Journal of Intelligent Decision Support System (IDSS) 6 (4) (2023), pp. 285-294

Published by: Institute of Computer Science (IOCS)

Journal of Intelligent Decision Support System (IDSS)

Journal homepage: www.idss.iocspublisher.org

Data Security Analysis with Triple DES Cryptographic Algorithm

Liskedame Yanti Sipayung[1], Megaria Purba[2]

[1,2]*Prodi Teknik Informatika, STMIK Pelita Nusantara Medan, Indonesia*

Article Info

Article history:

Received Dec 9, 2023
Revised Dec 20, 2023
Accepted Dec 30, 2023

Keywords:

Cryptography
3DES (Triple Data Encryption Standard)
DES (Data Encryption Standard)
Encryption
Decryption
Key

ABSTRACT

The Data Encryption Standard (DES) is the first and most significant modern symmetry encryption algorithm. DES was released by the United States' National Bureau of Standards in January 1977 as an algorithm used for unclassified data (information that has nothing to do with national security). Although DES has been widely used in many applications around the world, there are still many controversial issues regarding the security of this algorithm. Therefore, this paper aims to discuss the security analysis of the DES cryptographic algorithm and also the security analysis of the DES algorithm variations, namely Double DES and Triple DES. Because the level of secrecy of the 3DES algorithm lies in the length of the key used, the use of the 3DES algorithm is considered more secure than the DES algorithm. To facilitate the use of the 3DES algorithm, a 3DES algorithm program was created with computer software tools, namely Matlab 7.0.4 which can encrypt and decrypt files with the .txt extension.

Corresponding Author:

Liskedame Yanti Sipayung,
Teknik Informatika,
STMIK Pelita Nusantara Medan,
Jl. Iskandar Muda No.1, Merdeka, Kec. Medan Baru, Kota Medan, Sumatera Utara 20222, Indonesia
Email: liskedamesipayung@gmail.com

Introduction

Through this fast advancement in systems, the security dividers will be break by unapproved people without any problem. In this manner, by making a successful and effective Data assurance by utilizing the private insurance for scrambling the information and picture will be convert into the garbled information so customary sign will be occur before encryption and after decoding (Srivatsava & Sheeja, 2020).

The most widely used encryption algorithm in the world is the Data Encryption Standard (DES) which was adopted by NIST (National Institute of Standards and Technology) as the US Federal information processing standard(Mohamad et al., 2017). Data is encrypted in 64 bit blocks using a 56 bit key. DES transforms a 64 bit input in multiple encryption stages into a 64 bit output. As such, DES belongs to the block cipher family(Des et al., 2021). With the same stages and key, DES is used to reverse the encryption. DES is widely used to protect data in the world of electronics, especially in banking, finance, and e-commerce (Hooker et al., 2020). Unfortunately, DES also has controversies about its security. To find out, let's look at the history of its creation. In the late 60s, IBM started research in cryptography which was later called LUCIFER and sold to a company in London in 1971(Permana et al., 2020). LUCIFER was a block cipher that operated on 64 bit input blocks, using a 128 bit key. A version of LUCIFER was later developed that was more immune to cipher analysis, but with a reduction in key size, to 56 bits, in order to fit on a single chip. Meanwhile, the American Bureau

of Standards required a national standard cipher. IBM registered its cipher which eventually became DES (Data Encryption Standard) in 1977. There were two problems in this case. The original 128 bit key was reduced by 72 bits to only 56 bits. A reduction that was too large, making it vulnerable to brute force attacks. The second problem was that the design of DES's internal structure, its substitution part (S-box), was still secret. This S-box was changed following the NSA's suggestion. As a result, we cannot be sure that the internal structure of DES is free of intentionally hidden weak points, which allow the NSA to crack the cipher without knowing the key.

Cryptography is a method used to create secure communication by manipulating sent messages during the communication occurred so only intended party that can know the content of that messages.(Li et al., n.d.) According to cryptography experts' research, DES is so carefully designed that if you randomly change these S-boxes, it is very likely that the DES you produce will become easier to crack. Through this fast advancement in systems, the security dividers will be break by unapproved people without any problem

Confidentiality is a service that is used to protect the contents of the

information from anyone except those who have the authority or secret key to open / peel the information that has been encrypted. Data integrity, is related to guarding against unauthorized changes to data. To maintain data integrity, the system must have the ability to detect data manipulation by unauthorized parties, including insertion, deletion and substitution of other data into the actual data. Authentication, is related to identification/recognition, both in terms of the system unit and the information itself. Two communicating parties must introduce themselves to each other. Information sent through the channel must be authenticated for authenticity, data content, delivery time, and so on. Non-repudiation or denial is an attempt to prevent denial of the sending/creation of information by the sender/creator.

Aspects of confidentiality, data integrity, authentication and non-repudiation that strengthen the writer to discuss the topic of Data Security Analysis with Triple DES Cryptographic Algorithm.

Method

Triple DES is also known as TDES, or more standardly TDEA (Triple Data Encryption Algorithm). TDES applies DES three times, with three keys(Informatika et al., 2018).

Here is the global scheme of TDES:

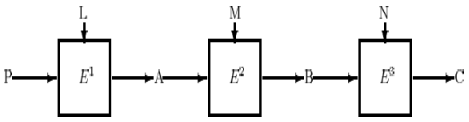


Fig 1. Global schematic of TDES a. Logical Operators

Binary operators are identical to bits on a computer, involving 0s and 1s. The operator used in the 3DES algorithm is XOR. The XOR operator is used for two inputs. If the two inputs are equal then the output value is 0, and if the two inputs are different then the output value is 1.

Table 1. Operator XOR

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

b. Basic Math and Function Relation

Definition 3.1 A relation *f* from A to B is said to be a function if every x ∈ of A is paired or mapped to exactly one element in B (Bartle, 1994)

Definition 3.2 f: A → B is called an injective or one-one function if :

$$x_1 \neq x_2 \Rightarrow f(x_1) \neq f(x_2)$$

or if

$f(x_1) = f(x_2) \rightarrow x_1 = x_2$ (*Bartle, 1994)*

The encryption process and decryption process can be expressed in mathematical notation as follows:

$$E_K(P) = C \quad \text{and} \tag{1}$$

$$D_K(C) = P \tag{2}$$

and the whole can be expressed as:

$$D_K(E_K(P)) = P \tag{3}$$

The relation between the set P (plaintext) and the set C (ciphertext) must be a one to one correspondence function. That is, in the decryption process there is only one element of C that represents one element of P

Results and Discussions

Data Encryption Standard

DES operates on a 64-bit block size. DES encrypts 64-bit plaintext into 64-bit ciphertext by using a 56-bit internal key generated from an external key that is 64-bit long(Patil et al., 2016).

Key Processing

The inputted external key will be processed to obtain 16 internal keys. First, the 64-bit long external key is substituted in the PC-1 compression permutation matrix(Kriptografi & Transposition, 2017). In this permutation, every eighth bit (parity bit) of the eight bytes is ignored. The result of the permutation is 56-bit in length, which is then divided into two parts, namely the left (C0) and right (D0) each 28-bit in length. Then, the left and right parts perform bit shifts on each round by one or two bits depending on each round. In the encryption process, the bit shifts to the left (left shift). As for the decryption process, the bits shift to the right (right shift). After experiencing a bit shift, Ci and Di are combined and substituted into the compression permutation matrix using the PC-2 matrix, resulting in a 48-bit length. The process is done 16 times repeatedly.

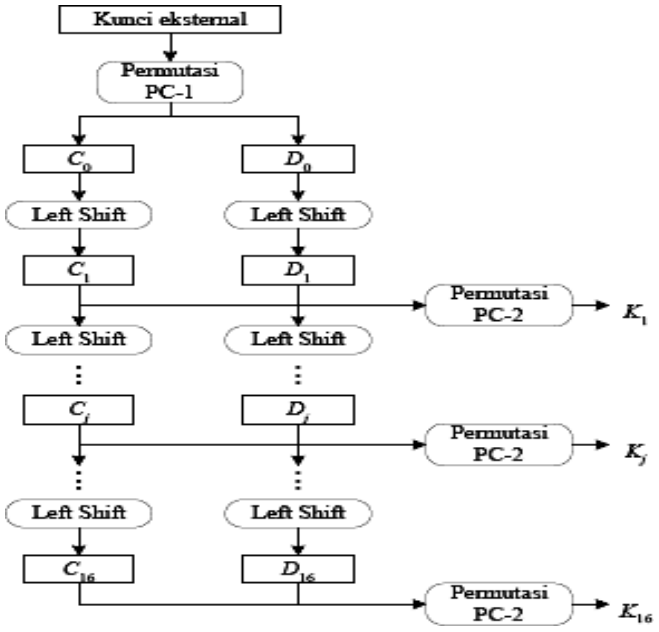


Figure 2. DES Internal Keys Generation Process (Stinson, 1995)

Encryption Process

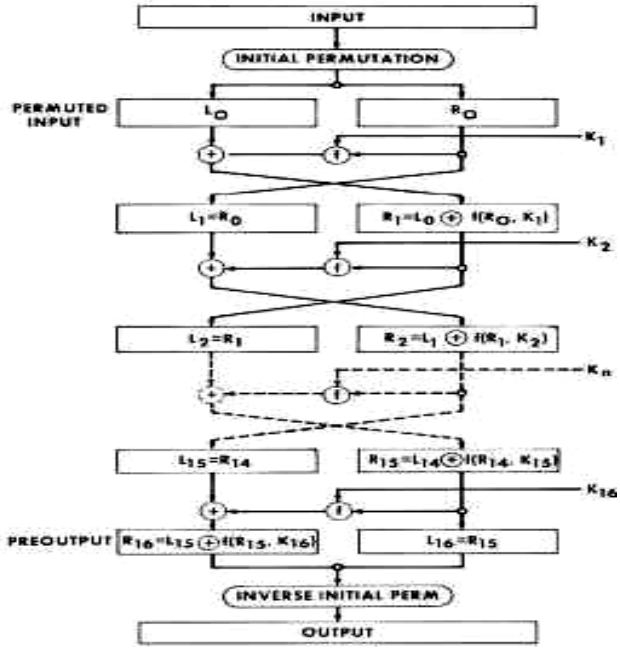


Figure 3. DES Encryption Process (NIST, 2004)

The inputted plaintext will first be subattributed to the initial permutation matrix or IP of 64-bit length. Then it is divided into two parts, namely the left (L) and right (R) each being 32-bit long. These two parts enter into 16 rounds of DES. One round of DES is a Feistel network model, mathematically the Feistel network is expressed as follows:

$$L_i = R_{i-1;\ 1\leq} \ i \leq 16 \quad (4)$$

$$R_i = L_{i-1} \ ? \ f(R_{i-1}, k_i) \quad (5)$$

The R part is substituted in the 48-bit length expansion function and then XORed with the internal key that has been processed previously in the key generation process (in the first round using the first internal key, and so on). The XOR result is then substituted in the S-box which is grouped into 8 groups, each 6-bit result becomes 4-bit. The first 6-bit group uses S1, the second 6-bit group uses S2, and so on. After the S-box process, the length becomes 32-bits. Then it is substituted again in the P-box permutation matrix, then XORed with the L part. The result of the XOR is stored for the next R part. Meanwhile, the L part is obtained from the previous R part. The process is done 16 times. After 16 rounds are completed, the L and R parts are combined and substituted in the inverse initial permutation matrix or IP-1, the result is a 64-bit ciphertext.

Decryption Process

The decryption process of the ciphertext is the reverse of the encryption process. DES uses the same algorithm for the encryption and decryption process. If in the encryption process the internal key sequence used is k1, k2, ..., k16 then in the decryption process the internal key sequence used is k16, k15, ..., k1(Informatika & Cikarang, 2017).

In the first stage, the inputted plaintext is operated with the first external key (K1) and performs the encryption process using the DES algorithm. Thus producing the first pre-ciphertext. The second stage, the first pre-ciphertext generated in the first stage, is then operated with the second external key (K2) and performs the encryption process or decryption process (depending on the encryption method used) using the DES algorithm. This produces the second pre-ciphertext. The last

stage, the second pre-ciphertext produced in the second stage, is operated with the third external key (K3) and performs the encryption process by using the DES algorithm, resulting in the ciphertext (C).

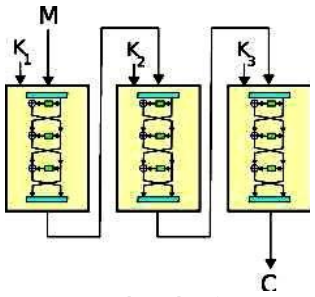


Figure 4. DES algorithm (NIST, 2004)

Key Selection

There are two options for external key selection of the 3DES algorithm, namely: a) K1, K2, and K3 are mutually independent keys

A. $K_1 \neq K_2 \neq K_3 \neq K_1$

b) K1 and K2 are mutually independent keys, and K3 is equal to K1

$\mathbf{K_1 \neq K_2}$ dan $\mathbf{K_3 = K_1}$ (*NIST, 2004*)

Encryption and Description Process

The encryption and decryption process of the 3DES algorithm can be achieved in several ways, that is:

Table 2. How to encrypt and decrypt

How to	Encryption	Decryption
1	DES – EDE2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $C = E [D \{E (P, K_1), K_2\}, K_3]$	DES – DED2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $P = D [E \{D (C, K_3), K_2\}, K_1]$
2	DES – EEE2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $C = E [E \{E (P, K_1), K_2\}, K_3]$	DES – DDD2 ▪ $K_1 \neq K_2, K_3 = K_1$ ▪ $P = D [D \{D (C, K_3), K_2\}, K_1]$
3	DES – EDE3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $C = E [D \{E (P, K_1), K_2\}, K_3]$	DES – DED3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $P = D [E \{D (C, K_3), K_2\}, K_1]$
4	DES – EEE3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $C = E [E \{E (P, K_1), K_2\}, K_3]$	DES – DDD3 ▪ $K_1 \neq K_2 \neq K_3 \neq K_1$ ▪ $P = D [D \{D (C, K_3), K_2\}, K_1]$

The design begins with the creation of a context diagram, in the form of a system overview of the application of the 3DES algorithm.

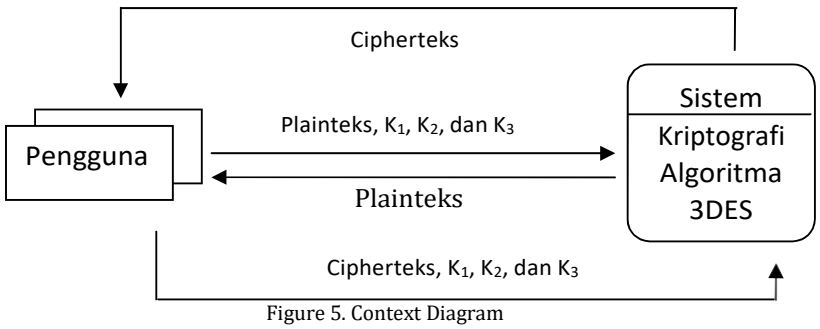
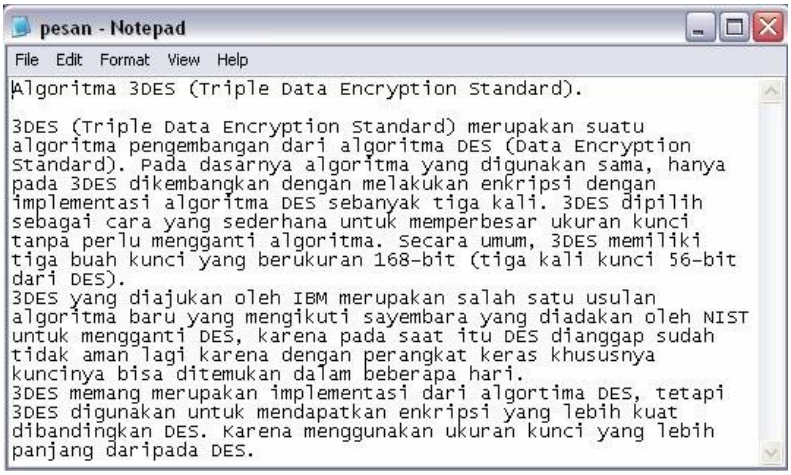


Figure 5. Context Diagram

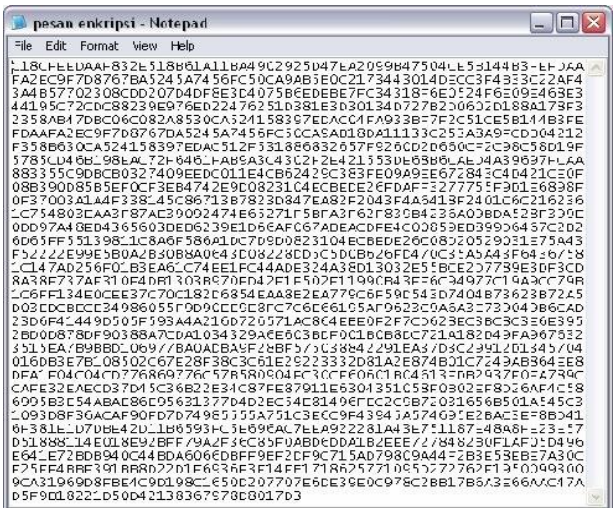
Program Results

The following example files to be encrypted and decrypted are taken from files with the .txt extension which are 1 KB (Kilo Byte) and the keys used are mutually exclusive (K1 ≠ K2 ≠ K3 ≠ K1), namely: Key 1: Encryption, Key 2: Security, Key 3: Decryption. The chosen encryption method is DES - EDE3 and the chosen decryption method is DES - DED3. Example of plaintext file:



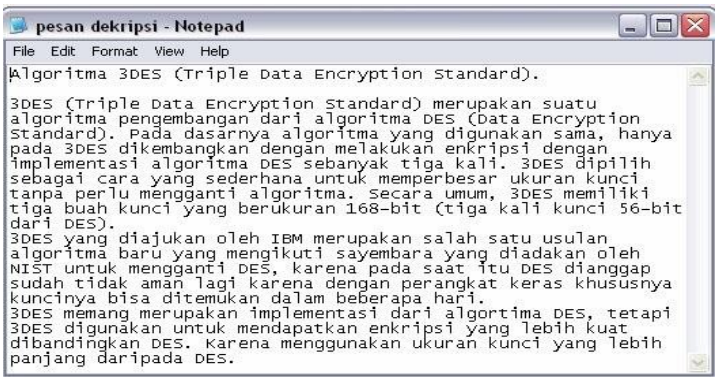
The application to be displayed is as follows:

Example of a ciphertext file:



The ciphertext above will be decrypted using the same three keys in the encryption process. The application that will be displayed is as follows:

Then the result will be the same as the original plaintext, that is:



The following will show the file process for the DES algorithm and the 3DES algorithm, with the keys used as follows: Key 1: Software, Key 2: Computer, Key 3: Hardware

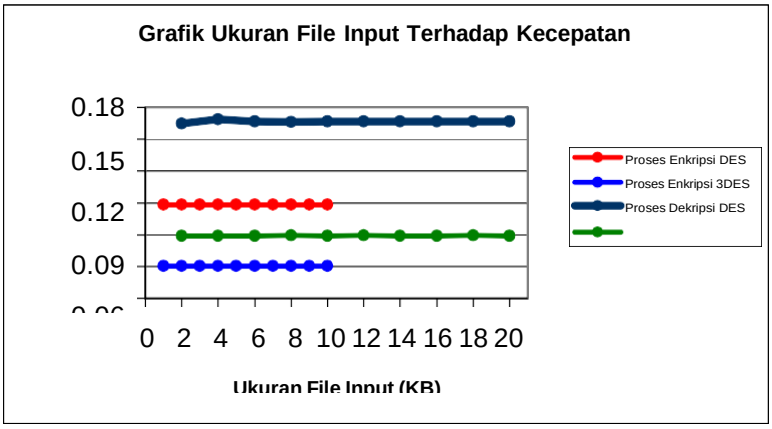
Table 3. Processing Time and Speed for Encryption Process with DES Algorithm and 3DES Algorithm

No	File Name		File Size (KB)		Processing Time (Seconds)		Speed (KB/detik)	
	Input	Output						
		DES 3DES	Input	Output	DES	3DES	DES	3DES
1	P1.txt	EP1 DES.txt EP1	1	2	11.34	33.093	0.08818	0.03022
2	P2.txt	EP2 DES.txt 3DES.txt EP2	2	4	22.658	66.197	0.08827	0.03021
3	P3.txt	EP3 DES.txt 3DES.txt EP3	3	6	33.98	99.302	0.08829	0.03021
4	P4.txt	EP4 DES.txt 3DES.txt EP4	4	8	45.26	132.324	0.08838	0.03023
5	P5.txt	EP5 DES.txt 3DES.txt EP5	5	10	56.586	165.29	0.08836	0.03025
6	P6.txt	EP6 DES.txt 3DES.txt EP6	6	12	67.924	198.463	0.08833	0.03023
7	P7.txt	EP7 DES.txt 3DES.txt EP7	7	14	79.262	231.15	0.08831	0.03028
8	P8.txt	EP8 DES.txt 3DES.txt EP8	8	16	90.733	264.882	0.08817	0.03020
9	P9.txt	EP9 DES.txt 3DES.txt EP9	9	18	101.909	297.451	0.08831	0.03026
10	P10.txt	EP10 DES.txt 3DES.txt EP10	10	20	113.342	330.389	0.08823	0.03027
		Average Speed					0.08828	0.03024

Table 4. Processing Time and Speed for Decryption Process with DES Algorithm and 3DES Algorithm

No	File Name				File Size (KB)		Processing Time (detik)		Speed (KB/detik)	
	Input		Output							
	DES	3DES	DES	3DES	Input	Output	DES	3DES	DES	3DES
1	EP1 DES.txt	EP1 3DES.txt	DP1 DES.txt	DP1 3DES.txt	2	1	12.135	33.887	0.16481	0.05902
2	EP2 DES.txt	EP2 3DES.txt	DP2 DES.txt	DP2 3DES.txt	4	2	23.726	67.731	0.16859	0.05906
3	EP3 DES.txt	EP3 3DES.txt	DP3 DES.txt	DP3 3DES.txt	6	3	36.015	101.707	0.16660	0.05899
4	EP4 DES.txt	EP4 3DES.txt	DP4 DES.txt	DP4 3DES.txt	8	4	48.062		0.16645	0.05913
5	EP5 DES.txt	EP5 3DES.txt	DP5 DES.txt	DP5 3DES.txt	10	5	59.978	135.303	0.16673	0.05909
6	EP6 DES.txt	EP6 3DES.txt	DP6 DES.txt	DP6 3DES.txt	12	6	72.044		0.16656	0.05915
7	EP7 DES.txt	EP7 3DES.txt	DP7 DES.txt	DP7 3DES.txt	14	7	84.009	169.244	0.16665	0.05910
8	EP8 DES.txt	EP8 3DES.txt	DP8 DES.txt	DP8 3DES.txt	16	8	95.941		0.16677	0.05907
9	EP9 DES.txt	EP9 3DES.txt	DP9 DES.txt	DP9 3DES.txt	18	9	107.959	202.868	0.16673	0.05911
10	EP10 DES.txt	EP10 3DES.txt	DP10 DES.txt	DP10 3DES.txt	20	10	119.877	236.904	0.16684	0.05906
								270.864		
								304.506		
								338.645		
									0.16667	0.05908

Where P is the message, EM is the message encryption, and DP is the message decryption.



Key Secrecy Level

The longer the key used, the stronger the secrecy level. The 3DES algorithm uses a key that is 168 bits long, so the total number of possible key combinations that must be tried to decipher the ciphertext is 2168 = 3,741x10^{50} times. This is because there are 168 bit-filling positions, each of which has two possible values, 0 and 1.

Conclusions

The process of encrypting and decrypting data with the 3DES algorithm is done by implementing the DES algorithm three times, according to the key selection and the selected process sequence. The time required for the encryption and decryption process is influenced by the file size, the specifications on the hardware, and other processes being performed by the hardware. Plaintext processed with key 1, key 2, and key 3 produces ciphertext with a larger number of characters, due to the padding process and is stored in hexadecimal form. If one of the keys or all three keys are changed, the ciphertext will also change.The speed for the encryption and decryption process is the same for every 1 KB increase in the input file size. For the 3DES algorithm, the average speed of the encryption process is 0.03024 KB/second and the average speed of the decryption process is 0.05908 KB/second. As for the DES algorithm, in the encryption process the average speed is 0.08828 KB/second and in the decryption process the average speed is 0.16667 KB/second.To get the plaintext without knowing the key, the number of possible key combinations that must be tried is 3,741 x 10^{50} times.The time required to try all possible keys by a brute force attack is 1,183 x 10^{43} years. The time required is very short and the speed in the encryption and description process is affected by the condition of the file to be processed.

References

Achmad, Ikhwanudin. 2007. *An Application of Generalized Inverse Matrices on the Hill Cipher*, (online), http://www.ikhwan.web.ugm.ac.id, (diakses 22 Januari 2008).

Des, A. T., Siahaan, M., & Manurung, J. (2021). *Perancangan Aplikasi Penyandian Teks Menggunakan*. *3*(3), 197–201.

Felix, Fidens. 2006. *Dasar Kriptografi,* (online), http://www.ilmukomputer.com, (diakses September 2007).

Hasan, Rusydi. 2003. *Mengenal Algoritma DES,* (online), http://www.ilmukomputer.com, (diaksesSeptember 2007).

Hooker, S., Midorikawa, K., Rosenzweig, J., Environments, P., Gottfried, K. C., Dinh, H. N., Randolph, K., & Adam, Z. (2020). *Time Evaluation Of Different Cryptography Algorithms Using Labview TIME EVALUATION OF DIFFERENT CRYPTOGRAPHY*. https://doi.org/10.1088/1757-899X/745/1/012039

Informatika, T., & Cikarang, S. (2017). *METODE ALGORITMA DES DAN METODE END OF FILE Ajar Rohmanu*. *2*(1), 1–11.

Informatika, T., Teknik, F., & Riau, U. I. (2018). *PERBANDINGAN METODE DATA ENCRYPTION STANDARD (DES) DAN ADVANCED ENCRYPTION STANDARD (AES) PADA STEGANOGRAFI FILE CITRA*. 229–236.

Kriptografi, K. K., & Transposition, C. (2017). *ANALISA DAN IMPLEMENTASI KRIPTOGRAFI PADA PESAN RAHASIA*. *3*(1), 1–11.

Li, X., Measurements, L., & Bechtol, K. (n.d.). *Implementation Cryptography Data Encryption Standard (DES) and Triple Data Encryption Standard (3DES) Method in Communication System Based Near Field Communication (NFC) Implementation Cryptography Data Encryption Standard (DES) and Triple Data Encryption Standard (3DES) Method in Communication System Based Near Field Communication (NFC)*.

Mohamad, M., Ahmad, I., & Fernando, Y. (2017). Pemetaan Potensi Pariwisata Kabupaten Waykanan Menggunakan Algoritma Dijkstra. *Jurnal Komputer Terapan*, *3*(2), 169–178. http://repository.teknokrat.ac.id/198/

NIST. 2004. *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*, (online), http://www.csrc.nist.gov, (diakses 22 Januari 2008).

Patil, P., Narayankar, P., Narayan, D. G., & Meena, S. M. (2016). A Comprehensive Evaluation of Cryptographic Algorithms : DES ,. *Procedia - Procedia Computer Science*, *78*(December 2015), 617–624. https://doi.org/10.1016/j.procs.2016.02.108

Permana, A. A., Informatika, P. T., Teknik, F., & Tangerang, U. M. (2020). *IMPLEMENTASI STEGANOGRAPHY PADA AUDIO MENGGUNAKAN ALGORITMA END OF FILE (EOF)*. *9*(2017), 91–98.

Stinson, Douglas. 1995. *Cryptography: Theory and Practice,(online),* http://www.easywebtech.com, (diakses 22 Januari 2008).

Srivatsava, J. G. M., & Sheeja, R. (2020). *Implementation of Triple DES ALGORITHM in Data Hiding and Image Encryption Implementation of Triple DES ALGORITHM in Data Hiding and Image Encryption Techniques*. *January*.

Risanto. 2006. *Keamanan Data dengan Kriptografi Kunci Simetris Algoritma DES*. Skripsi tidak diterbitkan. Bandung: Program PascasarjanaUNPAD.