the significant issues is file management. Secure file management has become an important technology along with the increasing use of computer systems [1]. The recommended file security is storing encrypted files to be inaccessible to irresponsible people [1][2]. With the rapid development of network technology, attacks over the Internet are also diverse, traditional encryption algorithms (single data encryption) is not enough to ensure information security on the internet[3]. According to [4], AES (Advanced encryption standard) is the best encryption algorithm, that was proposed by NIST. On the other hand, Blowfish is the fastest algorithms, but the security level is lower than that of AES [7].

There were a number of researches, such as[4][5][6] tried to combine several encryption algorithms in order to improve the security. They prove that the combination of algorithm shows better security level, however, no one of them observes the impact of combination order to the security performance.

According to [7]Blowfish and AES 256 are the safest and most efficient algorithm. Moreover, Blowfish can only be attacked with brute force, and takes a long time. Password security using MD5 because the algorithm is the fastest and [8]safest [9].

Based on the existing result Blowfish is faster than AES for encryption and decryption[10]. In this study, the authors tried to compare the combination of Blowfish and AES 256 algorithms with AES 256 and Blowfish for the fastest time of encryption and decryption. Differences in the order of combinations of algorithms will be observed at their safety level.

Blowfish was designed by Bruce Schneider in 1993 as an alternative algorithm for rapid encryption [11]. Blowfish is included in 64-bit Chipper block encryption with a key length of at least 32-bit to 448-bit [12]. Made for use on computers with large microprocessors (32-bit and above with large data cache) [13]. Blowfish was created by Schneier Bruce to allow someone to use encryption that is free of patents and copyright [11].

[14]In 1997, the National Institute of Standards and Technology (NIST) announced to select the DES generation. In 2001, NIST selected the Advanced Encryption Standard as a replacement for DES and 3DES. AES (Advanced Encryption standard) was developed by Vincent Rijmen, Joan Daemen in 2001. AES256 (Rijndael) is included in a type of symmetry cryptographic algorithm and cypher block [15][14]. [16]Data blocks are inserted and keys are operated in array form. Each member of the array before generating a ciphertext output is called a state. Each state will undergo a process that outline consists of four stages namely, AddRoundKey, SubBytes, ShiftRows, will be repeated as much as 14 times, while MixColumns did 13 times round or not will be done in the last stage.

Message Digest 5 (MD5) hash function is proposed by Rivest to improve the previous version MD4 and published in 1992. MD5, similar to other cryptographic hash algorithms, retrieves messages of a free size and produces fixed-size output (128 bit)[17].

This paper focuses on performance comparison between combination AES256 and Blowfish in a different order. Firstly AES256 followed by Blowfish, and secondly, Blowfish followed by AES256. We observe decryption time in order to justify the security level. Previous works attempted to combine AES256 and Blowfish[4][5][6]without

978-1-5386-5529-0/18/$31.00 ©2018 IEEE

observing the impact of the different order. Therefore, this paper focus on the observation of the impact of algorithms order.

II. Related Works

[7][10][18][19][20]comparing the security aspect of some symmetric and asymmetric algorithms. Blowfish is proven as the most secure algorithm, because of int strength against brute force attacks. While AES security come after the blowfish algorithm[10]. Blowfish algorithm is work based on entropy. According to previous studies [7], AES is considered the most secure. In term of time performance, the Blowfish algorithm takes the shortest time, both in encryption and decription[10]. However, other researchers conclude that AES[18] is the fastest encryption and decryption. In term of memory usage, the Blowfish algorithm utilizes the smallest size of memory[20][10][7].

[21]Successfully implemented Message Digest5 (MD5) improve document security and keep the message authenticity. In [22], Blowfish and RCA combined and implemented in VHDL (VHSIC (Very High Speed Integrated Circuit) Hardware Description Language)[22].

[23]Made an experiment to combine Blowfish and AES as well as twofish and AES. He used ECDH (Elliptic curve Diffie Hellman) mechanisms to generate and exchange the key. A time comparison had been observed and a combination of AES and Twofish is faster in both encryption and decryption compared to AES and Blowfish. There was no detail explanation on the security aspect of their paper.

[6][4][5]made an experiment on Comparing several algorithms combination. They reported that combined algorithms perform faster encryption and decryption than that of the original algorithm. [6][4]Comparing AES, Blowfish and combined AES and Blowfish, and reported that the fastest encryption and decryption is achieved by the combination of AES and Blowfish. [5]also reported that the combination of AES and Elgamal performs better than the original AES or Elgamal.

III. Methodology

We carried out several experiments to prove our hypothesis. The key is generated between 4-12 characters. The generated key contains lowercase, uppercase alphabets, numbers and special characters. According to [24]the minimum key length is suggested to be 8 characters, therefore a 4-12 characters key is generated in our experiments. The hypothesis of this research is that the combination order affects the security performance. The combined algorithms are AES256 and Blowfish.

In order to observe the security performance, an experimental design is proposed with three input variable (x1, x2 and x3) and two output variables (y1 and y2).

x1 = the key length, varied between 4,6,8,10, and 12 character containing alphabets in upper and lowercase, number and special characters
x2 = the length of the plain text
x3 = algorithm order
y1 = encryption time
y2 = decryption time
where security level will be estimated from the decryption time needed.

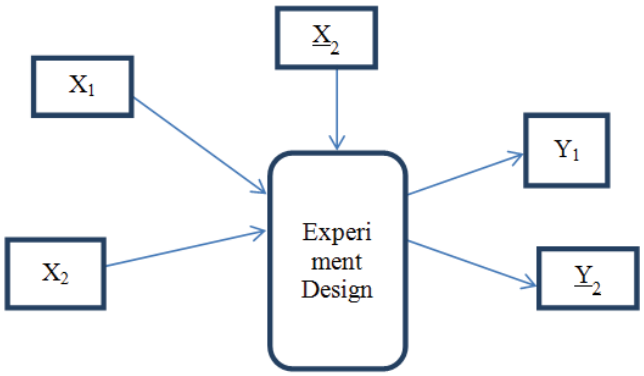

Fig. 1. Experiment design

Assumed the attacker knows the length of the key, estimated time needed for brute force attack is depended on the number of all possible key combination. The number of possible key combination for certain key length is described in the following formula [25].

$$C = I^L \tag{1}$$

Where C is all possible key combination, I am the possible characters contained by the key and L is the length of the key.

In the real brute force, the attacker examines all possible combination of the key. Therefore, the total time of performing brute force attack to get the password right in certain key length is calculated by Equation 2.

$$t_{bf} = t_{dec} \times C \tag{2}$$

Where t_{bf} is the total brute force time, t_{dec} is the average decription time for every single key, C is the key population (the number of all possible key).

Due to the huge number of possible key combination in this research, performing actual brute force need millions of seconds, therefore, it is not possible to be carried out on this research. Therefore, assuming possible key combination is a population, a random sampling is applied to get a set of the representative sample. According to Slovin's formula sample size can be calculated using the Equation 3 below.

$$n = \frac{N}{1+Ne^2} \tag{3}$$

Where n = the sample size, N the population and e is the margin of error. According to formula 1, the possible combination can be calculated. By substituting N with the C we get the number of key population.

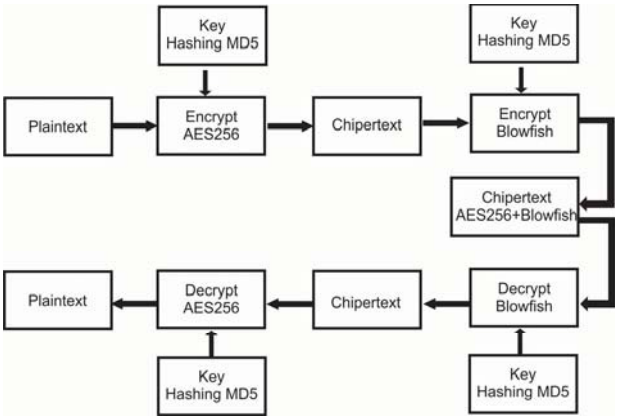

Fig. 2. Experiment Design Combination AES256 dan Blowfish Algorithm

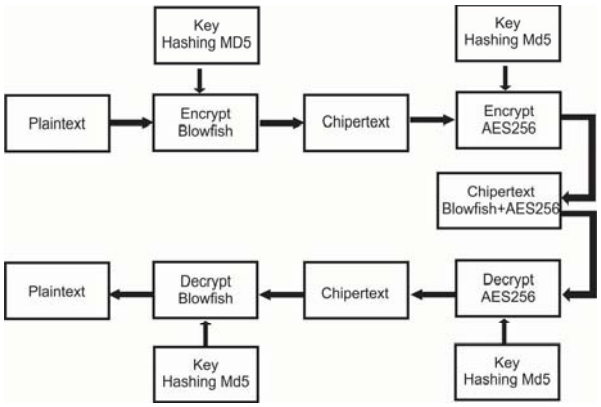

Fig. 3. Experiment Design Combination Blowfish dan AES256 Algorithm

In Fig 2 and Fig 3 are the design of the experimentdesign in order to observe the first scenario a combination of AES followed by blowfish (Fig 2) and the second scenario a combination of Blowfish followed by AES (Fig 3). The generated key is hashed by the MD5 algorithm in both experiments design.

IV. Experimental Results and Performance Analysis

A. Key sampling

The combination of AES256 and Blowfish encryption algorithms was successfully implemented using the C language. Using the slovin formula (3) taken from the possibility of a key of 4 characters with a combination of alphabets, number and special characters.

We have a sample of 4, 6, 8 10 and 12 characters key length. we use formula (1) to calculate the key population. The number of characters may contain in a single key are 128 characters consists of alphabets in lower and uppercase, number and special characters. According to slovin formula (3) for 95% certainty level, the number of the sample can be calculated using formula (3) with e = 0.05 (5%). Therefore the key population (N) for 4, 6,8,10 and 12 characters key length are listed in the table below

TABLE I. Sample For Minimum

No	Key length (Character)	N (Password Population)	n (sample)	Minimum Sample Size
1	4	128^4	$n = \frac{128^4}{1 + 128^4 . 0.05^2}$	399,99
2	6	128^6	$n = \frac{128^6}{1 + 128^6 . 0.05^2}$	400
3	8	128^8	$n = \frac{128^8}{1 + 128^8 . 0.05^2}$	400
4	10	128^{10}	$n = \frac{128^{10}}{1 + 128^{10} . 0.05^2}$	400
5	12	128^{12}	$n = \frac{128^{12}}{1 + 128^{12} . 0.05^2}$	400

TABLE II. Experiments Output

No	Input			Output (Average of)	
	x_1	x_2	x_3	y_1	y_2
1	4	100	AES -Blowfish	0,005933333	0,010333333
2	6	100	AES -Blowfish	0,006066667	0,010966667
3	8	100	AES -Blowfish	0,006866667	0,0112
4	10	100	AES -Blowfish	0,007116667	0,012683333
5	12	100	AES -Blowfish	0,007116667	0,013433333
6	4	100	Blowfish-AES	0,010666667	0,005966667
7	6	100	Blowfish-AES	0,010933333	0,006066667
8	8	100	Blowfish-AES	0,011366667	0,006666667
9	10	100	Blowfish-AES	0,01155	0,006783333
10	12	100	Blowfish-AES	0,0126	0,007433333

In our experiments, 400 key samples are being tested in order to satisfy the minimum requirement of all key size as listed in table 1.

The final result of our experiment is presented in table 2, where the definition of X_1, X_2, X_3, Y_1 and Y_2 with respect to the experiment design in figure 1.

B. Comparison Security Level

In order to focus on security level comparison the decryption time is further observed. Figure 4 presents the decryption time comparison between AES – Blowfish (in blue colour) and Blowfish AES (in red colour). Generally, AES-Blowfish needs longer decryption time.

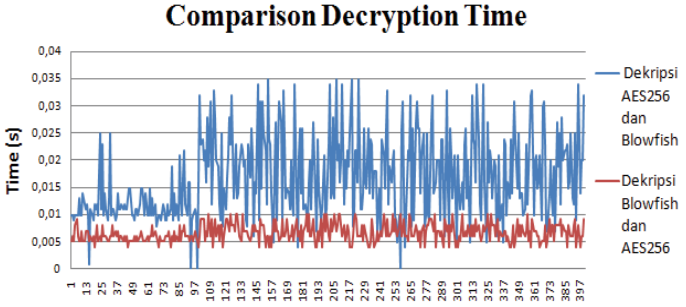

Fig. 4. Comparison Decryption time for 4 key length

Figure 4 depicts the decryption time comparison between combination AES256-Blowfish against Blowfish-AES256, on 4 character key length with 400 experiments. In order to compare the strength of the cryptosystem against the brute force attack, among 400 experiments, an average of time

taken and compare in all scenarios on 4,6,8,10 and 12 key length. Figure 5 presents the comparison between original algorithms and their combination in different order.

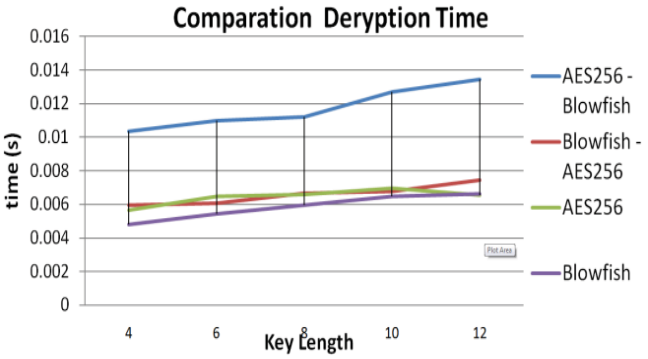

Fig 5. Comparison of Decryption time

Figure 5 compares the average decryption time between AES256, Blowfish, AES256 and Blowfish combined, as well as a combination of Blowfish and AES256. In general, the longest decryption time is achieved by the combination of AES256 followed by Blowfish. It shows significantly longer than that of other combination or the original algorithms for all key length.

In order to calculate the possible time required to crack a password by using a brute force attack, formula (2) is utilized. The required time to performs brute force attack proportional to the average decryption time multiplied by the number of the possible key combination. For instance, if the key length is equal to eight characters as suggested by NIST[24], the possible combination of key is equal to 128^8 = 7,20576 X 1016. The complete possible combination and required time to carry out a brute force attack are presented in table 4.

In order to justify our finding that there is a difference in decryption time between AES-Blowfish and Blowfish-AES a statistical T-test is performed and presented in table 3.

According to the test above, it is statistically proven that there is a differences between AES-Blowfish compare to Blowfish AES in term of decryption time. A one-tail t-test is performed and results from a P-values less than 0.05 as P values references. As can be seen in figure 4, the AES-Blowfish decryption time is generally longer than that's of Blowfish –AES. According to Equation (2), the total brute force time can be calculated refer to decryption time average. The total brute force time for each key-length is presented in table 4.

TABLE III. COMPARATIVE STATISTICAL ONE TAIL T-TEST RESULT BETWEEN AES-BLOWFISH AND BLOWFISH-AES EXPERIMENT RESULT

No	Key Length (Character)	P-Value	Analysis
1	4	3,91E-86	Statistically significant different
2	6	1,86E-05	Statistically significant different
3	8	4.03E-26	Statistically significant different
4	10	0.001837	Statistically significant different
5	12	0.001947	Statistically significant different

TABLE IV. COMPARISON POSSIBLE TIME BRUTE FORCE ATTACK

Password Length	Possible Key	Possible Time Brute Force Attack (Year)			
		AES256	*Blowfish*	*Combination AES256-Blowfish*	*Combination Blowfish-AES256*
4	2,68E+08	4,80E-02	4,09E-02	8,80E-02	5,08E-02
6	4,40E+12	9,02E+02	7,58E+02	1,53E+03	8,46E+02
8	7,21E+16	1,50E+07	1,36E+07	2,56E+07	1,52E+07
10	1,18E+21	2,60E+11	2,43E+11	4,75E+11	2,54E+11
12	1,93E+25	4,01E+15	4,07E+16	8,24E+15	4,56E+15

The simulation is shown in Table 4, possible time comparison can be cracked with the longest brute force attack is AES256 and Blowfish encryption algorithms combined. According to [26], longer decryption time leads to more secure cryptosystem. Therefore, we can conclude that the most secure algorithm in regard to table 4 above is a combination of AES –Blowfish.

V. CONCLUSION

The AES256-Blowfish algorithms combination are generally taken longer time decryption time compared to original algorithm or Blowfish-AES256. A statistical test has been performed to prove that the difference is significant. As presented in table 2, in all of the experiments (4,6,8,10 and 12) key length, the one tail t-test result shows P-values under 0.05. The null hypothesis that there are no differences between AES-Blowfish compared to Blowfish-AES is rejected. Therefore according to the statistical T-Test, we conclude that there a significant differences between AES-Blowfish and Blowfish-AES in term of decryption time. In order to measure the security level of a cryptosystem against the brute force attack, the longer decryption time lead to the stronger cryptosystems. As presented in table 3, the calculation of brute force required time is much longer on AES256-Blowfish compared to any other scenarios. Therefore, AES-Blowfish is considered the most secure algorithm compare to Blowfish-AES256, Blowfish or AES256. According to the experiments result, a strong conclusion that algorithm order significantly affects the security performance in the combination of AES256 and Blowfish.

Although a strong conclusion has been drawn, there are still remain a research challenge in the future such as combining newer encryption algorithms, evaluating the strength of the key. We limit our calculation on blind brute force attack, where the attacker examines all possible key combination. At the moment, guided attack such as minimizing possible combination by the list of possible words in 1–grams to n-grams can be the more effective attack. It would be interesting if the future research considers different attacking techniques.

REFERENCES

[1] S. Jang, "Developing File Security for Windows Operation System," vol. 10, no. 5, pp. 36–39, 2010.

[2] M. A. and M. A. Hossain, "C Loud C Computing S Ecurity in B Usiness," Int. J. Netw. Secur. Its Appl., vol. 6, no. 1, pp. 25–36, 2014.

[3] J. Chauhan, N. Dedhia, and B. Kulkarni, "Enhancing Data Security by using Hybrid Cryptographic Algorithm," Int. J. Eng. Sci. Innov.

Technol., vol. 2, no. 3, pp. 221–228, 2013.
[4] V. kaul A. P Shaikh, “Enhanced Security Algorithm using Hybrid Encryption and ECC,” IOSR J. Comput. Eng., vol. 16, no. 3, pp. 80–85, 2014.
[5] S. Rani, "Implementation and comparison of hybrid encryption model for the secure network using AES and Elgamal," vol. 8, no. 3, pp. 254–258, 2017.
[6] S. K. N. Shaikh Ammarah P, Vikas Kau and Thakur, “Security Enhancement Algorithm for Data Transmission using Elliptic Curve Diffie - Hellman Key Exchange,” vol. 2014, no. Icwac, pp. 10–16, 2014.
[7] G. Yadav, “A Comparative Study of Performance Analysis of Various Encryption Algorithms,” no. March, pp. 70–73, 2017.
[8] S. Nazar, S. Gupta, and A. Thapa, “Performance Analysis of MD-5 AND SHA-1 Hashing Algorithms,” vol. 8491, pp. 333–336, 2012.
[9] K. Shahbazi, M. Eshghi, and R. Faghih Mirzaee, “Design and implementation of an ASIP-based cryptography processor for AES, IDEA, and MD5,” Eng. Sci. Technol. an Int. J., 2017.
[10] P. Patil, P. Narayankar, D. G. Narayan, and S. M. Meena, “A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish,” Procedia Comput. Sci., vol. 78, no. December 2015, pp. 617–624, 2016.
[11] P. C. Mandal, “Superiority of Blowfish Algorithm,” Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 2, no. 9, pp. 196–201, 2012.
[12] B. Dakhare, N. N. Shinde, S. S. Salvi, A. H. Kadam, and P. G. Wagh, “Performance Analysis of Data Encryption Algorithms using AES BLOWFISH and SNAP,” vol. 8, no. 3, pp. 16466–16468, 2018.
[13] Bruce Schneier, “Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish),” 1994.
[14] J. Vashishtha, “Evaluating the performance of Symmetric Key Algorithms : AES (Advanced Encryption Standard) and DES (Data Encryption Standard),” IJCEM Int. J. Comput. Eng. Manag., vol. 15, no. 4, pp. 43–49, 2012.
[15] R. Bhanot and R. Hans, “A Review and Comparative Analysis of Various Encryption Algorithms,” vol. 9, no. 4, pp. 289–306, 2015.
[16] J. E. Grindley and A. J. Tickle, “Advanced Encryption Standard with Galois Counter Mode using Field Programmable Gate Advanced Encryption Standard with Galois Counter Mode using Field Programmable Gate Array .,” 2018.
[17] J. Majumder, “Dictionary Attack on MD5 Hash,” vol. 2, no. 3, pp. 721–724, 2012.
[18] A. Hossain, B. Hossain, S. Uddin, and S. Imtiaz, “International Journal of Advanced Research in Performance Analysis of Different Cryptography Algorithms,” vol. 6, no. 3, pp. 659–665, 2016.
[19] J. Thakur and N. Kumar, “DES, AES and Blowfish: Symmetric key cryptography algorithms simulation based performance analysis,” Int. J. Emerg. Technol. Adv. Eng., vol. 1, no. 2, pp. 6–12, 2011.
[20] A. Ramesh and A. Suruliandi, “Performance analysis of encryption algorithms for information security,” Proc. IEEE Int. Conf. Circuit, Power Comput. Technol. ICCPCT 2013, pp. 840–844, 2013.
[21] Qashlim and A. Rusdianto, “Implementasi Algoritma Md5 Untuk Keamanan Dokumen,” J. Ilm. Ilmu Komput., vol. 2, no. 2, pp. 10–16, 2016.
[22] P. J. Ashenden, The designer’s Guide to VHDL. 2008.
[23] M. K. Neha, “Enhanced Security using Hybrid Encryption Algorithm,” Int. J. Innov. Res. Comput. Commun. Eng., vol. 4, no. 7, pp. 13001–13007, 2016.
[24] J. L. Fenton et al., “Digital Identity Guidelines,” Natl. Inst. Stand. Technol., 2017.
[25] B. Yoshogi, I. T. Bandung, and J. G. Bandung, “Peluang dan Kombinasi pada Penjebolan Password,” pp. 1–5, 2011.
[26] Inayatullah, Analisis Penerapan Algoritma MD5 Untuk Pengamanan Password, vol. 3, no. 3. 2007.