Journal of Positive School Psychology
2022, Vol. 6, No. 3, 8415–8422

Analysis on Computational Time of Hybrid Cryptography in Email System

Zolidah Kasiran[1] , Azrina Dalil [2], Mohd Zaki Ghazali[3]

[1]Senior Lecturer, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia
[2]Graduate Student, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia
[3]Associate Professor, Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia
Email: [1]zolidah@tmsk.uitm.edu.my, [2]azrinadalil23@gmail.com, [3]mhzaki@tmsk.uitm.edu.my

Abstract

E-mail is one of the most prominent, fastest and low-cost means of communication. Millions of people have made E-mail as a part of everyday life where it changes the way of work and collaboration as the E-mail messages can be sent to an individual or groups. However, the inherent vulnerability of the E-mail system can cause immense risks of information disclosure and misuse. The common threats to the E-mail system are malicious threats whether it is a malware attachment or a URL leading to malware, phishing or exploitation, spam, spoofing and unauthorized access. The aim of this project is to enhance the E-mail security system from being attacked by threats. This paper had proposed to use modified AES and RSA hybrid algorithms to increase the E-mail security system. Time execution of the algorithm is taken as the result of this research paper. It can be concluded that by modifying AES bit key to 320-bit, it can increase the E-mail security system.

Keywords: AES, cyber threats, e-mail, performance analysis, RSA.

I. INTRODUCTION

Email was once used for internal communication in the organization and has become widely used with the advancement of internet technology. Giant internet companies such as google offer an email account for free for everyone. E-mail is one of the most prominent, fastest and low-cost means of communication. Millions of people have made email as a part of everyday life where it changes the way of work and collaboration (Gavankar & Vidhani, 2017). The ability of multiple recipients and forwarding in email features has made the email become very convenient methods of disseminating information. A single email can be spread among millions of email users within a few moments and for those reasons, email has become a widely used medium for communication of terrorists as well (Rani, 2015).

For example, in 2015, a mining consultant named Mr Fouché, gave a written mandate to Global and Local Investments Advisors (PTY) LTD to act as his agent and invest money with a bank on his behalf. The mandate had stipulated that all instructions addressed to Global are either by fax or by E-mail with Mr Fouché's signature. However, in August 2016, Mr Fouché's Gmail account had been hacked by fraudsters where three E-mails were sent to the Global using his E-mail credentials, with instructions saying that funds to be transferred to a third party. The E-mails ended with the word "Nick". The Global paid out R804 000,00 from Mr Fouché's account to the unknown third party on three different days in August 2016. As soon as Mr Fouché became aware of this, he

notified Global that the E-mail was not sent by him and hence, he claimed for a refund. This case shows that online commercial transactions are vulnerable to hacking and clients should always be extra careful when it comes to online transactions.

As the world become digitized more and information technology becomes universal, leading to electronic and digital phenomena such as digital economy, e-government, e-business, e-bank, e-health system, e-education. These digitization had causing the traditional illegal activities are piercing into the virtual world as well, resulting an entirely new type of criminal activities has arrived threatening the security and integrity of the system, network, and information(Purevjav et al., 2016).

Email Security and threats

Nowadays, security is a key aspect in the field of information and communication technology. The inherent vulnerability of the E-mail system causes immense risks of information disclosure and misuse. Since E-mail is widely deployed, well understood and usually communicates with untrusted external organizations, it is often the target of attacks. Common threats to the E-mail systems are malicious threats to businesses whether it is a malware attachment or a URL leading to malware, phishing or exploitation, spam, spoofing and unauthorized access. Not only businesses people who will be infected by this problem but also health organizations, government and other E-mail users, as E-mail is their common communications tool to exchange information(Rawdhan & Ibrahim, 2017).

As email travel across the network from sender to the receiver, three basic security triad of confidentiality, integrity and availability are essential to be observed. Many attempts have been made by other researchers such as using authentication, compression and cryptography, E-mail encryption using hybrid cryptosystem, E-mail spam detection using genetic programming with SMOTE and by using machine learning to secure the E-mail system from all those threats. However, these methods still could not solve the problem completely(Kumar & Rana, 2016).

If the E-mail system is not secured well, attackers will exploit E-mail to conquer management over a corporation, access confidential information or disrupt IT access to resources. Hence, the confidentiality of information that is sent by the end users is not guaranteed anymore. This may result in the rights of the end users of E-mail service being neglected.

The aim of this project is to enhance the security of E-mail in order to acquire a secure communication and good system performance. The usage of hybrid cryptography in E-mail is an active research area as the technology evolved and criminals threaten its security.

II. RELATED WORKS

E-mail is one of the oldest applications on the Internet that is function to store and route E-mail messages between senders and recipients (Sabir & Yousaf, 2018) and it is regularly used as a primary form of communication and most widely used in daily life (Joseph Amalraj & John Raybin Jose, 2016), (Singh et al., 2017). It was first introduced in the 1960s, however it became available in the current structure in the 1970s. No meaningful attention was given in securing the application when email was first designed in 1971(Ruoti et al., 2019).

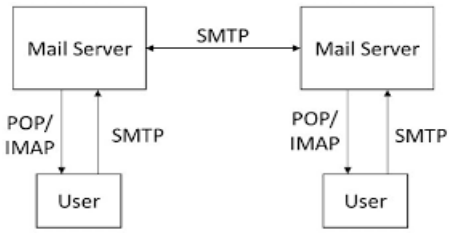

Figure 1: E-mail Architecture

Based on Figure 1, the E-mail communication is using three protocols generally, that are IMAP, POP and SMTP. IMAP, stands for Internet Mail Access Protocol, it is used while receiving an E-mail. If one uses IMAP, the E-mail will be present in the server and not downloaded to the user's mailbox. Then, the E-mail will be deleted from the server. This helps in decreasing the memory used in the local computer and server memory is increased.

While POP, stand for Post Office Protocol, is used for incoming emails too. The main difference between POP and IMAP is that POP downloads entire E-mail into the local computer and the data on the server will be deleted once it is downloaded. The current version of POP is POP3.

First the sender needs to enter the E-mail address of the recipient along with the message using an E-mail application such as Gmail, Hotmail, Yahoo! Mail and many more. This should be done at the local computers. Once it is finished, the "Send" button is clicked and the E-mail will be going to the MTA (Mail Transfer Agent). This communication is done via the SMTP protocol.

The next step is DNS lookup. The system sends a request to find out the corresponding MTA of the recipient. This will be done with the help of the MX record (Mail Exchanger record). In the DNS zone, for the receiver address' domain, there will be an MX record. It is a DNS resource record, which specifies the mail server of a domain. So, after the DNS lookup, a response is given to the requested mail server with the IP address of the recipient's mail server. This way the 'to' mail server is identified.

The next step is transferring the message between the mail servers. The SMTP protocol is used for this communication. Now our message is with the recipient mail server (MTA).

The message is transferred to the Mail Delivery Agent and then it is transferred to the recipient's local computer. Two type of protocols could be used either POP3 or IMAP. In the event of POP3 protocol is used, then the whole email is downloaded to the local computer and the copy at the server gets deleted. However, if the protocol used is IMAP, then the email message is stored in the mail server itself, but the user can easily manipulate the E-mails on the mail server as in the local computer. This is the difference when using both protocols and this is how the E-mail gets delivered. If some error occurred to send the E-mail, the E-mails will be delayed. There is a mail queue in every mail server and these mails will be pending in the mail queue. The mail server will keep trying to resend the email. Once the email sending failed permanently, the mail server may send a bounce back email message to the sender's email address.

Cryptography

Cryptography is a process by which it provides several security services to the messages or information such as confidentiality, data integrity or authentication to information communication systems that is sent from one user to another user (Kumar & Rana, 2016). Cryptography refers to secure information and communication techniques derived from mathematical concepts and a set of rule-based calculations called algorithms to transform messages in ways that are hard to decipher. These deterministic algorithms are used for cryptographic key generation and digital signing and verification to protect data privacy, web browsing on the internet and confidential communications such as credit card transactions and email. It includes techniques such as microdots, merging words with images, and other ways to hide information in storage or transit.

However, in today's computer-centric world, cryptography is most often associated with scrambling plaintext into cipher text, then vice versa, known as decryption. Three types of cryptography Asymmetric, Symmetric and hash function that are widely researched and applied.

Advanced Encryption Standard (AES) Algorithm and RSA Algorithm

Even though E-mail encryption tools remain underused by people who usually conduct sensitive business over E-mail (Lerner et al., 2017) but there are many researchers that are using encryption to secure E-mail security. AES is one of the encryption techniques that are used regularly because of its high efficiency and simplicity (Kumar & Rana, 2016). It is a symmetric block cipher that uses the same key for encryption and decryption process. In AES, the block and key size can be chosen independently from 128,160,192,224 and 256 bits. Besides, in the case of AES, the entire data

block is processed in parallel during each round using the substitutions and permutations.

Advanced Encryption Standard (AES) is a popular asymmetric cryptosystems. Researchers of this area keep enhancing the algorithm by combining AES with other algorithm such as Rivest-Shamir-Adleman algorithm (RSA) (Liu et al., 2018). The researchers used hybrid encryption algorithm that combines the advantages of fast encryption of AES algorithm, easy management of RSA algorithm key and digital signature. This is to ensure the secure transmission of confidential documents. Another researcher modified AES algorithm by changing the key size to 320 bits instead of the usual three different key sizes such as 128, 192 and 256 bits(Kumar & Rana, 2016), (Wu et al., 2017).

There are other ways that researchers did to modify AES algorithm, such as by modify AES SubBytes transformation to make it a round key dependent, to ensure that a change in the key is easily discovered in the cipher text and by modify the Shift Rows operation by randomizing the entire operation (Abikoye et al., 2019). Next, by modified the Mix Columns Transformation to bit permutation technique, where this technique alters the value of the state and shuffles the bits of the states or rearrange the bits across the state (Gamido et al., 2018).

Other than combining AES with RSA, method of using symmetric key, public key encryption system together with hash function is also improves the security performance (Purevjav et al., 2016). For the symmetric key, the author uses Ping Pong-128, public key cryptography uses RSA and the hush function uses MD5. Meanwhile (Plata et al., 2019) used Triple Data Encryption Standard (DES) to detect suspicious keyword in the email.

III. RESEARCH METHODS

This research focused on modifying the parameters of the AES algorithm. The key size was changed to 320 bits instead of 128,192 and 256 bits. As the AES parameters are depending on its key size, hence, the number of rounds has been increased to 16 rounds. The security of the system is increased by increasing the number of rounds and results in providing privacy to the unauthorized users.

Based on the Figure 2 , it shows the process of AES encryption proposed technique. AES parameters are depending on its key size. In the proposed algorithm, the number of rounds has been increased to 16 rounds, where for the first round it took 15 rounds for Sub Bytes, Shift Row and Mix column operations. While the last round, which is the 16th round, took place at the last process which is the Sub Bytes and Shift Row operations. Increase the number of rounds helps in providing more security to the system as it would take too much time for the attackers to hack the system and have a better performance. Therefore, key size of 320 bits has been chosen in order to provide a better result.

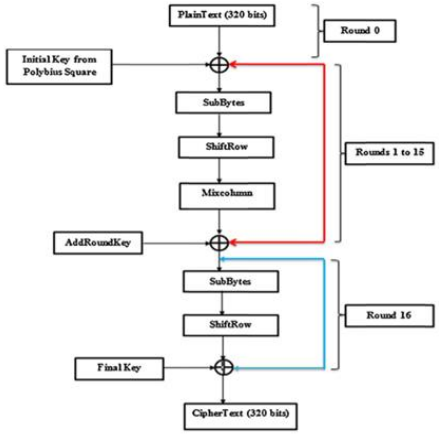

Figure 2: Process of AES Proposed Technique

Implementation of the Proposed Modified of AES Algorithm

The AES algorithm, its bits have been changed to 320-bits from its original bits that is 256-bits. The modification occurs at the line 20, the RoundKey, where the key is expanded into 272 rounds. RoundKey is the key expansion that derived from the cipher key using AES key schedule. AES uses key schedule to expand a short key into a number of separate round keys. The key is modified to 40 where it is the number of key used by a cryptographic algorithm. Upon the completion of the process the substitution box (sbox) is modified into 320 key size, where cryptographic algorithm based

on. The input and output of the process had been interpreted as polynomials over Galois Field (GF).

RSA Algorithm

The RSA algorithm also has been modified from the user who will decide the prime numbers to the algorithm that will generate its own prime numbers. This is because, for this algorithm, there will be no user input, hence it has been modified to generate its own prime numbers. The purpose of these prime numbers is to create the public and private keys. Thus, when it is being implemented into E-mail system, the algorithm can automatically generate its prime numbers so that it can create its own public and private keys.

```
377
378     cipher = powMod(i, e, n);
379     printf("The cipher key is: %d\n", cipher);
380
```

Figure 3: Main Function of RSA Algorithm

The Figure 3 shows the main function of RSA algorithm. At line 378 is where the encryption for AES key occurs. The mathematical function uses power mode where i is the variable of the AES key, e is the variable for power and n is the variable for modular. Script in line 379 shows the cipher function is called and shows the cipher value of the AES key. The RSA algorithm is combined with the AES algorithm to encrypt the AES key so secure the data been transmitted along the server.

Computational Effort

The time taken of the RSA algorithm is being calculated to ensure its efficiency in encrypting the data. The coding to calculate the time taken is located in the main function. The purpose to calculate the time taken is to compare the encryption process of original algorithm and the modified hybrid algorithm.

```
479     clock_t start, end;
480     double cpu_time_used;
481
482     start = clock();
483
```

Figure 4: Start Time to Calculate the Time Taken

The Figures 4 shows the 'start time' coding for execution time started to calculate the time taken for the algorithm to execute. This coding is located at the top of the main function.

```
528 end = clock();
529     cpu_time_used = ((double) (end - start)) / CLOCKS_PER_SEC;
530
531     printf("The time elapsed is %f seconds\n", cpu_time_used);
532
```

Figure 5: End Time to Calculate the Time Taken

The Figure 5 shows the 'end time' coding for execution time to end the calculation of the time taken for the algorithm to execute. The result will show in seconds. This coding is located at the bottom of the main function. Hence, the coding is actually calculating the time taken of the main process that is located between the start time and the end time.

IV. RESULTS AND FINDINGS

In this part, findings of the result will be the main of objective. The findings is based on the results shows.

Execution Time Taken Between AES 256-Bit and Modified AES 320-Bit Key.

For the first result, this experiment is based on the AES algorithm key parameter. The time execution is compared between the usage of AES algorithm that use 256 bit-key length and AES algorithm that use 320 bit-key length.

The data in Table 1 shows that the time taken for AES 256-bit to execute is 0.005 seconds, while the time taken for AES 320-bit to execute is 0.008 seconds. There is a 0.003 seconds difference in both results. Thus, this indicates that the modification of AES algorithm to 320-bit key has a slower execution time than AES algorithm 256-bit key. This is because of the key length used for AES 256-bit key is shorter than the key length used for AES 320-bit key. However, in terms of security, the longer the

key length used, the stronger its security. It is because it would take the attacker too much time to attack the algorithm via brute-force attack.

Table 1: Execution Time According to Algorithm

Algorithm Type	Execution Time
AES 256-Bit	0.005
AES 320-Bit	0.008
Hybrid AES 256-Bit	0.006
Hybrid AES 320-Bit	0.048

Execution Time Taken of Hybrid Cryptography

For the second result, this experiment is done on the hybrid algorithm, which is the combination of AES algorithm and RSA algorithm. The parameter taken for this experiment is the use of AES algorithm 256 bit-key length and AES algorithm 320 bit-key length in this hybrid algorithm.

Based on the Figure 6, the time execution for hybrid cryptography that uses AES 256-bit key took 0.006 seconds to execute, while for hybrid cryptography that uses AES 320-bit key took 0.048 seconds to execute. Here shows that there is a 0.042 seconds difference between both times taken. Thus, hybrid cryptography that uses AES 320-bit key has a longer execution time than the hybrid cryptography that uses AES 256-bit key. In terms of security, the higher the key length used, the more secure the algorithm is, as the attacker finds it hard and takes too much time to breach the algorithm.

Figure 6: Execution Time of Modified keybits Algorithms

Comparison of Execution Time Taken Between AES 320-bit Key Only and the Proposed Hybrid Algorithm.

For the third result, this experiment is done on the AES 320 bit-key length only and the proposed hybrid algorithm that uses AES 320 bit-key length. The time execution is taken for each algorithm and compared.

Figure 7 shown the time execution of hybrid cryptography that uses AES 320-bit key took 0.048 seconds to execute, while the AES 320-bit key alone took 0.008 seconds to execute. Here shows the hybrid cryptography that uses AES 320-bit key has a longer execution time than the AES 320-bit key only. This comparison is actually to prove which algorithm is better and this experiment shows that the hybrid cryptography is better. Even though the computational time of hybrid cryptography is longer that the non-hybrid, but it will ensure the security. It is because, the higher the key length used, the more secure the algorithm is, as the attacker finds it hard and takes too much time to breach the algorithm. Hence, it is better to implement the modified AES and RSA hybrid algorithm.

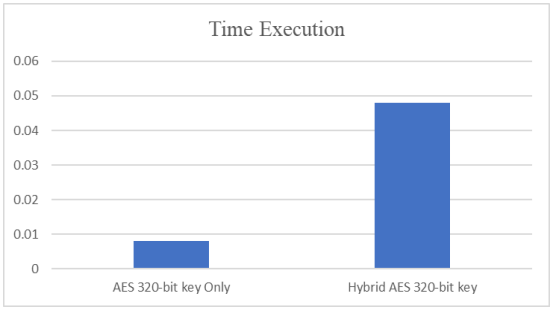

Figure 7: Comparison of Execution time taken between AES 320-bit and Proposed Hybrid Cryptography

Complexity of the Brute Force Attack

Below is the length keys of the AES algorithm and the numbers of operations required to try all of the possible keys to crack the algorithm. The bigger the key length used, the higher the number of operations that it takes to brute force the algorithm.

Table 2: Complexity of Brute Force Operations

	Key Length	No. of Operations
AES	128	2^{64}
	192	2^{192}
	256	2^{256}
	320	2^{320}

Based on the Table 2 it is believed that it would make the attacker difficult to crack the algorithm through Brute Force attack as the key length used is bigger than usual key length. As the key length increased, the number of rounds also increasing. Thus, it will take too much time to attack the algorithm.

V. CONCLUSION

As a conclusion, Hybrid Cryptography algorithm was successfully developed and had met the aim and objectives to increase the security performance for E-mail system purpose. The objective of this project is to design and develop a more secure E-mail system via modified AES and RSA hybrid algorithm and to evaluate the efficiency of the proposed technique in terms of E-mail security. While the aim of this project is to find an effective and efficient method to mitigate the threats from attacking the E-mail while enhancing the security of the E-mail system.

Through the development of this algorithm, all the objectives have been achieved. Testing and analysis have been done by compare the time computational process of the algorithm based on its key length used. From this project, the developer has gained a whole new knowledge in understanding the cryptography algorithm and also learnt new things in fixing the errors that occurred while completing this project.

VI. LIMITATIONS OF THE PROJECT

Limitations of this project is to link the E-mail message to the algorithm so that the algorithm can encrypt the message when it is being transmitted through the server. It is because, there are very few references about how to link the message to the algorithm, so it is hard to pursue to that extend. Besides, the public key of RSA also cannot be shared to the decryption part at the receiver due to the need of the key exchange in the E-mail server. This is because, the encryption part is at the server while the decryption part is at the receiver and the key exchange is needed so that the receiver can get the public key of RSA.

Other limitations of this project are the memory allocation error that occur when executing the algorithm. It is because of the invalid access to the memory. It is not wrong but because of the Windows feature that does not allowed it. Hence, the algorithm needs to be changed to a new one.

VII. RECOMMENDATIONS

The recommendation for this project is to continue it with the decryption part and implement it into the E-mail system. For the decryption part, ensure that the RSA public key is able to reach the receiver so that the E-mail message can be decrypted.

VIII. ACKNOWLEDGEMENT

In the name of Allah, the Most Gracious and the Most Merciful. Alhamdulillah, praise and thanks to Allah SWT, for all the graces and blessings and Selawat and Salam to the Prophet Rasulullah SAW, hopefully His syafa'at will be abundant in days later. A great honor to all the lecturers in UiTM Shah Alam for their patience, help and kind advice during the process of completing the project.

BIBLIOGRAPHY

1. Abikoye, O. C., Haruna, A. D., Abubakar, A., Akande, N. O., & Asani, E. O. (2019). SS symmetry for Information Security. 1–16.
2. Gamido, H. V., Sison, A. M., & Medina, R. P. (2018). Modified AES for text and image encryption. Indonesian Journal of Electrical Engineering and Computer Science, 11(3), 942–948.

https://doi.org/10.11591/ijeecs.v11.i3.pp942-948

3. Gavankar, S., & Vidhani, S. (2017). Email Security System. 8(3), 347–351.
4. Joseph Amalraj, A., & John Raybin Jose, J. (2016). A study of secured E-mail security system using certificateless cryptography and domain name system. International Journal of Control Theory and Applications, 9(26), 267–273.
5. Kumar, P., & Rana, S. B. (2016). Development of modified AES algorithm for data security. Optik, 127(4), 2341–2345. https://doi.org/10.1016/j.ijleo.2015.11.188
6. Lerner, A., Zeng, E., & Roesner, F. (2017). Confidante: Usable Encrypted Email: A Case Study with Lawyers and Journalists. Proceedings - 2nd IEEE European Symposium on Security and Privacy, EuroS and P 2017, 385–400. https://doi.org/10.1109/EuroSP.2017.41
7. Liu, Y., Gong, W., & Fan, W. (2018). Application of AES and RSA Hybrid Algorithm in E-mail. Proceedings - 17th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2018, 701–703. https://doi.org/10.1109/ICIS.2018.8466380
8. Plata, I. T., Panganiban, E. B., & Bartolome, B. B. (2019). A security approach for file management system using data encryption standard (DES) algorithm. International Journal of Advanced Trends in Computer Science and Engineering, 8(5), 2042–2048. https://doi.org/10.30534/ijatcse/2019/30852019
9. Purevjav, S., Kim, T., & Lee, H. (2016). Email encryption using hybrid cryptosystem based on Android. International Conference on Advanced Communication Technology, ICACT, 2016-March, 426–429. https://doi.org/10.1109/ICACT.2016.7423418
10. Rani, N. (2015). Suspicious Email Detection System via Triple DES Algorithm : Cryptography Approach. 4(5).
11. Rawdhan, F. A., & Ibrahim, M. K. (2017). Enhancement of Email Security Services. International Journal of Scientific & Engineering Research (IJSER), 8(1), 2090–2095.
12. Ruoti, S., Andersen, J., Dickinson, L., Heidbrink, S., Monson, T., O'Neill, M., Reese, K., Spendlove, B., Vaziripour, E., Wu, J., Zappala, D., & Seamons, K. (2019). A usability study of four secure email tools using paired participants. ACM Transactions on Privacy and Security, 22(2). https://doi.org/10.1145/3313761
13. Sabir, M. Z., & Yousaf, M. (2018). Design and implementation of an end-to-end web based trusted email system. Procedia Computer Science, 141, 231–238. https://doi.org/10.1016/j.procs.2018.10.176
14. Singh, P., Arora, S., Williamson, K., & Atrey, P. K. (2017). C3Email: A method for securing emails from service providers. 2017 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2017, 2017-Janua, 2170–2175. https://doi.org/10.1109/SMC.2017.8122941
15. Wu, J., Long, Y., Huang, Q., & Wang, W. (2017). Design and Application of IBE Email Encryption Based on Pseudo RSA Certificate. 2016 12th International Conference on Computational Intelligence and Security (CIS), 282–286. https://doi.org/10.1109/cis.2016.0071