Scientific Journal of Informatics
Vol. 13, No. 2, May 2026
p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

An Evaluation of AES-256 Based PDF Document Security Using Cryptographic Metrics and Blockchain Based Hash Storage

Ellen Chandra[1], Nur Rochmah Dyah Puji Astuti[2*]
[1, 2]Department of Informatics, Universitas Ahmad Dahlan, Indonesia

Abstract.
Purpose: This research aims to design and implement a digital document security by integrating the Advanced Encryption Standard (AES) 256-bit algorithm with blockchain technology to protect documents from manipulation, forgery, and information leakage.
Methods: The study involved implementation using Python. AES-256 was applied to encrypt digital documents, while the hash value of the ciphertext was stored on the blockchain to ensure integrity and authenticity. System performance was evaluated through avalanche effect testing, entropy analysis, and process time measurement.
Result: The average efficiency is 49.989% for small documents, 49.988% for documents 50-200 KB and 49.998% for documents larger than 200 KB, using good bits. An average entropy value of 7.99 indicates a high performance. The average sending time is 1.62 ms, and the average blockchain sending time is 1061.862 ms making it possible to send in sending digital documents.
Novelty: This study integrates AES-256 encryption with blockchain-based hash storage to simultaneously ensure confidentiality, integrity, and authenticity of digital documents with high efficiency.

Keywords: AES-256, Avalanche Effect, Blockchain, Entropy, Processing Time
Received March 2026 / **Revised** April 2026 / **Accepted** May 2026

INTRODUCTION

The development of document digitization is currently an important need because it can facilitate organizations and individuals in carrying out work to the distribution process through various electronic media. But such ease is also followed by increased security risks, such as duplication of documents and dissemination of digital documents without the permission of the rightful owner. This risk becomes serious if the document has a high level of confidentiality and strategic value, because leaks and changes in the content of information can have a significant negative impact [1]. These conditions can trigger various adverse consequences, such as financial losses, declining reputation to disruption of operational activities. As the digital exchange of information increases, the need for document protection is also increasing. This requires stronger document security systems and mechanisms to ensure availability, confidentiality and data integrity are maintained from security threats and abuse [2]. One of the security technologies that can be applied to protect documents is the use of AES algorithm cryptographic mechanisms and utilizing blockchain-based storage to maintain the integrity and authenticity of distributed data.

Advanced Encryption Standard (AES) is a cryptographic algorithm using the same key in the encryption and decryption processes. AES is available in several key length variations, one of which is AES-256 which is known to have a very high level of security [3], [4], [5]. In the AES algorithm there are several modes of operation, one of which is the cipher feedback (CFB) mode. Cipher Feedback (CFB) Mode is one of the modes of operation in the AES algorithm that converts block ciphers into stream ciphers, where the encryption process is carried out continuously by utilizing the encryption results from the previous block as input in the next process [6]. Through the application of the AES algorithm, documents can only be accessed by parties who have encryption keys, and are able to improve document protection from unauthorized access and the risk of digital document forgery by unauthorized parties [7], [8].

*Corresponding author.
Email addresses: rochmahdyah@tif.uad.ac.id (Astuti)*
DOI: 10.15294/sji.v13i2.45421

Blockchain technology is a distributed system that is able to maintain the security and integrity of data by recording every transaction or change into a series of interconnected blocks that are difficult to modify. Blockchain has the characteristics of transparency, decentralization, and immutability, so that the data that has been recorded is not easily changed [9], [10] because the storage is not centralized on a single server, attempts to manipulate or steal data become much more difficult without the approval of the network. Each transaction can also be verified by all parties involved, so that the validation process takes place more quickly, efficiently and does not require a third party [11], [12].

Various previous studies have shown that cryptographic algorithms and technologies have been widely used to improve the security of digital data. Research conducted by Shibghotullah Al Murod and Suhirman (2024) [13] uses an android-based e-voting system development method built by integrating AES-128 to secure voter data, votes and voting results. The results showed that the system is able to maintain the confidentiality and integrity of data through the encryption process, as well as improve efficiency compared to conventional methods. The performance metrics used include aspects of data security, system functionality and process efficiency show an increase in system performance [13]. Research conducted by Machvira Ul Husna and Prita Delliab (2021) [14] implements blockchain technology for the optimization of the transportation document security system on driver licenses and vehicle registration, which aims to improve the security and authenticity of documents. The method used is a consortium blockchain-based system by utilizing the Corda framework, smart contracts, and network consensus mechanisms. The results showed that the system is able to improve security through data decentralization, immutability, and traceability capabilities that can prevent falsification and manipulation of data. The performance metrics used are qualitative, including aspects of security, decentralization, transparency, traceability and system reliability [14].

Compared to research conducted by Shibghotullah Al Murod and Suhirman (2024) [13] which only uses the AES-128 algorithm on centralized systems, as well as research by Machvira Ul Husna and Prita Delliad (2021) [14] which focuses on blockchains without strong encryption on the data content. This study combines the AES-256 algorithm and Blockchain, AES-256 is implemented using Cipher Feedback (CFB) mode to encrypt data, thus enabling the encryption process in streaming without padding and suitable for various file sizes, while blockchain technology is used as distributed storage to ensure the integrity and authenticity of data through hash recording. The novelty of this study lies in the integration of two layers of security so as to meet the aspects of confidentiality, integrity and authenticity.

METHODS

The research methods used in the study, described in more detail through the flow chart shown in Figure 1.

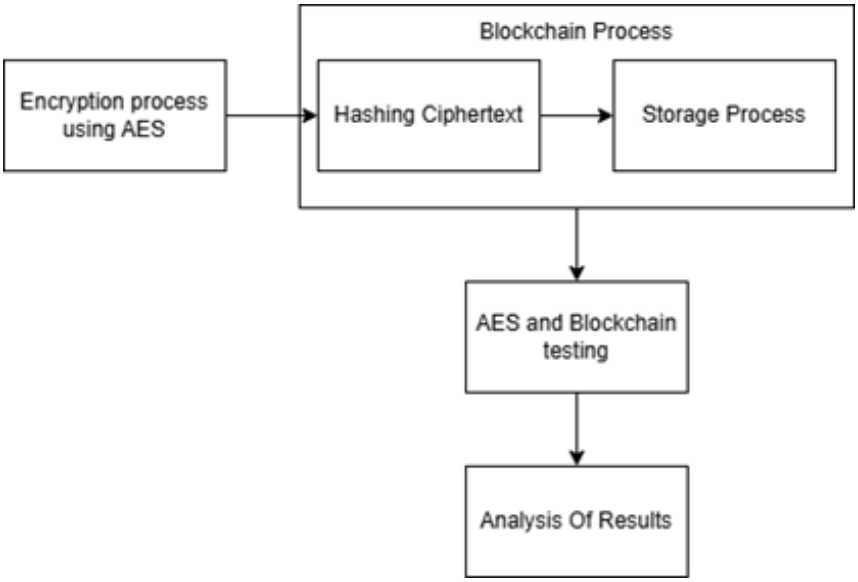

Figure 1. Research Flow

Encryption Process Using AES

Advanced Encryption Standard (AES) is a block cipher algorithm that uses symmetric keys for encryption and decryption processes. The AES algorithm has varying key lengths, one of which is 256-bit (32 bytes) and has as many as 14 turns [3], [4], [5], a key expansion process is required at each stage of encryption which aims to generate a series of keys derived from the main key, through the key schedule mechanism. The key schedule process consists of several stages of operation, namely, Rotate by rotating as many as 8-bits against a 32-byte key block. SubBytes replaces the 8-bit of the sub key with the value of the S-Box table. Rcon applies a constant calculated based on the exponent of two of the values specified by the user, as well as XOR performs logical operations with wi [- Nk], that is, the word derived from the previous position Nk [15].

The encryption process on the AES-256 algorithm involves four main transformation stages, namely SubBytes, ShiftRows, MixColumns and AddRoundKey. Input Data is first entered into the state structure, then processed through a series of transformations repeatedly according to the length of the key and the size of the block used to produce ciphertext [5], [16]. The initialization vector (IV) in CFB is written first, then the result is XOR-ed with plaintext to produce ciphertext. The resulting Ciphertext is then used as input in the next process, resulting in a feedback mechanism (feedback). With this size, CFB mode does not use padding and is suitable for data with a size that is not fixed or streaming [6], [17].

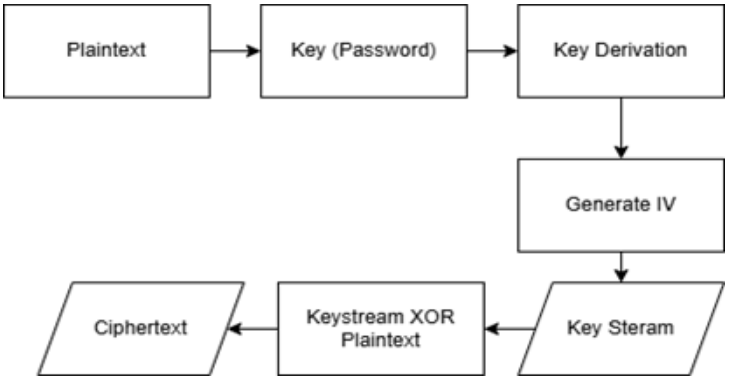

Figure 2. AES-256 Encryption Process

The flowchart in Figure 2, illustrates the flow of document encryption process using AES-256 algorithm with Cipher Feedback (CFB) mode of operation starting from plaintext data processed along with the user's password, the password is processed through the key derivation stage to generate a 256-bit cryptographic key that complies with the AES-256 standard. Once the key is formed, the system encrypts the Initialization Vector (IV) using the encryption key to form the keystream. Furthermore, the keystream is combined with plaintext through XOR operations to produce ciphertext.

Blockchain Process

Blockchain is a technology that supports the secure exchange of information through a decentralized system. Blockchain technology allows each user to record transactions on the system that cannot be changed and can be traced back, so that the verification process can be carried out without the need for an intermediary. In concept blockchain has the principle of distributed, transparent, secure and decentralized. Blockchain can be understood as a distributed database composed of a series of interconnected data blocks and secured using cryptographic mechanisms [18], [19], [20]. Each block contains transaction data along with the hash value of the previous block, the first block is called the genesis block, has an initial index and does not have the hash value of the previous block. The hash value generated from the genesis of the block is then used as a hash reference in the next block, thus forming links between blocks on an ongoing basis [21], [22].

Hash is an important component in blockchain technology that serves to maintain data integrity while increasing transaction security. Hash functions work by changing input data that has various sizes. One of the most widely used hash algorithms in blockchain systems is the 256-bit Secure Hash Algorithm (SHA-256), each hash generated is unique so that small changes to the original data will result in very different hash values. In the blockchain mechanism, the hash is used to connect each block with the previous block

through the insertion of the hash of the previous block in the header of the new block, if there is a change in data on one block, then the entire chain of hashes on the next block will also change [23], [24].

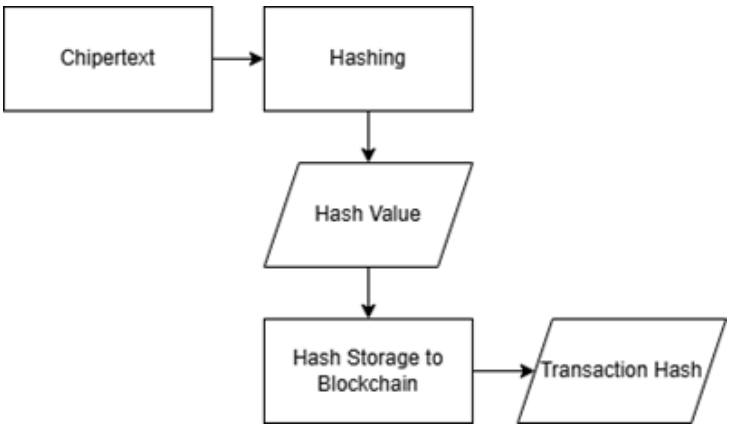

Figure 3. Blockchain Storage Process

The Flowchart in Figure 3 illustrates the process of storing data into the blockchain, the ciphertext resulting from the encryption is processed through the hashing stage using the SHA-256 hash algorithm to produce a unique hash value with a fixed length of 256-bits (32 bytes).

Avalanche Effect

Avalanche Effect is a test method used to determine the extent to which small changes in the original data (plaintext) can affect the encryption results (ciphertext) [25], [26]. The measurement is done by comparing the number of bits in the ciphertext to the total bits in the plaintext. The greater the percentage of bit changes produced, the better the diffusion rate and strength of the encryption algorithm. In general, a cryptographic algorithm is considered effective and safe if the bit change value is in the range of 45% to 60% with an ideal value close to 50%. The high percentage of changes causes the ciphertext pattern to be difficult to predict, so that attempts to analyze or break into data by unauthorized parties become much more difficult [27], [28]. Avalanche effect value is calculated using the formula:

$$\text{AE} = \frac{\text{Number of bits remaining}}{\text{Number of bits length}} \times 100\% \quad (1)$$

Entropy

Entropy is used as an indicator to measure the degree of randomness of the ciphertext resulting from the encryption process. The calculation of entropy is based on the concept of Shannon Entropy, which is expressed through mathematical equations.

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 p(x_i) \quad (2)$$

In the above formula, H(X) indicates the entropy value of the random variable X, while $p(x_i)$ states the chance of each xi symbol appearing in the encrypted data. The higher the entropy value obtained, the higher the level of randomness of the data indicating that the quality of ciphertext security becomes better and difficult to predict [29], [30] maximum entropy value is at a value of 8, which indicates a high level of randomness, the closer the value is, the encryption results have better ciphertext security quality [31].

Processing Time

Process time testing is carried out to determine the level of efficiency of the encryption process and the process of storing hashes into the blockchain. Measurement of encryption time aims to determine the length of time it takes AES-256 algorithm in converting plaintext into ciphertext. Blockchain storage time testing is done to measure the duration of sending the hash value to the blockchain network until the transaction is recorded in a block.

RESULTS AND DISCUSSIONS

The Dataset used for the research is in the form of a PDF file, which is obtained through GitHub https://github.com/JohnSnowLabs/pdf-deid-dataset/tree/main, a total of 30 PDF files with small (< 50 KB), medium (50-200 KB) and large (> 200 KB) sizes, each size totaling 10 files.

Data Encryption Process

The encryption process starts from plaintext in the form of a digital document, then the password entered by the user is processed at the stage of derivation to generate an encryption key that complies with the AES-256 standard. The next process is generate initialization vector (IV) randomly which serves as the initial value in CFB operation mode. the next stage is the formation of a keystream, where IV is encrypted using AES-256 algorithm with a key that has been generated, the results of this IV encryption generate a keystream which is then used in the XOR process with plaintext. XOR keystream and plaintext operations generate ciphertext as the final output of the encryption process. For example, the resulting ciphertext as in the Figure 4.

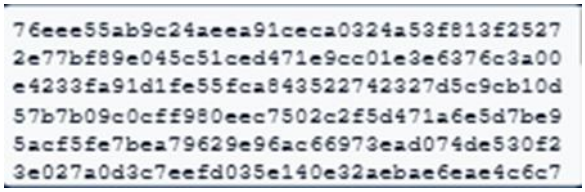

Figure 4. Ciphertext Examples

Figure 4 shows the ciphertext generated from the encryption process using the AES-256 algorithm with the CFB mode of operation. Ciphertext is then used as input data at the hashing stage before being stored into the blockchain system.

Blockchain Storage Process

After the encryption process generates the ciphertext, the next step is to store the encrypted metadata into the blockchain system to ensure the integrity and authenticity of the data. Ciphertext is not directly stored in the blockchain, but is hashed to produce a hash value that is unique and representative of the contents of the data. The hashing process aims to convert ciphertext into a digital fingerprint that cannot be returned to the original form, but can still be used for verification.

Table 1. Hash Value Result

Nama File	Nilai Hash
EHR_Small_1.pdf	39ee79235f573a62cc182793b1d21485d6a15e234965ee6318364e98a54624db
EHR_Small_2.pdf	d4fce517aa6a820027e141615d43eed165c3c93d60d734ec13e1c866fed351b4
EHR_Small_3.pdf	84aa641bd87f974459b49bb15dc031d70b22b36d65703c450eba333426f59312
EHR_Small_4.pdf	0a099c76781e9cf997287be779a2c085795a6356b6e9bed28393bd252bf51d30
EHR_Small_5.pdf	659babd6b4b62c1340beb3c31aa576745d39b9d89de41e372dadd369864300d0
EHR_Small_6.pdf	0bfd2be362278e33ff35fd57c4b0b7132355e5fe4c64f75031d42bf48546476b
EHR_Small_7.pdf	7d627c526a4eacae25a019e06ff2c81d59340719330f89b7f6c5f35776926897
EHR_Small_8.pdf	5ab282cf02e1577b7dca3feff270b558f458f2bef295f440200bbaafc8da9c19
EHR_Small_9.pdf	33201b2894a43c955be44349c01d035cc40a94d2c8bc8702cb4a5c5c0107bfb1
EHR_Small_10.pdf	12a8a7b0181f17b6d00b2b760deb281e620857b79b1a2c680f5dc2407e14fce3
EHR_Medium_1.pdf	32c0f671e42a4bd6c9b711d7706b09d1da671c460daafe098cbe700cf4d152ff
EHR_Medium_2.pdf	c6defb6f45e1bdafbc973f65aaa5f9f9ae4b07169b16d0ab52eaf3a849801c77
EHR_Medium_3.pdf	e88bddf408a22469dbf9bf06b50cd5326cd41633f74b3b53f520fe1099e02da6
EHR_Medium_4.pdf	52d7a00b5b2be1739449356eab4aa0d1b324a538fdf7c0fa88bd60d9b48933dc
EHR_Medium_5.pdf	ddfd60daf293ceef0dd3a6d844c0c11fd737b76a785dd7fcf48fd35e7da10a70
EHR_Medium_6.pdf	42cc1617c5a076f5d4bc1cb65b4616cdc0d29805eb75c0550227f77a52022b2d
EHR_Medium_7.pdf	c9d050afcfc709c3a945621b00296b7192855eb5fd94d7cf2c9c28f5782a1fa0
EHR_Medium_8.pdf	06fbbc63520dace170bf703593ff2b22fdc9ccba878bdcd6487ef2ed5857ad06
EHR_Medium_9.pdf	6d1035a5ace88258c59a7cd97461131263c021149ab124558799efb010edc384
EHR_Medium_10.pdf	4c0b38365097dab1ad66574f9b45314c29aa8f9383ab03cc84244344b299f7d9
EHR_Large_1.pdf	323383abd40806bc34a51ead2ef5ebb450716c3a910e916cd92e0c7ddecfccb7
EHR_Large_2.pdf	081e275869b6f600b2b489003fe97393b0c1ac3109ced7f2b584688c56fdafa4
EHR_Large_3.pdf	e75425ac3bb4a204d43a00210c380407e8bc6a4521d03c39de24695900ac031c
EHR_Large_4.pdf	87fcf63a86e779f1c430d972251ddbbbe3c3a7df1dfd8770d6bd2e816d73012f
EHR_Large_5.pdf	48767310fa70f2ca7261bd3978ef015513bfe2752d98ad6a9e206cd254fcd266
EHR_Large_6.pdf	5e6feb9d8416dafa1b4abb77635104150bad4967418af0665a97a3b5ec50c83a
EHR_Large_7.pdf	758a1f069ba2a371ea8bc1a802817cdbdb656f5db020bc1f38dbe48664ebeb49
EHR_Large_8.pdf	e5268d4fe19bcfd07c03098210905f752b0233808c81c1b3b7451d2a14ff14ee
EHR_Large_9.pdf	5e5358c5db44dd91d36916a94a5f13f6a1f4e0770019bca2fc25ed44772da638
EHR_Large_10.pdf	8f38443ef8387078358f15fa833a926dc6f67cd9fe1dde4136e47405678ad3df

The resulting hash value is then used as metadata stored into the blockchain via a transaction logging mechanism. Storage is done through a distributed network, so that the data that has been recorded is immutable and can be verified for authenticity whenever needed. Once the logging process is successful, the system generates a hash of the transaction as proof that the data has been permanently recorded on the blockchain.

Table 2. Hash Transaction Results

Nama File	Nilai Hash
EHR_Small_1.pdf	0xd126a4248cbba16a586742d6264b928e9cf175b45c0d2c8f86127362ec14a338
EHR_Small_2.pdf	0x32855d7d307ef68d8ff22c42ddeb3c4deae9e54eed783862c0a81553d5e32a50
EHR_Small_3.pdf	0xac24f1cea8c6d28f8a290bfdac86943a6d75dd96348c76fd95c7567a747409e2
EHR_Small_4.pdf	0xc753b060f700d94aa76645512f5827aae96c54dba04f2b3f07010861669efdff
EHR_Small_5.pdf	0xeda40c22e87e8654287c183413691c1f7975504a40effd76b865c710c2a4d2e1
EHR_Small_6.pdf	0x44c599ea5d71693fc83273f5ae9a374e39694785671dc67c4b6b655a4b938899
EHR_Small_7.pdf	0x21a8a5a2dcf41cb4add3b870d9d01e8b0baca9df57eca9af85cff65955d12fd7
EHR_Small_8.pdf	0x589be49035f9730a0c073773404071a82b7682f5d35fefb308f8b8c2c355d87c
EHR_Small_9.pdf	0x541cf69b49801917cd664b545e956ac43da626bddb7a1f84bbf3ae683818c8cf
EHR_Small_10.pdf	0x52abd935f02e62ce3ad5ae28f705889a5b75289f104eab150e25e0c48e5283d6
EHR_Medium_1.pdf	0x7d18a92b7b090c4ae9e4beba68704c7af652000ce2757846196dce487d546798
EHR_Medium_2.pdf	0xa62be87dea8386641ac91c2f914056d14d88a19a1051a823035bbb32fb6f2f2a
EHR_Medium_3.pdf	0x2a8d27e2499c3e81ba0266fadc514db53f7afef1e4988bb15d8ef2715717f4b6
EHR_Medium_4.pdf	0x4661478bc00e60cf4cf1b0982e734cb11cc6412699bbbfcdbb097e61066a18d8
EHR_Medium_5.pdf	0x7f3f7d494960681b3bf32bd843f11d0c96f5c47b045639be99f28fe827b10ef9
EHR_Medium_6.pdf	0xb7558df2f0f06e039e2d62fedf604aed77a65cd08d76ba6a55355c86d80f0b3c
EHR_Medium_7.pdf	0x9d48d40aca417fdbef7bb47b5c8683eee8a916631d6f6b80d15bff5931e950ec
EHR_Medium_8.pdf	0x44b464230dc53247815b34cf4d69877b7708c1b97ed516d1783ea7874d5d0d7d
EHR_Medium_9.pdf	0xab1ee947410b33214597eb8ef2ceba07dbfae313f545b872ebb5242fcfe1f06f
EHR_Medium_10.pdf	0x3877d52b1a471ea37c3e9b907d34e41314bb69d673ce97337f2ac55489347650
EHR_Large_1.pdf	0x4f80b6c603a779de6902ee5322052ebcd766d1e825b996744c181f051d0ae852
EHR_Large_2.pdf	0x5df1595e56594de49495169da87542c968dc9bea9283831a7cf2fb88970ba8e4
EHR_Large_3.pdf	0xd1abc765b95d0f08529335ac666cd8ff8ca659a1ba9083435183af29333fa5ef
EHR_Large_4.pdf	0xb4ec4b401d28996a74e80b8d8f251e7b339ae6794f61abd886469c16434be5df
EHR_Large_5.pdf	0x01f7da574f07c7ee3382b2595c3bedef462645b2cc3a19c779e08989d3dbe0b4
EHR_Large_6.pdf	0x909520089d5f90ab2f1c14cf0827850ccbf3469c0a62313b0386da2ad29a7954
EHR_Large_7.pdf	0xdc3baa3b91fb69e8159717c096f58f521ff72f8142d6bf99034f49273c854add
EHR_Large_8.pdf	0x2b35a7d6815c21f335098664b98fffff19e6d962f690c7535625fbe7f92ee119
EHR_Large_9.pdf	0x843c01c9d924fe12aee584d198b977469866c942ce72c1af6659c6474cc727f6
EHR_Large_10.pdf	0xc04f2c923faa3f0377eff37358d6219c103d0bcc23da59d9509d81ddcfa7bd60

Avalanche Effect Testing

The Avalanche Effect test is performed as an evaluation of the diffusion rate in the AES-256 algorithm by comparing the hexadecimal representation between plaintext and ciphertext, the number of bit changes is calculated to measure the extent to which the encryption process is able to produce significant differences between the original data and the encrypted data, so as to show the quality of the spread.

Table 3. Avalanche Effect Measurement Results

Filename	Filesize (KB)	Avalanche Effect Value %
EHR_Small_1.pdf	28,8	49,986
EHR_Small_2.pdf	28,54	50,065
EHR_Small_3.pdf	28,54	49,945
EHR_Small_4.pdf	28,81	49,960
EHR_Small_5.pdf	28,61	49,951
EHR_Small_6.pdf	28,66	49,874
EHR_Small_7.pdf	28,49	49,971
EHR_Small_8.pdf	28,49	49,978
EHR_Small_9.pdf	28,73	50,136
EHR_Small_10.pdf	28,72	50,025
EHR_Medium_1.pdf	266,73	50,015
EHR_Medium_2.pdf	258,49	50,013
EHR_Medium_3.pdf	260,82	50,008
EHR_Medium_4.pdf	249,82	49,986
EHR_Medium_5.pdf	250,35	49,981
EHR_Medium_6.pdf	262,84	49,988
EHR_Medium_7.pdf	232,63	49,956
EHR_Medium_8.pdf	262,83	49,965
EHR_Medium_9.pdf	247,12	50,004
EHR_Medium_10.pdf	211,87	49,965
EHR_Large_1.pdf	450,62	50,030
EHR_Large_2.pdf	453,67	50,031
EHR_Large_3.pdf	480,98	49,960
EHR_Large_4.pdf	447,05	49,999
EHR_Large_5.pdf	446,89	49,982
EHR_Large_6.pdf	455,64	49,984
EHR_Large_7.pdf	471,78	50,022
EHR_Large_8.pdf	455,76	49,982
EHR_Large_9.pdf	459,47	49,999
EHR_Large_10.pdf	469,2	49,990

Table 3, shows the results of avalanche effect measurements of PDF documents with three categories of file sizes. Each file is tested to determine the percentage of bit changes between plaintext and ciphertext after going through the encryption process using the AES-256 mode CFB algorithm, the value obtained shows how much change occurs due to the encryption process, so it can be used as an indicator of the diffusion rate of the algorithm before the analysis of the average value.

Table 4. Average Avalanche Effect

Filesize (KB)	Average
≤ 50	49,989
50-200	49,988
≥ 200	49,998

Based on the average avalanche effect results in Table 4, documents with a file size of less than 50 KB obtain an average value of 49.989, then files with a size of 50-200 KB obtain an average value of 49.988, while documents with a size of more than 200 KB obtain an average value of 49.998. Very small differences in values indicate that the file size has no effect on the diffusion rate of the AES-256 algorithm. The avalanche effect value on either category remains somewhere around 50%, this value indicates that a small change in the input data can produce a slight change in the ciphertext evenly.

Entropy Testing

Entropy testing is carried out to determine the level of randomness of the encrypted data, so as to ensure that the resulting ciphertext has good security quality. Calculation of entropy value using Shannon Entropy method, where each byte in hexadecimal representation is calculated the frequency of occurrence, then the probability of each byte is obtained by comparing the number of byte occurrences to the total length of the ciphertext.

Table 5. Entropy Test Results

Filename	Filesize (KB)	Entropi Value
EHR_Small_1.pdf	28,8	7,9938
EHR_Small_2.pdf	28,54	7,9924
EHR_Small_3.pdf	28,54	7,9932
EHR_Small_4.pdf	28,81	7,9941
EHR_Small_5.pdf	28,61	7,994
EHR_Small_6.pdf	28,66	7,9928
EHR_Small_7.pdf	28,49	7,9929
EHR_Small_8.pdf	28,49	7,9942
EHR_Small_9.pdf	28,73	7,9941
EHR_Small_10.pdf	28,72	7,9935
EHR_Medium_1.pdf	266,73	7,9993
EHR_Medium_2.pdf	258,49	7,9994
EHR_Medium_3.pdf	260,82	7,9993
EHR_Medium_4.pdf	249,82	7,9992
EHR_Medium_5.pdf	250,35	7,9993
EHR_Medium_6.pdf	262,84	7,9994
EHR_Medium_7.pdf	232,63	7,9991
EHR_Medium_8.pdf	262,83	7,9992
EHR_Medium_9.pdf	247,12	7,9992
EHR_Medium_10.pdf	211,87	7,999
EHR_Large_1.pdf	450,62	7,9996
EHR_Large_2.pdf	453,67	7,9995
EHR_Large_3.pdf	480,98	7,9996
EHR_Large_4.pdf	447,05	7,9996
EHR_Large_5.pdf	446,89	7,9996
EHR_Large_6.pdf	455,64	7,9995
EHR_Large_7.pdf	471,78	7,9996
EHR_Large_8.pdf	455,76	7,9996
EHR_Large_9.pdf	459,47	7,9996
EHR_Large_10.pdf	469,2	7,9996

Based on the results of entropy testing, it can be seen from Table 5 that all files that have gone through the encryption process using AES-256 produce a very high entropy value, which is close to the maximum value of 8 bits.

Table 6. Average Entropy Value

Filesize (KB)	Average (bits)
≤ 50	7,9935
50-200	7,99924
≥ 200	7,99958

From the results of the average entropy value obtained, it can be seen that all file size categories have a value that is very close to the maximum value of 8 bits. in small files, the average value of 7.9935 bits indicates a very high level of randomness, the average value increases in medium-sized files to 7.99924 bits and approaches the maximum value in large files with an average value of 7.99958 bits. An increase in the entropy value of the file as the file size increases indicates that the larger the data is encrypted, the distribution of ciphertext bits becomes more even and random. This shows that the AES-256 algorithm is able to produce an optimal level of randomness so that it has a very good level of security because it produces ciphertext that is difficult to analyze and predict.

Encryption and Blockchain Runtime Testing

Process time testing is carried out to evaluate the performance of the system based on two aspects, namely the duration needed in the document encryption process, as well as the time required to store the hash value into the blockchain network.

Table 7. Runtime Testing Results

Filename	Filesize (KB)	Encryption Time (ms)	Blockhain Time (ms)
EHR_Small_1.pdf	28,8	1,81	1810,92
EHR_Small_2.pdf	28,54	0,11	956,44
EHR_Small_3.pdf	28,54	0,93	1016,41
EHR_Small_4.pdf	28,81	0,14	985,86
EHR_Small_5.pdf	28,61	0,15	984,03
EHR_Small_6.pdf	28,66	0,09	985,69
EHR_Small_7.pdf	28,49	0,14	955,54
EHR_Small_8.pdf	28,49	0,15	967,69
EHR_Small_9.pdf	28,73	0,13	990,13
EHR_Small_10.pdf	28,72	0,13	965,91
EHR_Medium_1.pdf	266,73	2,38	1329,80
EHR_Medium_2.pdf	258,49	0,87	736,07
EHR_Medium_3.pdf	260,82	0,59	702,40
EHR_Medium_4.pdf	249,82	0,44	708,36
EHR_Medium_5.pdf	250,35	0,63	721,13
EHR_Medium_6.pdf	262,84	0,75	718,07
EHR_Medium_7.pdf	232,63	0,70	1075,62
EHR_Medium_8.pdf	262,83	0,57	734,19
EHR_Medium_9.pdf	247,12	0,54	696,04
EHR_Medium_10.pdf	211,87	0,55	689,33
EHR_Large_1.pdf	450,62	1,37	968,08
EHR_Large_2.pdf	453,67	1,25	868,47
EHR_Large_3.pdf	480,98	1,38	939,91
EHR_Large_4.pdf	447,05	1,27	937,93
EHR_Large_5.pdf	446,89	1,31	870,38
EHR_Large_6.pdf	455,64	3,06	948,75
EHR_Large_7.pdf	471,78	2,09	929,97
EHR_Large_8.pdf	455,76	1,41	949,54
EHR_Large_9.pdf	459,47	1,45	943,52
EHR_Large_10.pdf	469,2	1,65	927,81

Based on the results of the processing time test in Table 7, data on the time required for the encryption process and blockchain process in the test dataset is obtained. The resulting encryption time is in the millisecond (ms) range with value variations in each file, both small, medium and large sizes. Meanwhile, blockchain processing time has a greater value than encryption time. Differences in the value of time in each category indicates a variation in system performance that is influenced by the size of the file and the process performed, the following analysis of the average calculation on each file.

Table 8. Average runtime

Filesize (KB)	Encryption Time	Blockchain Time
≤ 50	0,378	1061,862
50-200	0,802	811,101
≥ 200	1,624	928,436

Based on the results of the average runtime, it can be seen that the encryption time increases as the file size increases. In small files obtained an average encryption time of 0.378 ms, then increased in medium-sized files with an average time obtained to 0.802 ms and again increased in large files to 1.624 ms. This shows the relationship between the size of the data with the time required in the encryption process. Meanwhile, the average blockchain storage processing time shows a different pattern, on small files it takes 1061.862 ms, then on medium files it takes 811.101 ms and increases on large files to 928.439 ms. The variation shows that blockchain storage process time is not only affected by file size, but also influenced by other processes such as transaction recording and validation mechanisms.

Testing Result Metrics

The metric Diagram in Figure 5 shows the test results of 4 main parameters, namely the avalanche effect value, entropy value, and encryption process time and blockchain storage time on each test file.

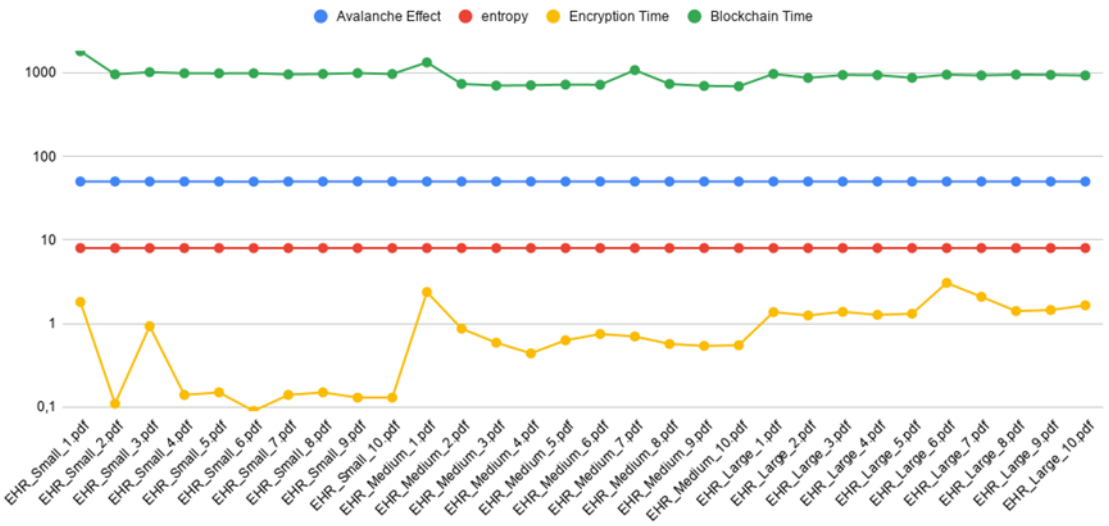

Figure 5. Metric diagram of measurement results

Avalanche effect value looks relatively stable in all test data, showing that small changes in input result in significant changes in ciphertext output, so that the encryption algorithm used has a good diffusion rate. The entropy value also shows stability and is at a value close to the maximum value, so it shows that the encryption results have a good level of randomness. Meanwhile, at the time of the encryption process showed a more volatile variation, especially in file size differences, although it remained in a relatively small span of time. For the blockchain storage process, time has a much higher value than encryption time, but tends to be stable throughout the test data, it shows that the blockchain storage process has more consistent time characteristics even though it involves more complex mechanisms.

CONCLUSION

Based on the results of the tests carried out, it can be concluded that the implementation of digital document security that combines the AES-256 algorithm with blockchain technology has gone well. The AES-256 algorithm is able to perform the encryption process of digital documents in PDF format optimally, which is proven through avalanche effect testing with values close to 50% and entropy values that reach the maximum value limit. The results show that the resulting ciphertext has a high level of diffusion and randomness, so that the ability of the AES-256 algorithm in maintaining the confidentiality of documents is very strong. The use of blockchain technology as a medium for storing metadata hashes from ciphertext successfully ensures the integrity and authenticity of digital documents. The process of recording hashes on the blockchain produces transaction hashes that serve as proof of data validation that is transparent and cannot be changed. In terms of performance, the results of the process time test show that the encryption process is very fast, while the data storage time to the blockchain is still quite efficient and feasible to be applied to securing digital documents.

As for the next study, it is recommended to compare other AES operating modes such as CBC and GCM using the same test parameters, such as avalanche effect, entropy, and runtime, so that the analysis results become more objective and measurable. In addition, system testing can be extended to network environments with a larger number of nodes in order to more realistically evaluate blockchain performance. The implementation of off-chain storage mechanisms can also be considered to improve storage efficiency, as well as carry out advanced security tests, such as brute force or known-plaintext attack simulations, to assess the level of resilience of the system in greater depth.

REFERENCES

[1] R. Baktimalis, I. Teguh Santoso, A. Oktafianto, S. Operasi Laut, S. Staf dan Komando Angkatan Laut, and J. Selatan, “Pengamanan Dokumen Rahasia Di Lingkungan Angkatan Laut Melalui

Sistem Administrasi Digital Terenkripsi." [Online]. Available: http://jurnalseskoal.id/index.php/jikk/index
[2] Moh. A. R. Akbar Ridho1*, "Sistem Pengamanan Dokumen Menggunakanalgoritma Kriptografi Advanced Encryption Standard (AES-256)," vol. 6, Oct. 2024.
[3] A. Farissi, A. Pradata, and K. Miraswan, "Securing Messages Using AES Algorithm and Blockchain Technology on Mobile Devices," *SinkrOn*, vol. 8, no. 2, pp. 1166–1171, May 2023, doi: 10.33395/sinkron.v8i2.12381.
[4] K. Nagamani and R. Monisha, "Physical Layer Security Using Cross Layer Authentication for AES-ECDSA Algorithm," *Procedia Comput. Sci.*, vol. 215, pp. 380–392, 2022, doi: 10.1016/j.procs.2022.12.040.
[5] D. Shivaramakrishna and M. Nagaratna, "A novel hybrid cryptographic framework for secure data storage in cloud computing: Integrating AES-OTP and RSA with adaptive key management and Time-Limited access control," *Alexandria Engineering Journal*, vol. 84, no. October, pp. 275–284, 2023, doi: 10.1016/j.aej.2023.10.054.
[6] P. Satyanarayana, N. Sriramdas, B. Madhavi, M. Arun, N. V. Phani Sai Kumar, and V. Gokula Krishnan, "Enhancement of Security in IoT Using Modified AES Algorithm for IoT Applications," *International Conference on Sustainable Communication Networks and Application, ICSCNA 2023 - Proceedings*, no. March, pp. 380–386, 2023, doi: 10.1109/ICSCNA58489.2023.10370606.
[7] P. Studi Sistem dan Teknologi Informasi, "Implementasi Teknologi Blockchain dan Algoritma AES 256 dalam Pengelolaan Rekam Medis Elektronik Alya Apriliyanti 18220050 (Author)."
[8] P. Bagane and S. Kotrappa, "Enriching aes through the key generation from genetic algorithm," *Indian Journal of Computer Science and Engineering*, vol. 12, no. 4, pp. 955–963, 2021, doi: 10.21817/indjcse/2021/v12i4/211204141.
[9] S. Restu Aji and W. Trisari Harsanti Putri, "Implementasi Teknologi Blockchain dalam Aplikasi E-Voting Berbasis Mobile," *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 14, no. 2, Dec. 2023, doi: 10.31849/digitalzone.v14i2.16682.
[10] F. Ullah *et al.*, "Blockchain-enabled EHR access auditing: Enhancing healthcare data security," *Heliyon*, vol. 10, no. 16, p. e34407, 2024, doi: 10.1016/j.heliyon.2024.e34407.
[11] T. Wira and E. Suryawijaya, "Memperkuat Keamanan Data melalui Teknologi Blockchain: Mengeksplorasi Implementasi Sukses dalam Transformasi Digital di Indonesia Strengthening Data Security through Blockchain Technology: Exploring Successful Implementations in Digital Transformation in Indonesia," vol. 2, no. 1, pp. 55–67, 2023, doi: 10.21787/jskp.2.2023.55-67.
[12] K. V. Deshpande *et al.*, "Blockchain-Based Decentralized Asset Sharing," *International Conference on Sustainable Computing and Smart Systems, ICSCSS 2023 - Proceedings*, vol. 7, no. 41, pp. 1–5, 2024, doi: 10.3390/app12189377.
[13] Shibghotullah Al Murod and Suhirman, "Aplikasi Keamanan E-Voting Pemilihan Ketua Osis Menggunakan Metode AES 128 Berbasis Android (Studi Kasus: MTSN 3 Poso)," *Decode: Jurnal Pendidikan Teknologi Informasi*, vol. 4, no. 3, pp. 1142–1154, Nov. 2024, doi: 10.51454/decode.v4i3.818.
[14] P. D. Machvira Ul Husnaa, "Implementasi Blockchain Untuk Optimalisasi Sistem Keamanan Dokumen," *Jurnal IlmiahTeknik Mesin, Elektro dan Komputer*, Nov. 2021.
[15] J. Syahputra Sianipar, N. Budi Nuugroho, I. Mariami, S. Informasi, and S. Triguna Dharma, "Pengamanan Data Gaji Karyawan Dengan Menggunakan Metode Advanced Encryption Standard (AES)," vol. 3, no. 1, pp. 35–45, 2024, [Online]. Available: https://ojs.trigunadharma.ac.id/index.php/jsi
[16] E. S. Marsiani *et al.*, "Implementasi Sistem Keamanan Aes 256-Bit Gcm Guna Mengamankan Data Pribadi."
[17] M. Ristic, B. Noack, and U. D. Hanebeck, "Cryptographically Privileged State Estimation with Gaussian Keystreams," *IEEE Control Syst. Lett.*, vol. 6, pp. 602–607, 2022, doi: 10.1109/LCSYS.2021.3084405.
[18] K. Kristanto, M. H. Nurjamil, K. Noppi, A. Jaya, S. Kom, and J. Jalianery, "Transformasi Hukum Dalam Era Revolusi Teknologi Blockchain."
[19] M. Oka Augusta, C. Putriana Oktaviandi Syeira, A. Hadiapurwa, and K. Kunci, "Penggunaan Teknologi Blockchain Dalam Bidang Pendidikan," vol. 437, no. 2, 2021, [Online]. Available: https://digitalcredentials.mit.edu
[20] R. Bhan, R. Pamula, P. Faruki, and J. Gajrani, *Blockchain-enabled secure and efficient data sharing scheme for trust management in healthcare smartphone network*, vol. 79, no. 14. Springer US, 2023. doi: 10.1007/s11227-023-05272-6.

[21] A. Nanda Sari and T. Gelar, "Blockchain: Teknologi Dan Implementasinya," 2024.
[22] J. Yong, X. Lei, Z. Huang, J. Dang, and Y. Wang, "A Blockchain-Based Supervision Data Security Sharing Framework," *Applied Sciences (Switzerland)*, vol. 14, no. 16, 2024, doi: 10.3390/app14167034.
[23] R. Mustaqim Handoko *et al.*, "Implementasi Blockchain Untuk Keamanan Sistem Pembayaran Digital dan Optimasi Transaksi Keuangan (Studi Kasus Industri Fintech di Indonesia)," *Jurnal Ilmu Teknik dan Informatika*, vol. 4, pp. 64–74, doi: 10.51903/teknik.
[24] E. Yogiyanti and I. M. Suartana, "Penerapan Teknologi Blockchain pada Sistem Laporan Keuangan Aplikasi Point of Sale," *Journal of Informatics and Computer Science*, vol. 06, 2024.
[25] M. Nuzula, Y. Away, K. Kahlil, and A. Novandri, "Optimizing Attendance Data Security by Implementing Dynamic AES-128 Encryption," *Sinkron*, vol. 8, no. 2, pp. 813–823, Mar. 2024, doi: 10.33395/sinkron.v8i2.13412.
[26] N. R. D. P. Astuti, I. Arfiani, and E. Aribowo, "Analysis of the security level of modified CBC algorithm cryptography using avalanche effect," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 674, no. 1, 2019, doi: 10.1088/1757-899X/674/1/012056.
[27] L. Budi Handoko, "Pengujian Avalanche Effect Pada Kriptografi Teks Menggunakan Autokey Cipher," *2 st Proceeding STEKOM*, vol. 2022, 2022.
[28] H. M. Taher, S. Q. A. Al-Rahman, and S. A. Shawkat, "Best S-box amongst differently sized S-boxes based on the avalanche effect in the advance encryption standard algorithm," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 6, pp. 6535–6544, 2022, doi: 10.11591/ijece.v12i6.pp6535-6544.
[29] A. L. F. Rahmawati, A. I. Hadiana, and Melina, "Sistem Data Loss Prevention Untuk Deteksi dan Enkripsi pada Dokumen Menggunakan Regex dan Format Preserving Encryption," *Jurnal Algoritma*, vol. 22, no. 2, Nov. 2025, doi: 10.33364/algoritma/v.22-2.2387.
[30] C. Rohlfsen, K. Shannon, and A. S. Parsons, "Entropy in Clinical Decision-Making: A Narrative Review Through the Lens of Decision Theory," *J. Gen. Intern. Med.*, no. 0123456789, 2025, doi: 10.1007/s11606-025-09868-x.
[31] Hasan Shadzily and Bambang Sujatmiko, "Analisis Tingkat Keamanan File Dokumen Menggunakan Algoritma Advanced Encryption Standard (AES)," *Inovate : Jurnal Ilmiah Inovasi Teknologi Informasi*, vol. 10, no. 1, pp. 37–44, Sep. 2025, doi: 10.33752/inovate.v10i1.9251.