Enhancing IoT Healthcare Security: A Lightweight Multi-Layer Cryptographic Approach with AES-256, Grain and HMAC-SHA256

***Eterigho Okpomo Okpu[1], Henry Peter Ovilli[2], Godwin Osakwe Ohumaehuni[3], Emuejevoke Francis Ogbimi[4], Chris Obaro Obruche[5] and Osu Joshua Orove[6]**

[1,3,5]Department of Cyber Security, Southern Delta University, Ozoro, Delta State, Nigeria

[2,4,6]Department of Information Systems & Technology, Southern Delta University, Ozoro, Delta State, Nigeria

***Corresponding Author**

DOI: https://dx.doi.org/10.51584/IJRIAS.2025.10120018

Received: 17 December 2025; Accepted: 24 December 2025; Published: 02 January 2026

ABSTRACT

IoT Healthcare equipment usually encounters variety of hardware constraints, which can influence the encryption techniques used. Several of these IoT Healthcare gadgets use low-power microcontrollers or System-on-Chip (SoC) architectures with limited computing capacity. The devices sometimes have insufficient RAM and flash memory accessible, which can influence the selection of encryption methods. Cryptographic algorithms that are computationally costly or need frequent memory access can quickly deplete the device's battery. This is where a decent encryption method comes into play, because lightweight symmetric-key ciphers like Grain is typically better suited to IoT devices with limited processing power and memory. This study suggested a solution that uses AES 256, Grain and HMAC SHA256 to encrypt and hash IoT healthcare data at the physical layer. AES-256 is typically the quickest of the three, particularly with hardware acceleration. The Grain is the most lightweight and efficient option for tiny data sets and resource-constrained contexts, but HMAC-SHA-256 strikes a reasonable compromise between performance and security, making it a popular choice for message authentication. The combination of these three components results in a comprehensive encryption solution, with AES-256 ensuring strong confidentiality by encrypting the data. Grain offers an extra layer of encryption, increasing total security, while HMAC-SHA256 provides integrity and authenticity, ensuring that the encrypted data has not been tampered with. Using this combination, the encryption result becomes extremely resistant to a variety of assaults, including brute-force, cryptanalysis, and tampering. The system was created using the Python programming language. The results show that combining AES-256, Grain and HMAC-SHA256 speeds up encryption and decryption while using less power.

Keywords: HMAC-SHA-256, IoT data security, Cryptographic algorithms, Grain and AES-256.

INTRODUCTION

According to a 2018 study led by Guy-Cedric and Suchithra, AES-256 (Advanced Encryption Standard) is a strong symmetric-key encryption algorithm that can be used to secure IoT Healthcare data. As per Nirwan et al. (2024), AES-256 algorithm having 256 bits key size makes it suitable for usage in high-security applications such as IoT Healthcare systems that process and transmit sensitive data. AES-256 algorithm can be applied using either hardware or software on IoT Healthcare devices so as to encrypt the data effectively as it gets sent from the Physical to other layers (Shuwandy et al., 2010, Kak, 2015). According to Nizamuddin and Dolly (2024), as they used 256 bits for the key size, brute-force attacks become infeasible. The Grain is essentially a stream cipher method that is intended for lightweight cryptography applications (Marcus et al., 2023; Sabri et al., 2025). Thus, it is a good candidate for use in constrained IoT Healthcare devices. According to Nasera and Naifa (2022), Grain has a small resource footprint and is efficient with respect to CPU and energy consumption. Thus, this algorithm is very suitable for physical layer encryption of IoT Healthcare system.

The low weight of grain makes it good for AES -256. IoT Healthcare might have low-weight restrictions (Luc et al., 2022) in devices. The combination of Grain and AES-256 can offer better security as they make use of two different algorithms to encrypt data making the attack harder. An HMAC (keyed-hash message authentication code) enables us to secure the integrity and authenticity of IoT data. According to Bhaumik et al. (2025), HMAC is a hash-based message authentication code that can function with a hash operation like SHA-256 or SHA-3 (Secure Hash Algorithm) (Chitra & Chelliah, 2025). Taofik et al. (2023) generates a digest (or hash) of encrypted data as well as a secret key. According to Nureni and Onyema (2018), HMAC was employed to confirm that data in IoT Healthcare was unchanged and that the source is trusted. This prevents an adversary from altering the ciphertext without detection, thereby adding another layer of security (Okpu et al., 2024; Faith, 2023). AES-256 will be the best encryption from cryptographic point of view as it is very powerful. For low-weight encryption Grain is provided which has proved its authentication feature. On integrity HMAC is supported to keep the integrity of data. The most useful element is it is of low power usage. Each technology has its advantages, and we can utilize them with the aim of meeting the specific requirements of various IoT systems like resource constraints, high-security etc. will achieve data protection by hybrid cryptography.

Cryptographic techniques can be used with IoT Healthcare devices. Nevertheless, the aforementioned adoption might be affected due to various hardware limitations of IoT Healthcare devices themselves. Most of the IoT Healthcare Gadgets are battery operated or are using low-powered energy sources (Naman et al., 2021). Therefore, power consumption is an essential aspect. Aside from that, any cryptographic algorithm available can guarantee that the key size, intermediate state, or code footprint meets the memory constraint of IoT devices. Lightweight symmetric-key ciphers such as Grain are often better suited for an IoT devices with low computational power. The suggested system physically encrypts assembly IoT health care data using AES 256, Grain and HMAC SHA256. Python was the programming language used to develop the system.

LITERATURE REVIEW

This study sets a fresh standard for security against cloud-stored data crisis through thorough testing and proportionate analysis, revealing that HMAC upgrade cryptographic techniques are increasingly impervious to common threats (Chitra & Chelliah, 2025). Nureni and Onyema (2018) propose and prove the HMAC-SHA256 method with Trust Based System to improve data authentication and integrity. The identified untrusty nodes are completely isolated from the trust worthy nodes. Implementation of HMAC-SHA512 JSON Web Token on the Web Service API with Two Frontend Frameworks (Syabdan et al., 2023) in other words JQuery and Laravel. According to Du and Xie (2024), the researchers addressed the key challenge of data privacy and safety in IoT-centric healthcare systems by offering a hybrid protection system that employs ECDSS, RC4 and SHA-256 to maintain integrity and confidentiality for data transmitted. In an account by Nirwan et al. (2024), ChatCorp platform provides AES-256-CBC for encrypting user conversations which includes texts, phrases and PDF format documents, with encryption execution carried out by server.js and message.js file with assistance of Multer for file uploading.

Research conducted by Vishwasrao et al. (2024) shows that despite the fact that AES and ECC are secure, they have high computational cost and delay. According to the study, encryption trades off with the encryption techniques used in healthcare IoT. Speck-64 and PRESENT have proven to be efficient lightweight ciphers. These ciphers have the fastest performances to date but sacrifices on lesser security. In the study conducted by Muhamad et al. (2024), researchers tested the results of the encryption structure with the payment system application for YAPE SPP. The AES 256 for encoding, decoding and algorithm of SHA256 to used. The AES algorithm is said to be the correct application to shielding any information stored in the system database. The features of Node.js facilitate real-time communication among the users was performed. The complete evaluation of the application based on Node.js is displayed for secure communication using AES 256 encryption (Goel et al., 2024).

According to the research conducted by Nikhil in 2020, the AES algorithm was effective in encrypting and decrypting plain text, files and images. The program openssl was used for testing purpose. The encryption performance improves with the proposed solution The time taken to get encrypted is reduced to 2.05 ms/100 Gbps. Additionally, it has surpassed AES-256, LBC and QSHE in performance (Musthafa et al., 2025). The study in question demonstrates how the file document encryption can be used efficiently. The researchers Okpu and his colleagues in (2025) also have presented a strong security framework that will ensure the integrity, confidentiality and authenticity of data transfer in IoMT. The writers implemented AES-256 encryption and HMAC (Hash-based Message Authentication Code) hashing at the physical level to reduce the risk of cyber attacks and illegal access.

Luc et al. (2022) applied the technique Grain-128AEAD which works on STM32F400, series ARM microcontroller mainly in data processing and transmission interfaces. The prediction and Provable Partitioned Secure Block Chain Principle (PPSBCP) technique were proposed in another study to protect healthcare data sharing (Madhumathi & Vishnu, 2025). In their study, Benhani et al. (2019) look at the possibility of encrypted communication of data between a protected software running on a trusted execution environment (TEE) and the secured logic component of a heterogeneous SoC. Hashim et al. (2020) explored the secured data privacy and confidentiality in an uncertain environment, in multimedia sharing between two IoT hops. Researchers Okpu and their colleagues desired to combine the fuzzy logic system's accuracy of identifying and classifying various types of attacks with the FeedForward neural network technique to train and evaluate the system in IoT health devices (2024).

METHODOLOGY

In a healthcare industry, any hacking or data breach is very expensive and dangerous. To maintain the integrity and secrecy of private data which these IoT Healthcare monitoring devices gather, the data has to be encrypted, decoded and hashed with use of AES 256, Grain and HMAC SHA256 combinedly. The AES-256 with Grain helps to protect the integrity and security of sensitive data in a computationally and energy-efficient way. The hash message authentication code (HMAC) ensures both the authenticity and integrity of data. The flowchart for the proposed system is depicted in figure 1, which gives an idea of what is taking place. The figure states that the system accepts healthcare raw data, encrypts that data using AES 256 and further strengthens that data using lightweight Grain. It also demonstrates how it performs HMAC safeguard along with the secure transportation of the information across the physical layer while maintaining power efficiency at the IoT.

As depicted in Figure 1, the system flowchart is presented by showing the preparation of the IoT healthcare data for processing, which is the entry of the system. IoT sensors generate raw healthcare IoT data. This is unrefined data from glucose meters, heart-rate meters and wearables. The raw data can be sent directly, meaning that if they are not securely sent then they can be attacked. With the help of AES-256 encryption algorithm the proposed system first encrypts the data. This is done by changing normal health care data into unrecognized cipher text. After AES, the cipher text is additionally encrypted with Grain targeting low power devices like IoT devices. This reduces power requirements without sacrificing the required level of security. The end system also creates HMAC (Hash-based Message Authentication Code) with SHA-256. The mechanism guarantees that the recipient can be assured the data was not altered. The physical layer of the IoT device (the wireless medium (Bluetooth, Zigbee or Wi-Fi)) sends this final secure packet.

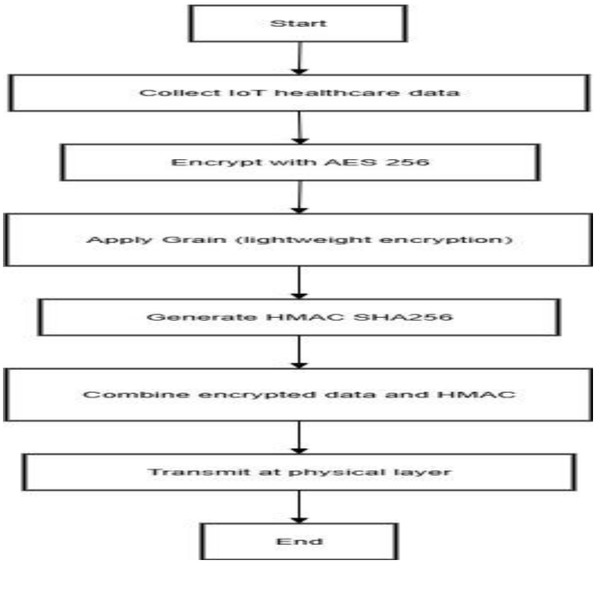
Start
Collect IoT healthcare data
Encrypt with AES 256
Apply Grain (lightweight encryption)
Generate HMAC SHA256
Combine encrypted data and HMAC
Transmit at physical layer
End

Figure 1: Flowchart of the Proposed System

The Algorithm of the proposed system is stated below;

Algorithm

```
1. Key Generation:
i.    Produce a 256-bit AES key.
ii.   Create a 256-bit HMAC key.
iii.  Generate a 128-bit Grain key.
2. Encryption:
For each row in the CSV data:
i.   Concatenate the row values into a single string.
ii.  Encode the plaintext using the encrypt_data() function:
1. AES-256 encrypts the plaintext using the AES key.
2. Grain encrypts the plaintext using the Grain key.
3. HMAC-SHA256 validate the concatenated AES ciphertext and Grain ciphertext using the HMAC key.
iii. Store the AES ciphertext, Grain ciphertext, and HMAC tag for the row.
3. Decryption:
For each encrypted row:
i.   Decode the data using the decrypt_data() function:
1. Confirm the HMAC-SHA256 tag using the HMAC key.
2. Decipher the AES ciphertext using the AES key.
3. Decode the Grain ciphertext using the Grain key.
ii. Store the decrypted row.
4. Output:
i. Print the original data (first 5 rows).
ii. Print the encrypted data (first 5 rows, including AES ciphertext, Grain ciphertext, and HMAC tag).
iii. Print the decrypted data (first 5 rows).
iv Print the encryption keys (AES, HMAC, Grain).
```

The dataset utilized for the system was obtained from (Goldberger et al., 2000; Vera Novak et al., 2010). Table 1 describes the dataset, which consists of blood pressure data from patients. Table 1 displays a patient's blood pressure together with the time and date the information was recorded.

Table 1: s0030-bp-verified-0

Date	Time	Sys.	Dia.	MAP	HR
1/11/2005	8:06	130	81	97	78
1/11/2005	8:07	133	76	97	78
1/11/2005	8:08	132	81	98	79
1/11/2005	8:29	126	76	95	80cr
1/11/2005	8:49	129	79	92	86
1/11/2005	9:09	130	73	92	81

RESULTS AND DISCUSSION

Table 2 illustrates five types of encrypted blood pressure data from patients. It also shows the file size, HMAC, encryption and GRAIN keys. Table 3 presents the complete encryption and decoding results. TECA stands for Time taken to encrypt CSV data with AES-256, TGGV for Time taken to produce GRAIN values, TGHV for Time taken to generate HMAC values and TDED for Time taken to decrypt the encrypted CSV data. The outcomes demonstrate that TECA, TGGV, TGHV and TDED scores for s0030-bp-verified-0 are 0.014912 seconds, 0.000000 seconds, 0.021086 seconds and 0.013965 seconds, respectively. Figure 2 depicts the entire encryption and decoding results in vivid detail. Figure 2 shows the implications of hashing, decrypting and coding data using the AES 256, Grain and HMAC methods.

Table 2: Encrypted Data

N/S	Encrypted file	File Size	Encryption Key	HMAC Key	GRAIN Key
1	s0030-bp-verified-0.csv	2.06 kB	5c243b09a0db411724edbd569bdd3e813278bec2c2f2296e9c2719a381526e71	d23dcd14be46e391fa8d1bf119456a7cfa07e09ac29c8339cfc7434b8519f28c	a899de093b521b0ed7140a9e71e5f7f6
2	s0033-bp-verified-0.csv	2.04 kB	5a9042387a5c01a250b554a0797ebf1b6b63112734202de6e57f340d1ba5d06b	3f016e50ac03f9d728d5cec4842bd9b99b69b7b8725af4e166c806e9fcf3f12e	403448e645f42ddb5a670719a1d36a01
3	s0044-bp-verified-0.csv	1.07 kB	59f9ef2f6d8d3b3142737952da19c60452fe8f04fbe3e31387d61dfd78a2e432	bb0eb306efccc985f0bb03be775771f6cc4192e0f8ea2fe01a06dacc93844881	0c89a6d4f5e8eab345111f20bb5e7f94
4	s0064-bp-verified-0.csv	1.92 kB	8276ec93b3875815f279e9f3588efbadb63f9926b2113795d06ed1434db20e1e	2dd164d93f21494d5449991739ec332c2ac5ff9e01dc431b11c3c98a035e940d	9c822031c276bdde0683614d59710d3f
5	s0068-bp-verified-0.csv	2.03 kB	e77376ce02824c3b46b3d246cb480166a4612247839d6e6cd9f7897f8d1fb0fc	df074fe96dd5e4c9d2e2570d348810661f6068c7cef8e28dbcfaddc66579f1dd	bebc78e66c49284043153cf296fc7c7d

Table 3: Complete Result of the proposed system

S/N	Data	TECA	TGGV	TGHV	TDED
1	s0030-bp-verified-0.csv	0.014912	0.000000	0.021086	0.013965
2	s0033-bp-verified-0.csv	0.012641	0.000000	0.011090	0.014280
3	s0044-bp-verified-0.csv	0.006761	0.000000	0.006112	0.008045
4	s0064-bp-verified-0.csv	0.010800	0.000000	0.011425	0.004986
5	s0068-bp-verified-0.csv	0.014943	0.000000	0.016808	0.012027

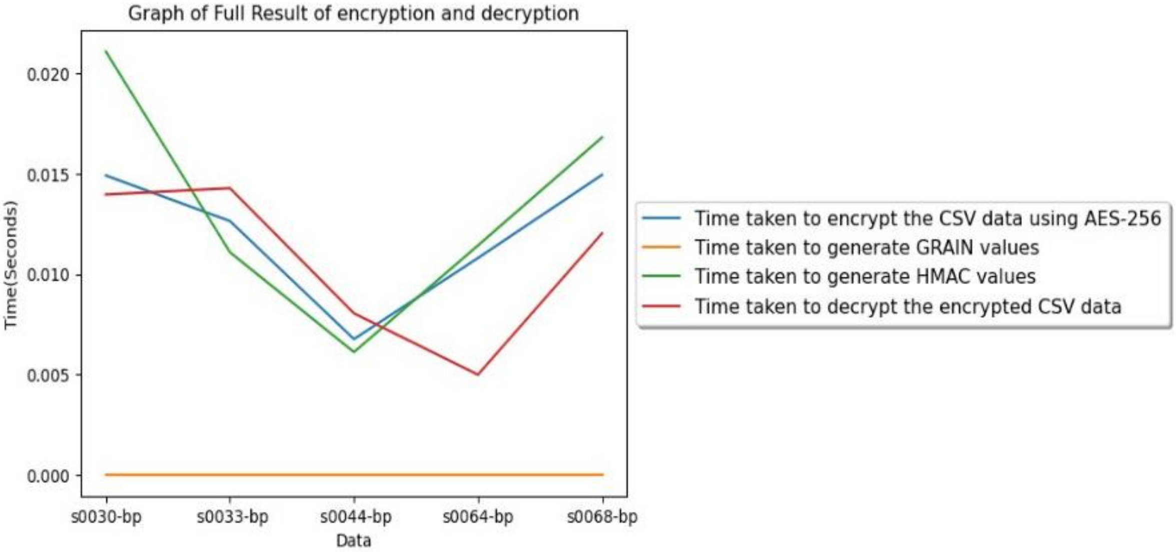

Figure 2: Complete Result of Encryption and Decryption using AES, GRAIN and HMAC

CONCLUSION

IoT Healthcare equipment is frequently battery-powered or have limited energy sources, making power consumption an important consideration. Cryptographic techniques that are computationally costly or need frequent memory access can deplete the device's battery. Lightweight, energy-efficient approaches, such as Grain may be more suited for IoT devices with stringent power limits. The validation for employing these methods is to provide a secure and reliable data encryption and decryption process. AES-256 is a widely used and highly secure symmetric-key method, Grain is a lightweight stream cipher designed for resource-constrained situations and HMAC-SHA256 provides data authentication and integrity. By combining these strategies, the code assures the security, integrity, low power consumption and validity of encrypted data in an IoT healthcare device. This powerful encryption technology is frequently used in applications requiring the protection of sensitive information, such as financial transactions, healthcare data or confidential conversations. The findings indicate s0068-bp-verified-0 The time taken to encode the CSV data using AES-256 is 0.014943 seconds, to generate GRAIN values is 0.000000 seconds, to generate HMAC values is 0.016808 seconds and to decode the encrypted CSV data is 0.012027 seconds. This demonstrates that the AES 256, Grain and HMAC Methods can provide data integrity, faster encoding and decoding and lower power usage.

REFERENCES

1. Benhani, E. M., Lopez, C. M., & Bossuet, L. (2019). Secure internal communication of a TrustZone-enabled heterogeneous SoC lightweight encryption, In 2019 International Conference on Field-Programmable Technology (ICFPT), 239-242, IEEE.

2. Bhaumik, R., Dutta, A., Inoue, A., Iwata, T., Jha, A., Minematsu, K., ... & Tessaro, S. (2025). Cryptographic Treatment of Key Control Security: In Light of NIST SP 800-108, In Annual International Cryptology Conference, 371-403, Cham: Springer Nature Switzerland.
3. Chitra, S. R. & Chelliah S. (2025). Enhanced Cloud Data Security by Employing HMAC for Advanced Cryptographic Protection, Innovations in Intelligent Systems and Advanced Engineering, Vol. 1(1), 1-10.
4. Du, L., & Xie, T. (2024). Towards Secure Internet of Things-Enabled Healthcare: Integrating Elliptic Curve Digital Signatures and Rivest Cipher Encryption, International Journal of Advanced Computer Science & Applications, 15(8), 581-589.
5. Faith, O. (2023). A Systematic Review of Attribute-Based Encryption for Secure Data Sharing in Iot Environment, Degree project at the master level to Computer and Systems Sciences Department, Stockholm University.
6. Goel, A., Baliyan, H., Tyagi, S., & Bansal, N. (2024). End to end encryption of chat using advanced encryption standard-256, International Journal of Science and Research Archive, 12(01), 2018–2025.
7. Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation [Online]. 101 (23), pp. e215–e220.
8. Guy-Cedric, T. B., & Suchithra, R. (2018). A Comparative Study on AES 128 BIT AND AES 256 BIT, International Journal of Scientific Research in Computer Science and Engineering, Vol. 6(4), 30-33, E-ISSN: 2320-7639.
9. Hashim, M. M., Salim, T. A., & Kalid, H. N. (2020). Secure Patient Data Transmission Using Information Hiding System and Medical IoT, Technology Reports of Kansai University, Vol. 62(8), 4572-4585, ISSN: 04532198.
10. Kak, A. (2015). Lecture notes on computer and network security, Purdue University, 1-94.
11. Luc, N. Q., Tran, T. N., Ngo, C. K., Tran, H, D., Nguyen, V. C., & Tran, T. A. (2022). Implementation Of Authenticated Encryption with Associated Data Grain-128aead Algorithm on Stm32f400 Processor Family, Transport and Communications Science Journal, Vol. 73(4), 427-438.
12. Madhumathi, C. S., & Vishnu K. K. (2025). Enhancing privacy in IoT-based healthcare using provable partitioned secure blockchain principle and encryption, Scientific Reports, 15(1), 29682.
13. Marcus, D. R., Georgios, M., & Jonathan, R. (2023). Grain-128PLE: Generic Physical-Layer Encryption for IoT Networks, arXiv:2309.15569v1 [cs.CR].
14. Muhamad, R. R., Agung, T., & Gatot, S. (2024). Combination of AES (Advanced Encryption Standard) and SHA256 Algorithms for Data Security in Bill Payment Applications, SAGA: Journal of Technology and Information Systems, Vol 2(1), 175-189, ISSN: 2985-8933.
15. Musthafa, M. M., Thangavel, P., & Anand P. (2025). Quantum Cryptography with Espresso Ciphers and Grain for Enhanced Security in Optical Communication Networks, ICTACT Journal on Communication Technology, Vol. 16(1), 3432-3436.
16. Naman, H., Hussien, N., Al-dabag, M., & Alrikabi, H. (2021). Encryption system for hiding information based on internet of things, 172-183.
17. Nasera, N. M., & Naifa, J. R. (2022). A systematic review of ultra-lightweight encryption Algorithms, Int. J. Nonlinear Anal. Appl., 13(1), 3825-3851, ISSN: 2008-6822 (electronic).
18. Nikhil, A. (2020). Using AES Algorithm Encryption and Decryption of Text File, Image and Audio in Openssl and Time Calculation for Execution, IOSR Journal of Computer Engineering (IOSR-JCE), Vol. 22(6), 39-44, e-ISSN: 2278-0661, p-ISSN: 2278-8727. DOI: 10.9790/0661-2206013944.
19. Nirwan, S., Hamidin, D., & Azzalea, S. E. (2024). Implementation of AES-256 Algorithm for Encryption on Chatting Platforms, Iota, ISSN 2774-4353, Vol. 4(4), 617-624. https://doi.org/10.31763/iota.v4i4.80.
20. Nizamuddin, A. K. & Dolly, V. S. Y. (2024). Implementasi Kriptografi Berbasis Web dengan Algoritma Advanced Encryption Standard (AES) 256 dan Kompresi Huffman untuk Pengamanan File di SMK Satria, Jurnal TICOM: Technology of Information and Communication, Vol. 12(2), 50-55.
21. Nureni, A. A., & Onyema, J. C. (2018). Achieving Data Authentication with Hmac-Sha256 Algorithm, GESJ: Computer Science and Telecommunications, 2(54), ISSN 1512-1232, 35-43.
22. Okpu, E. O., Taylor, O. E., Nwiabu, N. D., & Matthias, D. (2024). A hybrid machine learning approach for intrusion detection and mitigation on IoT smart healthcare, International Journal, 13(7), 82-90.

23. Okpu, E. O., Taylor, O. E., Nwiabu, N. D., & Matthias, D. (2024). Comparative Performance Analysis of Cryptographic Techniques for Securing the Physical Layer in Internet of Medical Things (IoMT) Systems, International Journal of Computer Science and Mathematical Theory (IJCSMT), 157-170.
24. Okpu, E., & Taylor, O. (2025). Analysing the Integration of AES-256 Encryption and HMAC Hashing in IoT Smart Healthcare Systems, Ci-STEM Journal of Digital Technologies and Expert Systems, 2(1), 18-24.
25. Pasaribu, H., Sitanggang, D., Damanik, R. R., & Sitompul, A. C. R. (2018). Combination of advanced encryption standard 256 bits with md5 to secure documents on android smartphone, In Journal of Physics: Conference Series, Vol. 1007(1), 1-9, IOP Publishing.
26. Sabri, O., Al-Shargabi, B., Abuarqoub, A., & Hakami, T. A. (2025). A Lightweight Encryption Method for IoT-Based Healthcare Applications: A Review and Future Prospects, IoT, 6(2), 23.
27. Shuwandy, M. L., Salih, A. K., Khaleel, F. L., & Habbal, A. M. M. (2010). Switching between the AES-128 and AES-256 Using Ks * & Two Keys, IJCSNS International Journal of Computer Science and Network Security, Vol.10(8), 136-139.
28. Syabdan, D., Emansa, H. P., & Muhammad, A. F. (2023). Restful Api Security Using Json Web Token (Jwt) With Hmac-Sha512 Algorithm in Session Management, IT Journal Research and Development (ITJRD), Vol. 8(1), E-ISSN: 2528-4053, P-ISSN: 2528-4061.
29. Taofik, I., Hura, I. A., Aziz, M. F. A., Pardamean, J., & Napitupulu, I. A. (2023). Implementasi JSON Web Token (JWT) untuk Authentication Data pada Aplikasi Bayeue Dengan Algoritma HMAC SHA-256, 1-8.
30. Vera Novak, Kun Hu, Laura Desrochers, Peter Novak, Louis Caplan, Lewis Lipsitz, and Magdy Selim (2010). Cerebral flow velocities during daily activities depend on blood pressure in patients with chronic ischemic infarctions. Stroke; a Journal of Cerebral Circulation, 41(1), 61–66. http://doi.org/10.1161/STROKEAHA.109.565556.
31. Vishwasrao, S., Abhishek, T., Chandrasekhara, M., Punit, Goel., & Anshika, A. (2024). Advanced Encryption Techniques in Healthcare IoT: Securing Patient Data in Connected Medical Devices, Modern Dynamics: Mathematical Progressions, Vol. 1(2), 224-247, ISSN: 3048-6661.