Available online at https://journal.rescollacomm.com/index.php/ijqrm/index

International Journal of Quantitative Research and Modeling

e-ISSN 2721-477X
p-ISSN 2722-5046

Vol. 5, No. 3, pp. 311-317, 2024

Enhancing Email Client Security with HMAC and PGP Integration to Mitigate Cyberattack Risks

Ayu Nur Oktaviani[1*], Asep Id Hadiana[2], Melina[3]

[1,2,3]*Department of Informatics, Faculty of Science and Informatics, Universitas Jenderal Achmad Yani, Jl. Terusan Jend. Sudirman, Cimahi, West Java, 40525, Indonesia*

**Corresponding author email: ayunuro20@if.unjani.ac.id*

Abstract

The rapid advancement of technology in the modern era has significantly increased the risk of data breaches and misuse, particularly in email communications. Ensuring data privacy and security is crucial to preventing information theft and mitigating cyberattack risks. This research focuses on enhancing email client security through the integration of Hash-Based Message Authentication Code (HMAC) and Pretty Good Privacy (PGP). HMAC is employed as a message authentication mechanism to ensure the integrity and authenticity of email messages, while PGP is utilized to generate public and private key pairs, enabling secure encryption and decryption processes. By integrating these two security methods into the email client system, we aim to enhance its resilience against cyber threats. The system's effectiveness was evaluated through black-box testing, demonstrating its capability to secure the email delivery process. Additionally, an analysis of key randomness using the entropy method revealed a maximum value of 6 bits, indicating a relatively high level of randomness and further strengthening the encryption process. The results of this study indicate that the combined use of HMAC and PGP provides a robust security solution for enhancing email client security and mitigating potential cyberattack risks.

Keywords: Cyberattack mitigation, encryption, hash, HMAC, Email Message, PGP.

1. Introduction

In the rapid development of information technology, the security of digital communications is becoming increasingly important. Email, as one of the main means of exchanging information in both business and personal contexts, is often the target of various types of cyber threats. These threats include illegal attempts to access, manipulate, damage or steal information sent over email networks. In 2020, incidents of information theft via email occurred at several major institutions. For example, around 25,000 email credentials from the World Health Organization (WHO), National Institutes of Health (NIH), and the Gates Foundation were stolen by the Neonazi group. The stolen credentials were then used to spread information related to the campaign and various conspiracy theories about COVID-19 (Wijaya 2020). There are many email management applications such as Ms. Office Outlook that allow users to send, receive, and manage emails from one or more email accounts, including personal, business, and academic emails. However, if they are not used carefully, messages in the inbox can be easily accessed by other parties, and there are no adequate means to protect emails from unauthorized individuals (Ridwan and Susano 2023). One way to secure information sent via email is to encrypt it. The application of a single cryptographic technique has a greater risk of data leakage compared to the level of security that can be achieved through the combined use of multiple cryptographic techniques. Therefore, to optimize the security of email messages before they are sent, it is essential to implement two cryptographic techniques (Zulfikar, Abdillah, and Komarudin 2019). A combination of several techniques shows better performance than a single technique (Melina et al. 2023). This research deepens the study of the use of two important security technologies, namely Pretty Good Privacy (PGP) and Hash-based Message Authentication Code (HMAC), with the aim of understanding, implementing, and testing their effectiveness. Through this exploration, the research is expected to significantly contribute to the development of better email security practices and enhance the protection of communications in a digital age that is increasingly vulnerable to cyber-attacks.

2. Literature Review

HMAC (Hash-based Message Authentication Codes) is a type of Message Authentication Code (MAC) that utilizes hash algorithms, such as MD5 and SHA, along with a secret key (Zulfikar, Abdillah, and Komarudin 2019). Research on the use of HMAC (Ichwan, Gustian, and Nurjaman 2018) combines a hash function (SHA256) with a secret key, effective in ensuring the integrity and authenticity of messages in home security systems. Avalanche effect test results obtained from the first 16 rounds in the round function contained in the SHA256 algorithm can produce an average Avalanche effect value of 62%, and after 64 rounds produce an Avalanche effect value of 85.9%. This shows that the output of the SHA256 algorithm has a good level of randomization so that someone cannot predict the input message only from the output of the SHA256 algorithm. The hash function used in this research is SHA-256. This hash function acts as a building block in Message Authentication Code (MAC). The hash function processes a variable-size string as input and produces a fixed-size Hex code as output. This output is called the hash code or message digest, which plays a crucial role in many modern applications. Hash functions are non-reversible and support a one-way property, which means it is impossible to retrieve the input data from the output after hashing (Ananda, Fauziah, and Hayati 2020).

Pretty Good Privacy (PGP) is a computer program that makes it possible to send messages, emails, or files by adding confidentiality features and additional user authentication, namely with digital signatures to ensure the contents of the message sent match what was received. PGP has a session key that is used to encrypt data and a key pair belonging to the sender and receiver of a file. (Rafael 2020). A number of previous studies that discuss email security include research (Ananda, Fauziah, and Hayati 2020) which examines the use of the Pretty Good Privacy (PGP) method and the Rivest Shamir Adleman Algorithm (RSA). The results of this study show that the Pretty Good Privacy method is quite effective in securing email, because the data information changes significantly based on tests conducted using 8 data sent randomly to several users.

There are previous studies that reveal that the combination of Pretty Good Privacy (PGP) to generate encryption keys and Hash-based Message Authentication Code (HMAC) for key authentication is a suitable combination for email security. The protocol proposed in this study uses PGP to generate the public key and private key used in message encryption and decryption, while HMAC is used to ensure the authenticity of the user-submitted key. This process enables automatic key exchange between users with the help of the mail server and the designed protocol (Maier n.d.).

3. Methods

The following illustrates the development of a method that combines HMAC and PGP to improve email client security is shown in Figure 1.

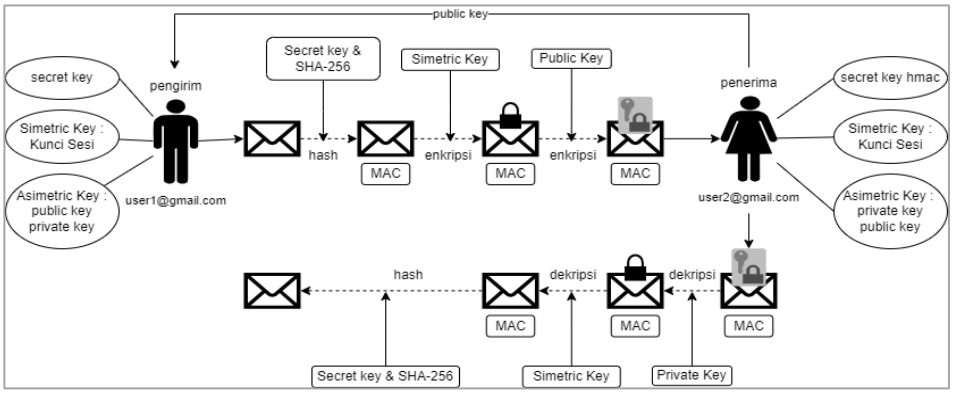

Figure 1: Methods

3.1. Hash-Based Message

3.2. Authentication Code (HMAC)

HMAC is used for authentication and data integrity through the use of secret keys, which is different from digital signature methods that utilize public key cryptography (Wattimena, Pakereng, and Wowor 2016). The HMAC algorithm is described in the following equation (1).

$$HMAC_K(m) = h\left((K \oplus opad)||h((K \oplus ipad)||m\right) \quad (1)$$

Where K is the secret key with a maximum length of 64 characters. This key has 0 bits added to make it 512 bits, m is the message used and h is the hash function, while ipad (inner padding) and opad (outer padding) are additional bits of size 512 bits (Sukarno 2022).

3.3. Pretty Good Privacy (PGP)

PGP is based on the concept of private key cryptography as the basis of authorization. This key is used to encrypt communication between two machines. To keep data confidential, cryptography converts plaintext messages into unrecognizable ciphertext. This ciphertext is then sent by the sender to the receiver. PGP generates two keys during key generation: a private key and a public key, where the public key is widely announced. A person who wants to send a message to another person must look up that person's public key on a website. The public key is used to encrypt the message, only the recipient has the private key to decrypt the message, so the ciphertext message passing through the network remains secure from intruders. One method used to create public and private key pairs is the Rivest, Shamir, Adleman (RSA) method (Kasau et al. 2021) (Melina et al. 2022).

RSA has a high level of security because it involves two keys, namely a public key that can be accessed by anyone and used to perform the encryption process, as well as a secret key that is only known by authorized parties and used to perform the decryption process. The RSA cryptographic algorithm consists of three stages, namely the key establishment stage, the encryption stage, and the decryption stage (Rifai, Christyono, and Santoso 2016). This algorithm is based on the exponential process and factoring a number into two prime numbers, which until now took a very long time to complete the factoring (Berliano Novanka Putra et al. 2022).

The following are the stages of using the RSA algorithm to generate public and private key pairs :

1. Choose two large primes, e.g. determine p (61) and q (53)
2. Calculated $n = p \times q = 61 \times 53 = 3233$ as the modulus for the public and private keys.
3. Calculated $\phi(n) = (p-1) \times (q-1) = (p-1) = 60 \times 52 = 3120$
4. Choose an integer e as the public exponent that satisfies $1 < e < \phi(n)$ and $gcd(e, \phi(n)) = 1$. e.g. *e=17*
5. Calculate d as the private exponent that satisfies $d \equiv e^{-1} \, mod \, \phi(n) = 17^{-1} \, mod \, 3120 = 2753$.
6. Public key: $(e, n) = (17, 3233)$ and private key: $(d, n) = (2753, 3233)$

3.4. Entropy

Entropy is used to test the randomness of the generated keys. The higher the entropy, the greater the uncertainty or surprise associated with the results generated from the distribution (Saraiva 2023). Experiments with entropy showed results close to the highest value. This method has a key space size of K with the best entropy value being 8. The larger the entropy value, the more difficult it is to crack the ciphertext. The entropy calculation is done using the equation (2) (Ravida and Santoso 2020).

$$H(X) = -\sum_{i=0}^{n} a_i \, 2\log(p(S_i)) \tag{2}$$

Where X is the message, S_i is the message symbol, $p(S_i)$ is the probability of occurrence of S_i, and a_i is the number of occurrences of S_i.

3.5. Black Box Testing

Black Box testing is a method for testing the functionality of an application system. Testing is done using random data as input, with the aim of obtaining clear results. Results are said to be clear if, in the event of an error, the information system rejects the input or the data is not stored in the database. Conversely, if the input data is correct, the data will be accepted and stored in the information system database. In this research, the testing technique applied to the Email Client application is the Equivalence Partitioning Technique. This technique divides the input data of the software unit into several data partitions, from which test cases can be derived. Basically, test cases are designed to cover each partition at least once. This method aims to define test cases that can reveal groups of errors, so as to reduce the number of test cases that need to be created (Uminingsih et al. 2022).

4. Results and Discussion

4.1. Aplication Design

The application design is made using activity diagrams for the process of creating public and private key pairs and the process of sending and receiving email messages. The design is shown in Figure 3 - Figure 5.

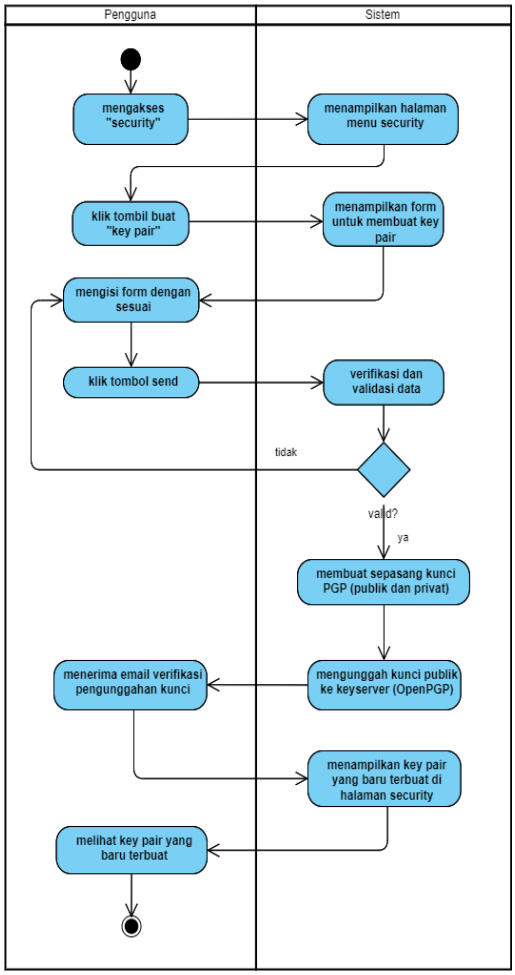

Figure 2: Activity Diagram of Creating a New Key Pair

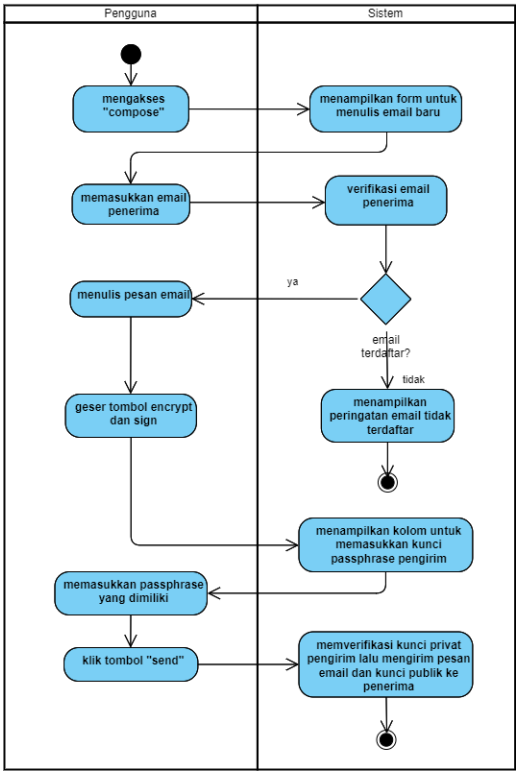

Figure 4: Activity Diagram of Sending Email

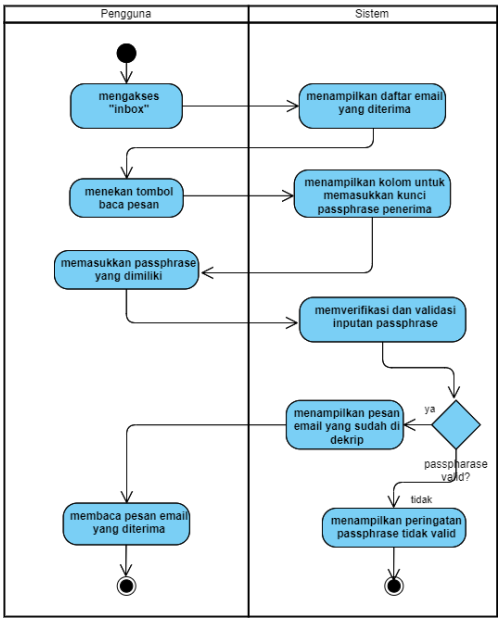

Figure 3: Activity Diagram of Receiving Email

4.2. Result

Based on the design made, this application was developed using the Django framework, resulting in an email client equipped with public and private key generation features, as well as the ability to send and receive emails protected through encryption and decryption processes using HMAC and PGP.

Figures 5 and 6 show the implementation of the RSA Algorithm to create a key pair, namely a public key and a private key, where both keys are stored and can be used using the generated passphrase key. Figures 7 and 8 show the process of sending an email starting with activating the sign and encrypt features on the compose form. After pressing the Send button, the system performs the encryption process on the message and then sends it. The process in it is that the system runs the HMAC function to authenticate the message, then runs the PGP encryption function, the public and private keys are used in the encryption process. Figures 9 and 10 are the results of the implementation of the process of receiving email done by opening one of the incoming emails on the inbox menu, then opening it using the passphrase that has been made before. The decryption process is to take the recipient's private key to decrypt the message by running the PGP decrypt function, then the decrypted message is authenticated using HMAC.

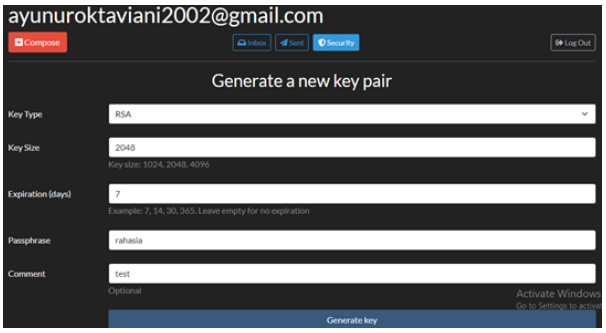

Figure 5: Create a New Key

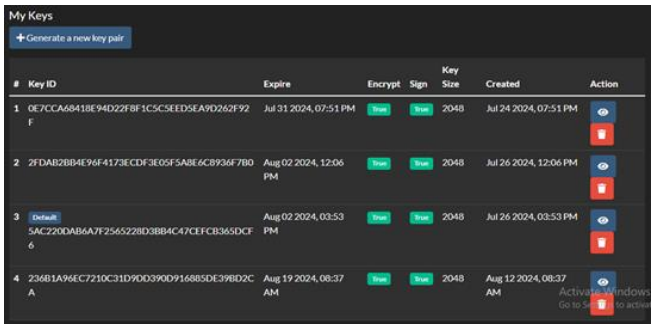

Figure 6: List of Owned Keys

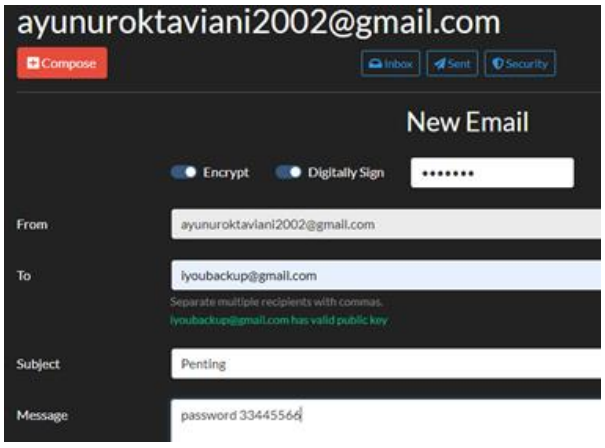

Figure 7: Composing Email Messages

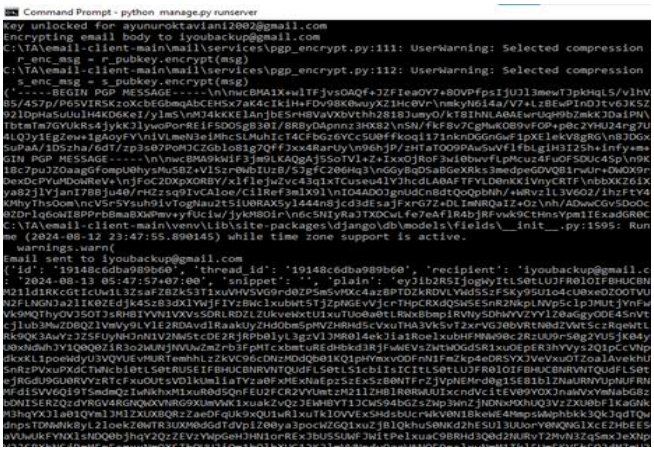

Figure 8: Process of Encrypting and Sending Messages

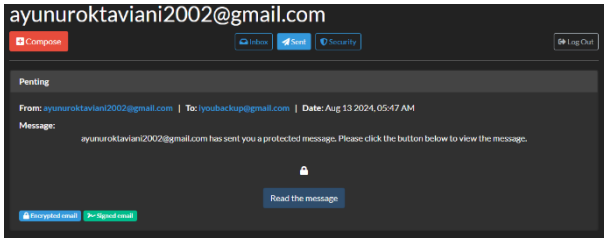

Figure 5: Opening One of the Emails

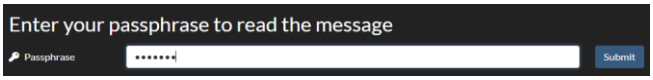

Figure 6: Entering a Passphrase Key

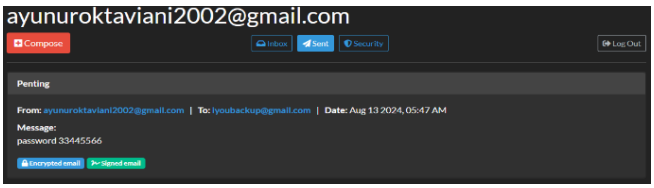

Figure 7: Display of Email Messages

4.3. Aplication Testing

Black box testing was conducted to test the process of creating keys, sending emails, and receiving emails. The results of black box testing are shown in Table 1.

Table 1: Black Box Testing

Scenario	Expected Results	Results
Sender and receiver fill out the form to create a key pair in the security menu.	Generate public and private keys and passphrase keys	Accepted
The sender activates the encrypt and sign features then composes the email message and presses the send button on the compose menu.	Generates an encrypted email message then sends it to the recipient	Accepted
The recipient receives the incoming email in the inbox menu and opens it using the passphrase key that has been created previously.	Decrypted secure message	Accepted

4.4. Security Testing

Testing is done by utilizing a tool in the form of an Online Calculator from planetcalc.com. The testing process focused on the private and public keys generated by PGP. This tool is used to calculate the key entropy, with the results expressed in units of bits. The entropy calculation was performed on five pairs of keys generated by PGP on the email client system as follows.

Table 2: Entropy Testing

Key Size	Key ID	Entropy (*bits*)	
		Private	Public
1024	B5C30D91D4C2A71A65F710F7117E720C7974C7A2	5.977	5.917
2048	F1B32837F56BD565BC5EAA0E4C44E55C45FB1559	6.007	5.990
4096	7FBAF5322822DF86CB2E05C334E0D613CBC13654	6.016	5.998
1024	ECF96C774DF1515A5D1AB1A208BEB7C9E3C90F79	5.981	5.968
2048	263F53AA8410F787F04E09BB2036C230C252D7B3	5.993	5.980

Based on the entropy calculation results above, it can be concluded that the highest value is 6,016 with the longest key size of 4096. This indicates that the longer the key size, the higher the entropy value.

5. Conclusion

The integration of HMAC and PGP has been demonstrated to significantly enhance email client security, effectively mitigating the risks associated with cyberattacks. HMAC provides a robust mechanism for ensuring the integrity and authenticity of email messages through unique hash generation, while PGP fortifies security by encrypting messages and enabling digital signatures via a public and private key pair. This dual-layered security strategy not only defends against unauthorized access and tampering but also maintains the confidentiality of email communications. The system's resilience was validated through comprehensive security testing, achieving a notable entropy value of 6.016 bits with a 4096-bit key length, underscoring its capability to withstand a variety of cyber threats. Future research should aim to expand this secure framework, enabling the transmission of images, videos, and other file types while maintaining the same high level of security, thereby further enhancing its effectiveness in combating evolving cyberattack techniques.

References

Ananda, Ridwan Ighfirlana, Fauziah, and Nur Hayati. 2020. “Email Security Using Pretty Good Privacy Method With Rsa Algorithm.” *Jurnal Ilmiah Informatika Komputer* 25(3): 213–24.

Berliano Novanka Putra, Nathanael et al. 2022. “Analysis of Asymmetric Cryptography Encryption of RSA Algorithm Based on Batch Programming on Flashdisk Media.” *Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)* 8(2527–5771): 51–61. https://tunasbangsa.ac.id/ejurnal/index.php/jurasik.

Ichwan, M, Milda Gustian, and Novan Rizky Nurjaman. 2018. “Implementation of Keyed-Hash Message Authentication Code in Home Security System.” *MIND Journal* 1(1): 9.

Kasau, Matius Irsan et al. 2021. "Security System ' Message ' Based Pretty Good Privacy." *SISITI: Seminar Ilmiah Sistem ...* X(1): 108–16. http://ejurnal.dipanegara.ac.id/index.php/sisiti/article/view/793.

Maier, Thomas. "Automated Key Management for End-To-End Encrypted Email Communication." (section III).

Melina, Melina, Firman Sukono, Herlina Napitupulu, and Valentina Adimurti Kusumaningtyas. 2022. "Electronic Signature Verification with Authentication Technique Based on Public Key Cryptography System Using Rivest-Shamir-Adleman Cryptographic Algorithm." *Jurnal Matematika Integratif* 18(1): 27.

Melina, Sukono, Herlina Napitupulu, and Norizan Mohamed. 2023. "A Conceptual Model of Investment-Risk Prediction in the Stock Market Using Extreme Value Theory with Machine Learning: A Semisystematic Literature Review." *Risks* 11(3).

Rafael, Riandy. 2020. "Penerapan Algoritma Pgp Untuk Enkripsi Csv File Di Pt. X." : 6.

Ravida, Roiya, and Heru Agus Santoso. 2020. "Advanced Encryption Standard (AES) 128 Bit for Internet of Things (IoT) Data Security of Hydroponic Plants." *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)* 4(6): 1157–64.

Ridwan, and Adhi Susano. 2023. "Multi-Platform Email Client Application with Java-based Data Encryption." *Jurnal Indonesia : Manajemen Informatika dan Komunikasi* 4(1): 279–90.

Rifai, Rizal Yunan, Yuli Christyono, and Imam Santoso. 2016. "Implementation of Cryptographic Algorithms Rivest Code 4, Rivest Shamir Adleman, and Steganography Methods for Securing Secret Messages in Digital Text Files." *TRANSIENT Jurnal IlmiahTeknik Elektro* 5(1): 86–91.

Saraiva, Paulo. 2023. "On Shannon Entropy and Its Applications." *Kuwait Journal of Science* 50(3): 194–99. https://doi.org/10.1016/j.kjs.2023.05.004.

Sukarno, B B. 2022. "Implementation of HMAC and SHA3 Algorithms on JWT for Web Authentication." *Informatika.Stei.Itb.Ac.Id* (18219017): 1–6.

Uminingsih, Muhamad Nur Ichsanudin, Muhammad Yusuf, and Suraya Suraya. 2022. "Functional Testing of Library Information System Software with Black Box Testing Method for Beginners." *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer* 1(2): 1–8.

Wattimena, Aldrien, Magdalena A. Ineke Pakereng, and Alz Danny Wowor. 2016. "Design of Message Authentication Code (MAC) Algorithm with 256 Bit Based Block Cipher Cryptography Approach on Dart Board Pattern Design of Message Authentication Code (MAC) Algorithm with 256 Bit Based Block Cipher Cryptography Approach.." : 1–25.

Wijaya, Robby. 2020. "Implementation of Aes and Rc4 Algorithms for Email Message Security." 11(2): 64–71.

Zulfikar, Muhammad Iqbal, Gunawan Abdillah, and Agus Komarudin. 2019. "Cryptography for Secure Email Delivery Using Blowfish and Rivest Shamir Adleman (RSA)." *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*: 19–26.