Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 2, April 2024
DOI : https://doi.org/10.33395/sinkron.v8i2.13412
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Optimizing Attendance Data Security by Implementing Dynamic AES-128 Encryption

Mukhsin Nuzula[1)], Yuwaldi Away[2)], Kahlil[3)], Andri Novandri[4)*]
[1,2,3,4)] Universitas Syiah Kuala, Banda Aceh, Indonesia
[1)]mukhsin_n@mhs.usk.ac.id, [2)]yuwaldi@usk.ac.id, [3)]kahlil@usk.ac.id, [4)]andrie.nov11@gmail.com

Submitted: Jan 19, 2024 | **Accepted**: Jan 30, 2024 | **Published**: Apr 1, 2024

Abstract: Data security protection is a crucial aspect when dealing with the transmission of sensitive information over communication networks. This article discusses the Advanced Encryption Standard 128 (AES-128) algorithm as an effective method to enhance data security. AES-128 is an encryption process that utilizes a 128-bit key length. The AES-128 encryption method involves a series of steps, including SubBytes, ShiftRows, MixColumns, and AddRoundKey. This entire process is repeated 10 times. Subsequently, there is a decryption phase that is the reverse of the encryption phase, where each step is inverted to decrypt previously encrypted data. This paper proposes the development of the Dynamic Advanced Encryption Standard 128-bit (Dynamic AES-128) method by developing a type of key that operates dynamically, known as a dynamic key. The dynamic key is a key method that changes every minute, by combining a secret keyword with a timestamp, thereby providing an additional layer of security. The development of this method is used to enhance data security in employee attendance systems, which include UID, Name, Division, Email, Address, In-Time, and Out-Time. Test results indicate that the dynamic AES-128 encryption algorithm demonstrates optimal performance, with successive encryption and decryption speeds of 14656.78 bit/s and 21898.21 bit/s, respectively. Meanwhile, the encryption and decryption process durations are 6.66 ms and 2.44 ms, respectively, with an avalanche effect value of 50.73% and an entropy of 6.67 bit/symbol. This research also surpasses some previous studies, demonstrating its stability in maintaining data security, even in the face of varying data lengths. Therefore, the implementation of the dynamic AES-128 method in attendance systems can provide advantages in addressing current digital-era information security challenges.

Keywords: Encryption, Decryption, AES-128, Dynamic key

INTRODUCTION

In the ever-evolving digital era, information security becomes a crucial aspect to consider, especially when transferring sensitive data through communication networks. To safeguard the confidentiality and integrity of data, cryptography emerges as the primary solution to enhance data security. The main objective of cryptography is to secure data communication from unauthorized access or data modification during the storage process (Vivi Wahdini et al., 2021). One proven effective cryptographic method is the Advanced Encryption Standard with a key length of 128 bits, commonly referred to as AES-128 (Andriani et al., 2018; M. Hidayat et al., 2023; Kaminsky et al., 2010). AES-128 provides robust protection against security threats, making it the preferred choice for securing data in various applications, ranging from online communication to file storage (Prameshwari & Sastra, 2018; Priyanka et al., 2016; Rachmawanto et al., 2017; Syafaat & Finandhita, 2019). AES-128 utilizes a 128-bit key length, thus providing security without compromising computational performance (Cristy & Riandari, 2021; Pammu et al., 2016; Tulloh et al., 2016). AES-128 operates using the same key for both encryption and decryption processes, making it a symmetric cryptographic algorithm. The encryption process in AES-128 involves a series of complex mathematical transformations on data blocks. The algorithm consists of ten rounds, where each round includes cryptographic operations such as SubBytes, ShiftRows, MixColumns, and AddRoundKey (Simbolon et al., 2020). SubBytes are responsible for substituting each byte in the data block with another byte according to a substitution table. ShiftRows shifts the rows within the data block, MixColumns performs linear transformations on the block's columns, and AddRoundKey adds the round key generated from the main key. This entire process is repeated ten times, creating a strong security layer against various types of attacks, including cryptanalysis analysis (A. Hidayat, 2022; Rohman & Ramdhani, 2022; Widyawan & Imelda, 2021). The advantage of AES-128

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 2, April 2024 e-ISSN : 2541-2019
DOI : https://doi.org/10.33395/sinkron.v8i2.13412 p-ISSN : 2541-044X

lies in its ability to provide robust protection against security threats with a balanced trade-off between security and efficiency (Lubis & Fujiati, 2023; M. Fahri H Damanik et al., 2022). The algorithm's versatility allows it to operate on various hardware and software, making it a highly reliable choice for securing online communication via HTTPS protocols, file encryption, as well as storage and exchange of data requiring high-security levels (R. A. Putra et al., 2023; Rachmat & Samsuryadi, 2019).

The AES-128 encryption method serves as a data security measure for online communication, file storage, or sensitive data exchange, with a key length of 128 bits. AES-128 is designed to provide strong protection against various security threats. The advantages of AES-128 include a high security level, good performance, and the ability to operate in various software applications. To demonstrate the security of the encryption method, several measurement methods are used to evaluate the performance of dynamic AES-128 encryption, including encryption/decryption speed, encryption/decryption duration, avalanche effect, and entropy. Avalanche effect is an analysis to assess the extent to which encryption is sensitive to input changes, while entropy is an analysis to evaluate the extent to which encryption can produce unpredictable outputs.

Research on the application of AES-128 has been discussed in various papers, as discussed in (Nuari & Ratama, 2020). The paper proposes the use of encryption-decryption techniques with the AES-128 algorithm in a system. The AES-128 algorithm is chosen because of its resilience against various types of attacks to avoid loss, theft, and reduce the risk of data damage. Then, in the paper by (Aryanto et al., 2023), the issue of data confidentiality, especially when the data is within a computer network connected to other networks, is discussed. In this study, the authors developed a cryptographic application using AES-128 encryption. Furthermore, this application was developed using PHP programming language and MySQL as a web application and database. Next, in the paper (Ravida & Santoso, 2020), the implementation of AES-128 encryption in securing Internet of Things (IoT) systems used to manage hydroponic plants is discussed. Based on the study's results, it is concluded that AES-128 encryption in IoT systems can provide a good level of security and is in line with standards. In this paper, AES-128 encryption with a static key method is used, where the key used is the same and does not change over time. The analysis results show an avalanche effect obtained at 50.37% and entropy at 6.45 bit/symbol. Therefore, to improve encryption performance, innovations from the AES-128 method are needed.

Based on the study, a development of the AES-128 encryption method is proposed to secure employee attendance data to prevent data manipulation. One proposed innovation is the use of dynamic keys. The concept of a dynamic key involves encryption techniques by changing the key periodically every minute. This step is used to enhance the level of data security and make it more difficult for unauthorized parties to decrypt encrypted data. The process of implementing a dynamic key in AES-128 involves scheduled and automatic key changes every minute. These changes are made using the timestamp as a seed to generate a new key. This approach provides additional variation and complexity to the encryption process, which can enhance the system's resilience against cryptanalysis attacks. The benefit of implementing encryption with a dynamic key is to strengthen and maintain the security of attendance data. Employees feel more confident that their personal information is safe from unauthorized access.

The contribution of this paper is a dynamic AES-128 encryption method using a dynamic key that can change every minute. The key utilizes a combination of a secret keyword and a timestamp. With the timestamp combination, the key can change according to the desired time interval. This method is then applied to the application of securing employee attendance data, aiming to prevent data manipulation. The secured data includes UID, Name, Division, Email, Address, In-Time, and Out-Time with varying data lengths.

LITERATURE REVIEW

Encryption and decryption are two key elements in the field of cryptography, a discipline related to information and data security (Muhammad Abdullah, 2017; Wachid Hidayatulloh et al., 2023). Encryption is a process of converting data into an incomprehensible form, known as ciphertext. This process involves the use of an encryption key, a series of numbers or characters, aimed at protecting data from unauthorized access (Suwarni et al., 2023). By using this key, information can be transformed into a format that can only be translated back to its original form by the recipient who possesses the corresponding decryption key. There are two main types of encryption: symmetric and asymmetric. Symmetric encryption uses the same key for both encryption and decryption, while asymmetric encryption involves a pair of keys, where one is used for encryption and the other for decryption (Aprizald et al., 2023; Kamali et al., 2010). Decryption is the process of restoring ciphertext to its original plaintext form using the appropriate key. Only the holder of the decryption key can convert the ciphertext back into understandable information. Similar to encryption, decryption can also be categorized into two main types: symmetric and asymmetric. The success of a cryptographic system depends on the security and complexity level of the encryption used, as well as the ability to maintain the confidentiality of the keys.

AES-128 is a symmetric cryptographic algorithm that operates using the same key for both encryption and decryption processes, making it one of the symmetric cryptography algorithms (Fathy et al., 2012; Heron, 2009). The value 128 refers to the 128-bit length of the key used in the algorithm. The key length affects the level of

*name of corresponding author

higher security (Muttaqin & Rahmadoni, 2020). However, a 128-bit length is considered sufficient for many security applications. The encryption and decryption processes in AES-128 involve a series of complex mathematical transformations on data blocks (Selent, 2014). By iterating through ten rounds, AES-128 can provide a high level of security against various attacks. The stages of AES-128 consist of various mathematical operations such as SubBytes, ShiftRows, MixColumns, and AddRoundKey. The advantage of this algorithm lies in the balanced combination of security and efficiency, as well as its ability to operate on various hardware and software platforms. AES-128 is widely applied in various security applications, such as online communication via HTTPS protocol, file encryption, data storage, and exchange (Garcia, 2015). The cryptographic resilience of AES-128 is proven through various cryptanalysis tests, ensuring strong protection against various attacks (Xinmiao Zhang & Parhi, 2002). Its ability to provide high security levels, and maintain efficiency and speed in encryption and decryption processes, makes it a common and reliable choice in the world of information security implementation (Saraf et al., 2014).

METHOD

Encrypt/Decrypt Method

The AES-128 encryption method involves a series of complex steps to secure data using a 128-bit key. The flowchart of the AES-128 algorithm in the encryption process is illustrated in Fig.1(a). In the first step, plaintext is mapped into blocks of data corresponding to the block length specified by the AES algorithm, which is 128 bits (Fachrozi & Fahmi, 2021). Subsequently, padding is performed to adjust the block length if the plaintext does not meet the desired block size. In the initial step, the AddRoundKey process is executed, followed by the encryption process starting with the SubBytes operation, followed by ShiftRows, which is then followed by the MixColumns operation, and finally, entering the AddRoundKey process.

In the AddRoundKey step, the state matrix, which is a representation of the block of data being processed, is bitwise XOR-operated with the round key. This round key is generated from the main encryption key, which is derived for each round. In the SubBytes step, each element in the state matrix is replaced by a corresponding value from a substitution table, known as the substitution box (S-box), shown in Fig. 2 (Randi et al., 2020). This S-box replaces each byte in the state matrix with a new byte, depending on the original hexadecimal value of that byte. In the ShiftRows step, each row of the state matrix is horizontally rotated to the left by a certain number of steps. The number of shift steps differs for each row, creating a diffusion effect that helps increase the complexity and security of the algorithm. The first row remains unchanged, the second row is shifted to the left by one step, the third row is shifted to the left by two steps, and the fourth row is shifted to the left by three steps. In the MixColumns step, each column of the state matrix is transformed using a matrix multiplication operation with a fixed matrix called the constant matrix. This constant matrix is chosen in such a way that ensures each column in the state matrix undergoes a strong linear transformation (Azhari et al., 2022; Kurniawan et al., 2018). These steps are repeated ten times, each representing one round of the AES algorithm. In the final round, the MixColumns operation is not performed. The final result of these ten rounds is the encrypted data block, ready to be securely transmitted or stored (Indra, 2023).

The decryption process follows the same steps, but in inverse form, as seen in Fig.1(b). The round keys generated during encryption rounds are rearranged for use in the AddRoundKey stage during decryption. The SubBytes, ShiftRows, and MixColumns processes are also modified specifically to ensure proper decryption (Ferdiansyah, 2021; Y. Putra et al., 2021). All these steps work together to create a strong and effective security layer in maintaining data confidentiality during transit or storage.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 2, April 2024
DOI : https://doi.org/10.33395/sinkron.v8i2.13412
e-ISSN : 2541-2019
p-ISSN : 2541-044X

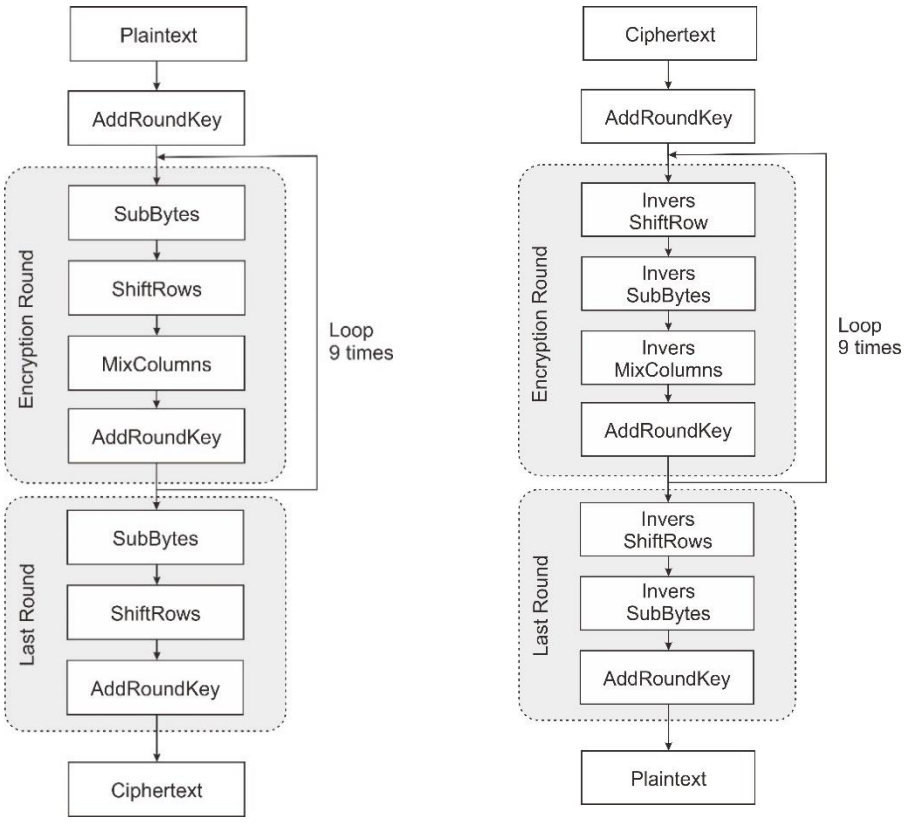


Fig. 1 Flowchart AES-128 Algorithm (a) Encryption Process, (b) Decryption Process

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Fig. 2 Substitution Box

Proposed Method

In this paper, a method of encryption using dynamic keys with the AES-128 algorithm is proposed, where the key can be changed every minute. The key used is a combination of a secret keyword with a timestamp. The value of this timestamp will continuously change over time, making the key dynamic and enhancing the level of security. The details of this dynamic encryption algorithm are illustrated in Fig. 3. This encryption method is specifically designed to secure attendance data, which includes UID, Name, Division, Email, Address, In-Time, and Out-Time. Based on the encryption and decryption process diagrams shown in Fig. 4, data must undergo encryption using the dynamic AES-128 encryption method before transfer. After the data is transferred and received, it must undergo the decryption process, which uses dynamic keys consisting of a combination of a secret keyword and a timestamp. The secret keyword on both sides must be identical. Additionally, to ensure the synchronization of timestamps between the sender and receiver, clock synchronization must be performed on both sides.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 2, April 2024
DOI : https://doi.org/10.33395/sinkron.v8i2.13412
e-ISSN : 2541-2019
p-ISSN : 2541-044X

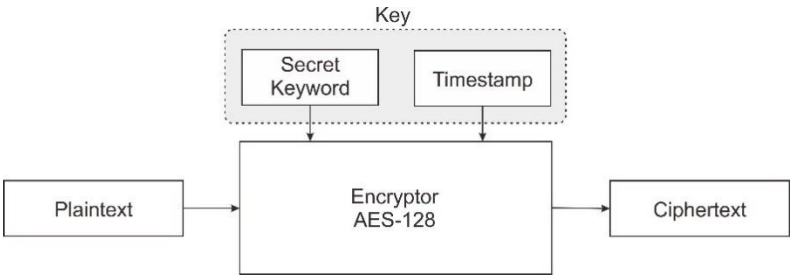


Fig. 3 AES-128 Dynamic Encryption Process Diagram

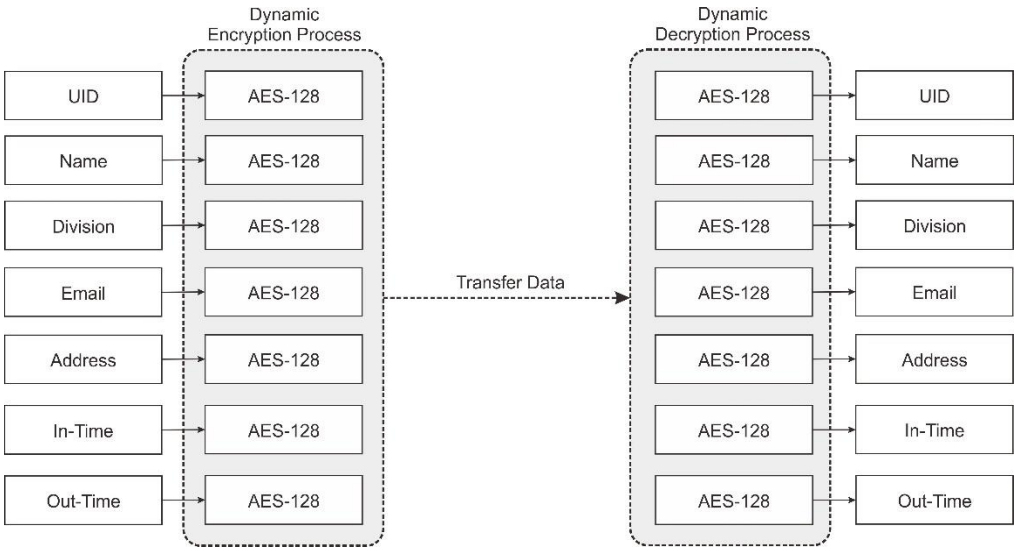


Fig. 4 The Encryption and Decryption Process for Attendance Data

Analysis Method

The analysis process was conducted to determine the performance level of the proposed method. The analysis includes measuring the encryption/decryption speed, measuring the encryption/decryption duration, calculating the avalanche effect, and entropy. Speed measurement was carried out to assess the system's performance in executing the encryption/decryption process using the dynamic key method with the AES-128 algorithm, while duration measurement was to determine the time required for the encryption/decryption process. The avalanche effect is a value that measures how small changes in the plaintext will result in significant changes in the ciphertext. The avalanche effect yields values between 0% and 100%. The higher the avalanche effect value, the better the cryptographic algorithm's avalanche effect, thus enhancing the algorithm's security (Abikoye et al., 2019; Adlani & Putra, 2022). The general formula for calculating the avalanche effect is as follows,

$$AE = \frac{N_d - N_s}{N_d + N_s} \times 100\% \qquad (1)$$

where, AE is the avalanche effect in percentage, N_d is the number of differing bits between two ciphertexts when one input bit is changed, and N_s is the length of the plaintext.

Entropy is a measure of uncertainty or randomness in data. Its use aims to measure the extent to which a key can provide security (Ravida & Santoso, 2020). The general formula for calculating entropy is as follows,

$$H = -\sum_{i=1}^{n} p_i \,.\log_2(p_i) \qquad (2)$$

where, n is the number of possible values that a variable can take, and p_i is the probability of the i value occurring.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 2, April 2024
DOI : https://doi.org/10.33395/sinkron.v8i2.13412
e-ISSN : 2541-2019
p-ISSN : 2541-044X

RESULT

The testing process involves encrypting data at different times to observe the changes in the key. The secret keyword used is “dynkey”, and the plaintext consists of a variable name, “Muksin”. The key change data can be seen in Table 1. Based on the table, it can be observed that the key changes every minute, resulting in continuous changes in the encryption output. This variation is one of the efforts to enhance the security of attendance data, reduce the risks of cyberattacks, and maintain information confidentiality. The dynamic AES-128 encryption system provides an additional layer of security by ensuring that the key used in the encryption and decryption processes changes continuously according to the specified time, thereby enhancing resilience against attacks.

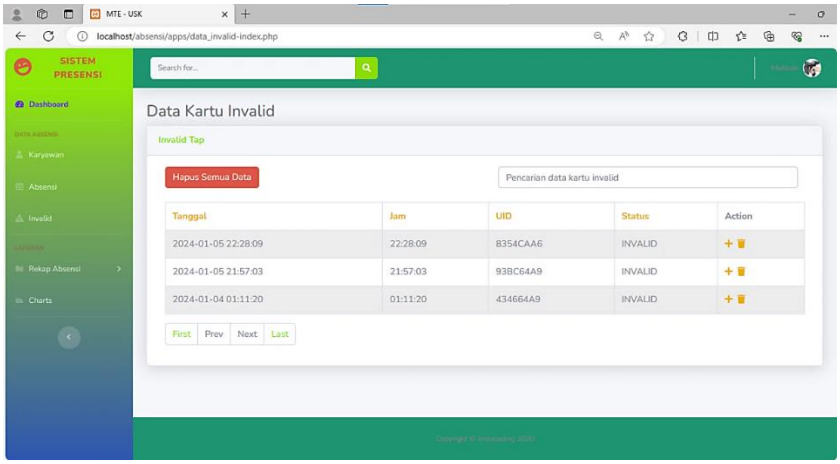


Fig. 5 The attendance database screen display

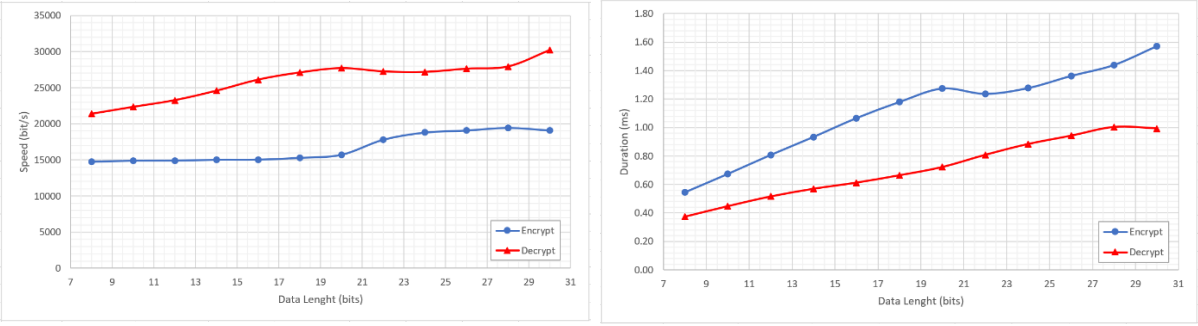
In Table 2, the performance measurement data of the attendance system using the proposed method are presented. This performance data includes the values of encryption and decryption speed, encryption and decryption duration, avalanche effect, and entropy for various variables using the dynamic AES-128 method. Meanwhile, Fig. 5 displays the attendance database application created on the local server. Fig. 6 shows a graph comparing the system’s performance based on the measurement results of encryption and decryption speed and duration against the data length in bits. On the other hand, Fig. 7 illustrates the measurement results of the avalanche effect and entropy against the data length in bits.

Table 1. Dynamic Key AES-128 Change Based on Time

Time	Type	Key	Ciphertext
09:47	Hex	64796e6b65795f323834323537363700	e002057f7253ec7b6b955d59f11a7dcf
	Base64	ZHlua2V5XzI4NDI1NzY3AA==	4AIFf3JT7HtrlV1Z8Rp9zw==
09:48	Hex	64796e6b65795f323834323537363800	ce559a4e11f12cdd32a2c32b8f028d38
	Base64	ZHlua2V5XzI4NDI1NzY4AA==	zlWaThHxLN0yosMrjwKNOA==
09:49	Hex	64796e6b65795f323834323537363900	8cb9afdb4acb9371eae414d88ea02ae4
	Base64	ZHlua2V5XzI4NDI1NzY5AA==	jLmv20rLk3Hq5BTYjqAq5A==
09:50	Hex	64796e6b65795f323834323537373000	1a2d8d0b319b31a07c9e92dc408173b0
	Base64	ZHlua2V5XzI4NDI1NzcwAA==	Gi2NCzGbMaB8npLcQIFzsA==
09:51	Hex	64796e6b65795f323834323537373100	65c1d80a6cd297d07efe84e4758caf7a
	Base64	ZHlua2V5XzI4NDI1NzcxAA==	ZcHYCmzSl9B+/oTkdYyveg==

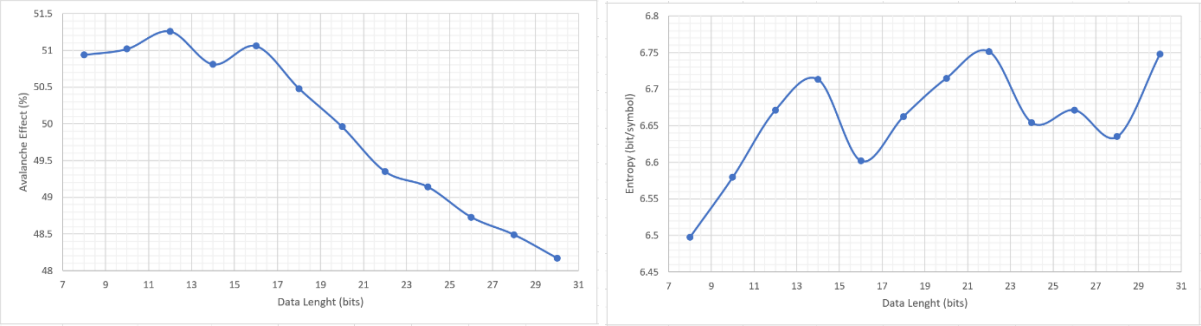
*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 8, Number 2, April 2024
DOI : https://doi.org/10.33395/sinkron.v8i2.13412
e-ISSN : 2541-2019
p-ISSN : 2541-044X



(a) (b)

Fig. 6 The Comparison Encryption and Decryption Results Based on Data Length (a) Speed, (b) Duration



(a) (b)

Fig. 7 The Result Based on Data Length (a) Avalanche Effect, (b) Entropy

Table 2. The Result Data using Dynamic AES-128

Variable	Data Length (bits)	Speed		Duration		Avalanche Effect (%)	Entropy (bit/symbol)
		Encrypt (bit/s)	Decrypt (bit/s)	Encrypt (ms)	Decrypt (ms)		
UID	8	14961.58	22366.37	4.28	1.53	51.22	6.678
Name	6	14762.36	21773.30	3.25	1.19	51.35	6.699
Division	8	14885.40	22541.51	4.30	1.53	51.32	6.669
Email	20	14715.54	21769.65	10.87	4.00	49.60	6.637
Address	27	14318.14	21836.37	15.09	5.53	49.24	6.659
In-Time	8	14645.61	21716.44	4.37	1.61	51.14	6.717
Out-Time	8	14308.84	21283.80	4.47	1.68	51.23	6.650

DISCUSSIONS

Based on the performance measurement results of encryption and decryption presented in Table 2, it can be concluded that the AES-128 encryption algorithm demonstrates optimal performance. The results obtained for encryption speed range from 14308.84 bit/s to 14961.58 bit/s, while decryption speed ranges from 21283.80 bit/s to 22541.51 bit/s. These results indicate high efficiency in executing the encryption and decryption processes. The results obtained for encryption process duration vary between 3.25 ms and 15.09 ms, while decryption duration ranges from 1.19 ms to 5.53 ms. These results show relatively consistent and constant timing. The results obtained for the avalanche effect range from 49.24% to 51.35%, indicating the algorithm's sensitivity to small changes in input. Meanwhile, higher entropy values for variables with larger data lengths indicate good randomness in the data. The average encryption and decryption speeds are 14656.78 bit/s and 21898.21 bit/s, respectively. Meanwhile, the average encryption and decryption durations are 6.66 ms and 2.44 ms, respectively. The average avalanche effect is 50.73%, and the entropy is 6.67 bit/symbol.

Based on the data on the performance of encryption and decryption with the dynamic AES-128 algorithm, there is an increase in encryption speed as the data length increases. Despite fluctuations, overall, the encryption

*name of corresponding author

speed increases from 14727.23 bit/s at a data length of 8 bits to a peak of 19454.19 bit/s at a data length of 28 bits. Decryption speed also shows a similar increase with increasing data length, reaching its peak at 27933.16 bit/s at a data length of 28 bits. The duration of the encryption and decryption processes shows a steady increase with increasing data length. The avalanche effect shows relatively small variations but remains consistent in the range of 48.49% to 51.26%. This indicates the algorithm's stability to input changes. Meanwhile, entropy shows a tendency to increase with larger data lengths. Overall, the data indicates that the AES-128 algorithm remains efficient and consistent in maintaining data security, despite facing variations in data length. These characteristics can assist in adjusting encryption parameters to optimize performance according to system needs.

Table 3. The Comparison of Results with Other Studies

Papers	Avalanche Effect (%)	Entropy (bit/symbol)
(Adlani & Putra, 2022)	48.6	-
(Abikoye et al., 2019)	49.97	-
(De Los Reyes et al., 2019)	24.31	-
(Ravida & Santoso, 2020)	50.37	6.45
Proposed Method	50.73	6.67

The comparison data with other studies is presented in Table 3. From the experimental results, the proposed method in this study successfully achieved an avalanche effect rate of 50.73% and an entropy of 6.67 bit/symbol. These results confirm that the proposed method successfully achieved a good avalanche effect and a high level of uncertainty in the data encryption process, outperforming several previous studies. This superiority proves that the proposed method can have a significant impact on the output changes when the input changes and it produces the desired level of uncertainty in the context of data security. The main finding from the evaluation of this proposed method shows an increase in the avalanche effect by 0.36% and entropy by 3.41% compared to the previous study results (Ravida & Santoso, 2020).

CONCLUSION

The dynamic AES-128 encryption system developed in this research provides an additional layer of security by implementing key changes in each encryption and decryption process. This dynamic algorithm method enhances data security by periodically changing the AES-128 keys. Key replacement at regular intervals can reduce the risk of data leakage. Performance evaluation shows optimal results, with average encryption and decryption speeds reaching 14656.78 bit/s and 21898.21 bit/s, respectively. The average encryption and decryption durations are 6.66 ms and 2.44 ms, respectively. The avalanche effect analysis achieves an average value of 50.73%, while the entropy value reaches 6.67 bit/symbol. Encryption and decryption speeds increase with the length of plaintext data. Comparison with previous studies indicates that the proposed method successfully achieves a good avalanche effect and a high level of uncertainty in the data encryption process. With an avalanche effect value of 50.73% and entropy of 6.67 bit/symbol, this method proves superior in achieving the desired level of uncertainty for data security. The main findings indicate an increase in the avalanche effect by 0.36% and entropy by 3.41% from previous research results. With the increase in avalanche effect and entropy from previous research results, this method proves superior in achieving the desired level of uncertainty for data security.

One weakness of this study is that if the frequency of key changes is too high, it can affect system performance by increasing the risk of key loss or inaccuracies in synchronization. Although dynamic key provides an additional layer of security, this method is specifically developed for implementation in employee attendance systems. Therefore, additional considerations are needed if this method is applied to other use cases that may have different characteristics or security requirements. This research can be further developed by modifying the algorithm to optimize the performance of dynamic AES-128 encryption, including efforts to improve the speed and efficiency of the encryption and decryption processes.

REFERENCES

Abikoye, O. C., Haruna, A. D., Abubakar, A., Akande, N. O., & Asani, E. O. (2019). Modified Advanced Encryption Standard Algorithm for Information Security. *Symmetry*, *11*(12), 1484. https://doi.org/10.3390/sym11121484

Adlani, M. A., & Putra, R. E. (2022). Pengamanan Mnemonic Phrase Menggunakan Modified Advanced Encryption Standart. *Journal of Informatics and Computer Science (JINACS)*, *3*(04), 425–434. https://doi.org/10.26740/jinacs.v3n04.p425-434

*name of corresponding author

Andriani, R., Wijayanti, S. E., & Wibowo, F. W. (2018). Comparision of AES 128, 192 And 256 Bit Algorithm For Encryption And Description File. *3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE)*, 120–124. https://doi.org/10.1109/ICITISEE.2018.8720983

Aprizald, A., Hasan, M. A., & Setiawan, D. (2023). Aplikasi Keamanan Data Berbasis Web Menggunakan Algoritma AES 128 Untuk Enkripsi Dan Dekripsi Data. *Jurnal Teknik Informatika*, *2*(2), 85–95. https://doi.org/10.58794/jekin.v2i2.225

Aryanto, M. B., Tahir, M., Devita, S. I., Mustofa, Z. N., Ainiyah, Q., & Sundoro, S. (2023). Implementasi Enkrip Dan Dekrip File Menggunakan Metode Advance Encryption Standard (AES-128). *Jurnal Ilmiah Sistem Informasi Dan Ilmu Komputer*, *3*(1), 89–104. https://doi.org/doi.org/10.55606/juisik.v3i1.434

Azhari, M., Mulyana, D. I., Perwitosari, F. J., & Ali, F. (2022). Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES). *Jurnal Pendidikan Sains Dan Komputer*, *2*(1), 163–171. https://doi.org/10.47709/jpsk.v2i01.1390

Cristy, N., & Riandari, F. (2021). Implementasi Metode Advanced Encryption Standard (AES 128 Bit) Untuk Mengamankan Data Keuangan. *Jurnal Ilmu Komputer Dan Sistem Informasi*, *4*(2), 75–85. https://doi.org/doi.org/10.9767/jikomsi.v4i2.181

De Los Reyes, E. M., Sison, A. M., & Medina, R. P. (2019). Modified AES Cipher Round and Key Schedule. *Indonesian Journal of Electrical Engineering and Informatics*, *7*(1), 28–35. https://doi.org/10.11591/ijeei.v7i1.652

Fachrozi, M. F., & Fahmi, H. (2021). Penerapan Metode AES-128 Untuk Pengamanan Data Absensi FingerPrint. *JIKOMSI [Jurnal Ilmu Komputer Dan Sistem Informasi]*, *3*(3), 1–8.

Fathy, A., Tarrad, I. F., Hamed, H. F. A., & Awad, A. I. (2012). Advanced Encryption Standard Algorithm: Issues and Implementation Aspects. In *Communications in Computer and Information Science* (Vol. 322, pp. 516–523). https://doi.org/10.1007/978-3-642-35326-0_51

Ferdiansyah, F. (2021). Penggunaan QR Code Berbasis Kriptografi Advanced Encryption Standard (AES) untuk Administrasi Rekam Medis. *Jurnal Syntax Admiration*, *2*(10), 1870–1884. https://doi.org/10.46799/jsa.v2i10.325

Garcia, D. F. (2015). Performance Evaluation of Advanced Encryption Standard Algorithm. *Second International Conference on Mathematics and Computers in Sciences and in Industry (MCSI)*, 247–252. https://doi.org/10.1109/MCSI.2015.61

Heron, S. (2009). Advanced Encryption Standard (AES). *Network Security*, *2009*(12), 8–12. https://doi.org/10.1016/S1353-4858(10)70006-4

Hidayat, A. (2022). Application of the AES Cryptographic Algorithm for E-mail Encryption and Description. *Jurnal Infokum*, *10*(5), 494–500.

Hidayat, M., Tahir, M., Sukriyadi, A., Sulton, A., A, C. A. S., & F, S. A. (2023). Penerapan Kriptografi Caesar Chiper Dalam Pengamanan Data. *Jurnal Ilmiah Multidisiplin*, *2*(03), 35–41. https://doi.org/10.56127/jukim.v2i03.619

Indra, I. G. (2023). Peningkatan Pengamanan Data File Menggunakan Algoritma Kriptografi AES Dari Serangan Brute Force. *Jurnal Media Informatika*, *4*(2), 102–109. https://doi.org/10.55338/jumin.v4i2.496

Kamali, S. H., Shakerian, R., Hedayati, M., & Rahmani, M. (2010). A New Modified Version of Advanced Encryption Standard Based Algorithm for Image Encryption. *2010 International Conference on Electronics and Information Engineering*, *1*(Iceie), 141–145. https://doi.org/10.1109/ICEIE.2010.5559902

Kaminsky, A., Kurdziel, M., & Radziszowski, S. (2010). An Overview of Cryptanalysis Research for the Advanced Encryption Standard. *The 2010 Military Communications Conference*, 1310–1316. https://doi.org/10.1109/MILCOM.2010.5680130

Kurniawan, D., Afyenni, R., & Hidayat, R. (2018). Implementasi Algoritma AES dalam Mengenkripsi Berkas Terintegrasi dengan Layanan Cloud Storage Berbasis Android. *Prosiding SISFOTEK*, *2*(1), 237–245.

Lubis, I., & Fujiati, F. (2023). Aplikasi Keamanan Data Program Acara TV Pada TVRI Menggunakan Metode

*name of corresponding author

AES. *Jurnal Salome: Multidisipliner Keilmuan*, *1*(2), 42–56.

M. Fahri H Damanik, Indra Gunawan, Zulaini Masruro Nasution, Sumarno, & Ika Okta Kirana. (2022). Pemanfaatan Algoritma AES Untuk Keamanann Data Karyawan PT. Telkom Indonesia Pematangsiantar. *STORAGE: Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *1*(1), 32–37. https://doi.org/10.55123/storage.v1i1.157

Muhammad Abdullah, A. (2017). Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data. *Cryptography and Network Security*.

Muttaqin, K., & Rahmadoni, J. (2020). Analysis And Design of File Security System AES (Advanced Encryption Standard) Cryptography Based. *Journal of Applied Engineering and Technological Science (JAETS)*, *1*(2), 113–123. https://doi.org/10.37385/jaets.v1i2.78

Nuari, R., & Ratama, N. (2020). Implementasi Algoritma Kriptografi AES (Advanced Encryption Standard) 128 Bit Untuk Pengamanan Dokumen Shipping. *Journal Of Artificial Intelligence And Innovative Applications*, *1*(2), 2716–1501.

Pammu, A. A., Chong, K.-S., Ho, W.-G., & Gwee, B.-H. (2016). Interceptive side channel attack on AES-128 wireless communications for IoT applications. *IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, 650–653. https://doi.org/10.1109/APCCAS.2016.7804081

Prameshwari, A., & Sastra, N. P. (2018). Implementasi Algoritma Advanced Encryption Standard (AES) 128 Untuk Enkripsi dan Dekripsi File Dokumen. *Jurnal Eksplora Informatika*, *8*(1), 52. https://doi.org/10.30864/eksplora.v8i1.139

Priyanka, M. P., Prasad, E. L., & Reddy, A. R. (2016). FPGA Implementation of Image Encryption and Decryption using AES 128-Bit Core. *International Conference on Communication and Electronics Systems (ICCES)*, 1–5. https://doi.org/10.1109/CESYS.2016.7889929

Putra, R. A., Yupianti, & R, E. P. (2023). Aplikasi Enkripsi Dan Dekripsi Pesan Teks Berbasis Android Menggunakan Algoritma Advanced Encryption Standard. *Jurnal Media Computer Science*, *2*(1), 57–62.

Putra, Y., Yuhandri, Y., & Sumijan, S. (2021). Meningkatkan Keamanan Web Menggunakan Algoritma Advanced Encryption Standard (AES) terhadap Seragan Cross Site Scripting. *Jurnal Sistim Informasi Dan Teknologi*, *3*(2), 56–63. https://doi.org/10.37034/jsisfotek.v3i2.44

Rachmat, N., & Samsuryadi. (2019). Performance Analysis of 256-bit AES Encryption Algorithm on Android Smartphone. *Journal of Physics: Conference Series*, *1196*(1), 1–6. https://doi.org/10.1088/1742-6596/1196/1/012049

Rachmawanto, E. H., Amin, R. S., Setiadi, D. R. I., & Sari, C. A. (2017). A Performance Analysis StegoCrypt Algorithm based on LSB-AES 128 Bit in Various Image Size. *International Seminar on Application for Technology of Information and Communication (ISemantic)*, 16–21. https://doi.org/10.1109/ISEMANTIC.2017.8251836

Randi, A., Lazuardy, K., Chandra, S., & Dharma, A. (2020). Implementasi Algoritma Advanced Encryption Standard pada Aplikasi Chatting berbasis Android. *Jurnal Ilmu Komputer Dan Sistem Informasi*, *3*(2), 1–10. https://doi.org/doi.org/10.9767/jikomsi.v3i1.76

Ravida, R., & Santoso, H. A. (2020). Advanced Encryption Standard (AES) 128 Bit untuk Keamanan Data Internet of Things (IoT) Tanaman Hidroponik. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *4*(6), 1157–1164. https://doi.org/10.29207/resti.v4i6.2478

Rohman, A., & Ramdhani, Y. A. (2022). Implementasi Algoritma AES Untuk Meningkatkan Keamanan Data Karyawan Pada PT PNM Cabang Karawang Barat Menggunakan PHP. *Jurnal Cahaya Mandalika*, *3*(3), 169–174. https://doi.org/doi.org/10.36312/jcm.v3i3.1132

Saraf, K. R., Jagtap, V. P., & Mishra, A. K. (2014). Text and Image Encryption Decryption Using Advanced Encryption Standard. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, *3*(3), 118–126.

Selent, D. (2014). Advanced Encryption Standard. *InSight: Rivier Academic Journal*, *6*(2), 207–223.

*name of corresponding author

Simbolon, I. A. R., Gunawan, I., Kirana, I. O., Dewi, R., & Solikhun, S. (2020). Penerapan Algoritma AES 128-Bit dalam Pengamanan Data Kependudukan pada Dinas Dukcapil Kota Pematangsiantar. *Journal of Computer System and Informatics (JoSYC)*, *1*(2), 54–60.

Suwarni, M., Wahyudi, J., & Khairil, K. (2023). Comparison of the DES Cryptographic Algorithm and the AES Algorithm in Securing Document Files. *Jurnal Media Computer Science*, *2*(1), 41–48. https://doi.org/doi.org/10.37676/jmcs.v2i1.3348

Syafaat, F., & Finandhita, A. (2019). Implementation of AES-128 Cryptography on Unmanned Aerial Vechile and Ground Control System. *Teknik Informatika–Universitas Komputer Indonesia*, 10–19.

Tulloh, A. R., Permanasari, Y., Harahap, E., Matematika, P., Matematika, F., Ilmu, D., & Alam, P. (2016). Kriptografi Advanced Encryption Standard (AES) Untuk Penyandian File Dokumen. *Jurnal Matematika UNISBA*, *2*(1), 1–8.

Vivi Wahdini, S., Hartama, D., Okta Kirana, I., Poningsih, & Sumarno. (2021). Pengamanan Data Pelanggan dan Penjualan Menggunakan Implementasi Algoritma Kriptografi. *Journal of Informatics Management and Information Technology*, *1*(3), 101–107.

Wachid Hidayatulloh, N., Tahir, M., Amalia, H., Afdlolul Basyar, N., Faizal Prianggara, A., & Yasin, M. (2023). Mengenal Advance Encrytion Standard (AES) Sebagai Algoritma Kriptografi Dalam Mengamankan Data. *Digital Transformation Technology (Digitech)*, *3*(1), 1–10.

Widyawan, D., & Imelda, I. (2021). Pengamanan File Menggunakan Kriptografi dengan Metode AES-128 Berbasis Web di Komite Nasional Keselamatan Transportasi. *SKANIKA*, *4*(1), 15–22. https://doi.org/10.36080/skanika.v4i1.2216

Xinmiao Zhang, & Parhi, K. K. (2002). Implementation approaches for the Advanced Encryption Standard Algorithm. *IEEE Circuits and Systems Magazine*, *2*(4), 24–46. https://doi.org/10.1109/MCAS.2002.1173133

*name of corresponding author