SAGA: Journal of Technology and Information Systems
Vol 2, Issue 3, August 2024, Page 304-310
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.v2i3.359

Comparative Analysis of Performance of the Encryption and Decryption Times of Cryptography

Made Yoga Mahardika[1], Agung Triayudi[2]*

Faculty of Communication and Information Technology, Universitas Nasional, Jakarta, Indonesia

Author Email: madeyogamahardika123@gmail.com[1], agungtriayudi@civitas.unas.ac.id[2*]

Abstract. Cryptography is the study of mathematical techniques to maintain information security, including the encryption process to protect data and decryption to return it to a readable form. The Rivest Shamir Adleman (RSA) algorithm and Elliptic Curve Cryptography (ECC) are two asymmetric key algorithms that are often used. RSA relies on large number factorization for security, while ECC uses elliptic curves that require more complex computation but are more efficient in resource usage. This study aims to compare the performance of the two algorithms in terms of encryption and decryption time, and analyze efficiency based on various key sizes and data amounts. The research method includes measuring encryption and decryption time with different data inputs using RSA and ECC at various key sizes. Experiments were conducted with data inputs of 128-bit to 512-bit and testing was conducted to measure the speed of each algorithm in the same situation. The results showed that RSA was faster in the encryption process than decryption, while ECC had a faster decryption time than encryption. However, the overall processing time for ECC is longer than RSA, especially when the key size is increased. These results provide insight into the advantages and disadvantages of each algorithm, which can be used as a basis for consideration in selecting algorithms for security applications that require high efficiency.

Keywords: Cryptography, Decryption, ECC, Encryption, RSA

1 Introduction

Cryptography is a science that studies mathematical techniques related to aspects of information security such as level of confidence, data integrity, entity authentication, and authentication of data authenticity [1].

The cryptographic process involves encryption and decryption. The main purpose of encryption is to protect the confidentiality of digital data stored in a computer system or transmitted over the internet or other computer networks. Meanwhile, decryption is the process of taking coded or encrypted text or data and changing it back into text that can be read and understood by you or a computer [2].

A cryptographic algorithm is said to be safe if the mathematical equations that describe the operation of the cryptographic algorithm are so complex that the algorithm is impossible to solve analytically. In addition, the time required to crack ciphertext exceeds the length of time the information must be kept confidential [3]. Currently the algorithms that are well known for their level of security are RSA and ECC. Much research has been done on these two algorithms.

According to [4] the purpose of each encryption and decryption algorithm is to make attacks on data difficult so that data transfer from sender to recipient is not interrupted because we encounter many incidents involving cybercrime (computer-oriented crime) that attacks data or networks.

Systems based on elliptic curves are an effective alternative to the RSA cryptosystem, because they involve a different mathematical approach. Elliptic curve cryptography is preferred in use because it requires a smaller key size despite its mathematical complexity compared to other algorithms. The ECC algorithm is faster than RSA because it requires less computation. ECC is more advantageous than RSA due to low memory usage, low CPU consumption, and shorter key size compared to RSA [5][6][7][8].

Longer RSA key lengths lead to the least secure algorithms due to the potential for side-channel attacks. Operations in RSA are relatively faster than ECC but in terms of security ECC is stronger than RSA. RSA is fast at encryption but slow at decryption, while ECC is slow at encryption but fast at decryption, overall ECC outperforms RSA in terms of performance and security [9][10][11].

The algorithms that will be used in this research are RSA and ECC. From these two algorithms, different efficiencies may be obtained, where the efficiency of the cryptographic algorithm depends on the speed of the encrypting and decrypting process.

SAGA: Journal of Technology and Information Systems
Vol 2, Issue 3, August 2024, Page 304-310
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.v2i3.359

Based on this, research will be carried out regarding comparative performance analysis between the RSA and ECC algorithms in terms of encryption and decryption times for the RSA and ECC algorithms. It is hoped that the results of this research will provide deeper insight into the advantages and disadvantages of each algorithm, so that it can assist security system developers in selecting the algorithm that best suits their application needs.

2 Methodology

The author uses literature studies, namely the internet, books, journals, and articles, as sources of information to find out the problems discussed in this study. This study uses the RSA and ECC algorithms implemented using the Python programming language. Furthermore, encryption and decryption experiments were carried out with different key size variations, namely using key sizes of 112-bit, 128-bit, 192-bit, 224-bit, 256-bit, 384-bit, 521-bit, 1024-bit, 2048-bit, 3072-bit, 4096-bit, 5120-bit, 6144-bit, 7680-bit, and 15360-bit. The selection of this key length is based on the highest level of security for the key length in RSA, which is 15360-bit equivalent to 521-bit key length in ECC.

2.1 Method of Collecting Data

So based on this, this study uses only the key length of RSA and ECC, where RSA has tested up to a key length of 15360-bits, as well as testing for ECC up to the same key length of 15360-bits. Then to determine the plaintext character, the number of plaintext characters, and the bit size of the plaintext character, the author uses the following characters:

Table 1. Plaintext Characters

Plaintext Character Count	Plaintext Character Bit Size	Plaintext Characters
64 Character	512-bit	abcdefghijklmnopqrstuvwxyzA BCDEFGHIJKLMNOPQRSTU VWXYZ012345678910

2.2 Data Source

Primary data was obtained directly from the results of encryption and decryption experiments using RSA and ECC, in the form of the time required for the process with various key and data sizes.

2.3 Research Design

This study uses a quantitative approach with an experimental method to measure and analyze the encryption and decryption time of the RSA and ECC algorithms. In addition, this study uses an experimental approach to measure and analyze the encryption and decryption time of the RSA and ECC algorithms. Meanwhile, the design of this study is a comparative experiment, with the aim of comparing the performance of the encryption and decryption time between the RSA and ECC algorithms. Furthermore, the experiment in this study was carried out by measuring the time required for encryption and decryption using various key sizes and data to obtain comprehensive and in-depth results.

2.4 Tools and Materials

The following software and hardware were used in this study:

a. Hardware: Laptop with 12th Gen Intel(R) Core (TM) i7 12650H 2.30 GHz 10 Core processor specifications, 16 GB RAM
b. Software Program: Programming Language: Python 3.12.2
c. Time Measurement Library: import time
d. IDE: Visual Studio Code
e. Operating System: Windows 11
f. Test Data: Plaintext character texts to be encrypted and decrypted 512-bit with 64 bytes size.

2.5 Research Limitations

SAGA: Journal of Technology and Information Systems
Vol 2, Issue 3, August 2024, Page 304-310
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.v2i3.359

This study has several limitations, including:

a. Algorithm Scope: This study only compares two cryptographic algorithms, namely RSA and ECC, and does not cover other algorithms.
b. Data Size Variation: Measurements are only performed on certain data sizes and may not cover all real application scenarios.
c. Device Specifications: Results may depend on the hardware specifications used, so they may differ if performed on devices with different specifications.

2.6 Design and Analysis

a. Cryptographic Analysis Phase: At this stage, the requirements for the research will be carried out. The author only uses a few references as research materials, such as:
   1. Analyze the RSA and ECC algorithms by describing them in a flowchart. Study the calculation model of the RSA and ECC algorithms in more depth, as well as their working techniques.
   2. Prepare the hardware and software that will be used to support the research.
b. Implementation and Evaluation Phase: At this stage, the author implements and evaluates the calculation of the RSA and ECC algorithms using visual studio code, to analyze the encryption and decryption times.
c. Results and Comparative Analysis Phase: At this stage, the author describes the results that have been analyzed and then compares them based on the encryption and decryption times, then draws conclusions.

3 Results and Discussion

Time is one factor in carrying out the encryption and decryption process. With time, the speed of the encryption and decryption process can be known. Meanwhile, how encryption works is the process of changing raw data, which is usually called plaintext, into encrypted data that is difficult to read, also called cipher text. This is done using mathematical methods and algorithms. This encryption process involves an algorithm, decryption, and key length. So the encryption algorithm uses data and keys during the encryption process, while decryption changes the encrypted message with the help of the key.

3.1 Measurement Results

The following table and graph present the relationship between key length and the time required for the encryption and decryption processes.

Table 2. Encryption and Decryption Processing Time (512-Bit Data Input)

Input data (512 bit) with 64 characters						
Keylength	RSA Encryption Time	RSA Decryption Time	Total Time	ECC Encryption Time	ECC Decryption Time	Total Time
112 bit	0.000013	0.001308	0.001321	0.636914	0.422003	1.058917
128 bit	0.000016	0.001738	0.001754	0.881859	0.581162	1.463021
192 bit	0.000016	0.003830	0.003846	2.710610	1.861974	4.572583
224 bit	0.000045	0.004604	0.004649	2.984255	2.034278	5.018532
256 bit	0.000013	0.006272	0.006285	3.790915	2.574052	6.364967
384 bit	0.000017	0.016324	0.016341	10.890596	7.329590	18.220185
521 bit	0.000011	0.030818	0.030829	22.606767	15.250364	37.857131
1024 bit	0.000014	0.172020	0.172034	83.222765	56.349696	139.572461
2048 bit	0.000012	1.059800	1.059812	463.592348	310.996716	774.589063
3072 bit	0.000011	3.188176	3.188187	1287.647282	862.517794	2150.165076
4096 bit	0.000012	7.379044	7.379056	2873.873687	1944.198424	4818.072111
5120 bit	0.000045	13.688820	13.688865	5359.814485	3576.303155	8936.117640
6144 bit	0.000036	22.694346	22.694382	8865.498197	5929.525088	14795.023284
7680 bit	0.000016	44.969971	44.969987	16423.644953	10959.883191	27383.528144
15360 bit	0.000020	336.937859	336.937879	115301.382459	76967.928163	192269.310622

SAGA: Journal of Technology and Information Systems
Vol 2, Issue 3, August 2024, Page 304-310
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.v2i3.359

Based on the research, with 512 bit and 64 character data input, the encryption and decryption process time increases as the key length increases. In the RSA algorithm, encryption is faster than decryption, with a large time difference. In contrast, in the ECC algorithm, encryption takes longer than decryption, but the time difference is small. Overall, the total encryption and decryption time in the ECC algorithm is longer than that of RSA, indicating that RSA is superior in terms of speed.

The following is a comparison table of RSA and ECC cryptography based on the information obtained:

Table 3. Comparison of RSA and ECC Cryptography

Aspect	RSA Algorithm	ECC Algorithm
Encryption Time	Faster than decryption	Longer than decryption
Decryption Time	Slower than encryption	Faster than encryption
Encryption and Decryption Time Difference	Huge time difference between encryption and decryption	Small time difference between encryption and decryption
Total Processing Time	Faster than ECC	Slower than RSA

3.2 Discussion

The following table compares the advantages and disadvantages of RSA and ECC cryptography based on the information obtained:

Table 4. Advantages and Disadvantages of RSA and ECC Cryptography

Aspect	RSA Algorithm	ECC Algorithm
Superiority	- Encryption is faster than decryption, especially with long keys. - More efficient for encryption in practical use, since many users are encrypting (using small exponents speeds up encryption). - On the same data, RSA is faster in total encryption and decryption processing time than ECC.	- Smaller key size, providing the same security as a longer RSA key. - Faster decryption than encryption because the math is simpler (inverse is faster than dot multiplication). - More efficient in that the key size is smaller, reducing the computational resources required.
Weakness	- Decryption is slower than encryption, mainly because the large private exponent makes the inverse process slower. - Requires more computational resources when using long keys, thus slowing down encryption and decryption.	- The encryption process takes longer than decryption because the mathematical operations are more complex on elliptic curves, especially in repeated dot multiplication. - Even though the key size is smaller, the total encryption and decryption time is still longer than RSA, especially when the key is shorter than RSA.
Security Level	- The longer the key, the greater the level of security, but this results in longer encryption and decryption times.	- With shorter keys, ECC can achieve the same level of security as RSA, but with a longer processing time.
Time Efficiency	- At the same key length, RSA excels in total processing time efficiency (encryption and decryption).	- ECC is slower in encryption and decryption than RSA even though it uses shorter keys, but is still faster in decryption than encryption.
Key Length Effect	- Encryption and decryption times increase as the key length increases, with decryption being the slowest process.	- Increasing the key length makes encryption time longer, although decryption is still faster than encryption.
Improvement Recommendations	- Using more efficient modular exponentiation optimization methods or reducing the size of private exponents without sacrificing security. - Using hybrid or parallel processing methods to speed up decryption time.	- Optimizing mathematical operations on dot multiplication to speed up encryption, for example by using certain elliptic curve optimization techniques. - Using more efficient keys or hardware-based computational operations that can speed up encryption.

SAGA: Journal of Technology and Information Systems
Vol 2, Issue 3, August 2024, Page 304-310
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.v2i3.359

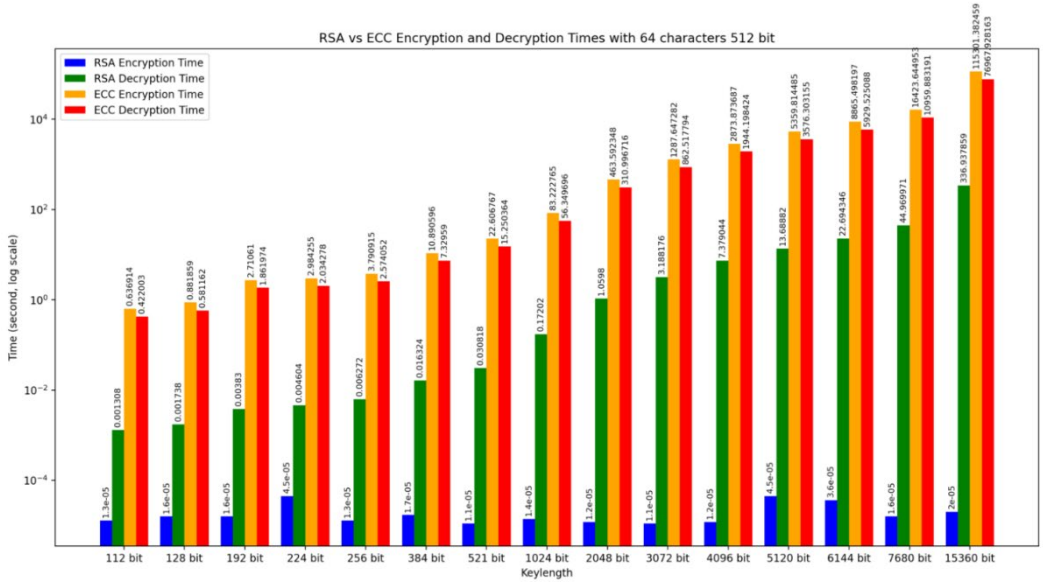

Figure 1. Encryption and Decryption Processing Time (512-Bit Data Input)

Based on table 2 and Figure 1, testing with the same key length shows that the results of the RSA algorithm for the encryption and decryption process time show that the encryption time is faster than the decryption time. Meanwhile, for the ECC algorithm, all research results show that the encryption time is longer than decryption.

In the RSA algorithm, decryption time takes a long time, especially at very long key lengths. This is caused by several factors that influence RSA decryption time. The RSA encryption process involves a fast exponentiation operation, namely fast exponentiation modulo (n), where (n) is the public key modulus. This operation is faster than the inverse modulo operation involved in RSA decryption, which requires looking up the private exponent value to invert the exponentiation operation.

The complexity of the RSA algorithm also differs between the encryption and decryption processes. The encryption process has lower mathematical complexity than the decryption process. The exponentiation operation involved in encryption is a simpler operation and is faster to perform.

Overall, this design maximizes operational efficiency on the sender side, while maintaining high security on the receiver side. The main reason encryption is made as fast as possible while decryption can be slower is for efficiency in daily usage practices and to maintain the security of the decrypted data. However, it should be noted that slower decryption is not intentionally designed to slow down time, but is a consequence of using a larger exponent for security reasons.

Thus, RSA encryption is made as fast as possible for operational efficiency, because many users are encrypting and sending messages. On the other hand, decryption is slower because it involves a larger exponent, which provides additional security. A longer time for decryption is acceptable because it is done less frequently and by the recipient having the private key.

Furthermore, the encryption time for the ECC algorithm shows that it is longer than the decryption time. This is due to several factors, namely differences in the mathematical operations carried out in each process. The ECC encryption process involves more complex mathematical operations, such as performing repeated dot multiplication using random private key numbers to get the public key point. This requires many complex mathematical steps, especially for large points on an elliptic curve. This process requires more computing resources and time, because it has to perform quite complex operations. Meanwhile, for the decryption process, the ECC algorithm has simpler mathematical operations compared to the encryption process. This is due to calculating the inverse of the dot product performed during encryption. This can be done more quickly than dot multiplication itself, making the operation simpler. So the time required for decryption tends to be shorter.

Some research results that support [16][5][6] ECC rely on mathematical computations to encrypt and decrypt, its strength depends on complexity, so the calculations are very large. The use of elliptic curve cryptography is preferred because it requires a smaller key size although it involves complexity in mathematics. The ECC algorithm's computational complexity is faster than the RSA method because it requires less computation.

This increase in encryption and decryption time can be seen in Figure 1. The comparison between the RSA and ECC algorithms is clearly visible, showing that ECC takes longer encryption and decryption times with shorter key usage than RSA. This is because the use of long keys in RSA causes the encryption and decryption processes to be slower and require more computing resources, while the use of shorter keys in ECC causes the encryption

SAGA: Journal of Technology and Information Systems
Vol 2, Issue 3, August 2024, Page 304-310
ISSN: 2985-8933 (Media Online)
DOI: 10.58905/SAGA.v2i3.359

and decryption processes to be faster and more efficient, and requires fewer resources. Computing power compared to RSA.

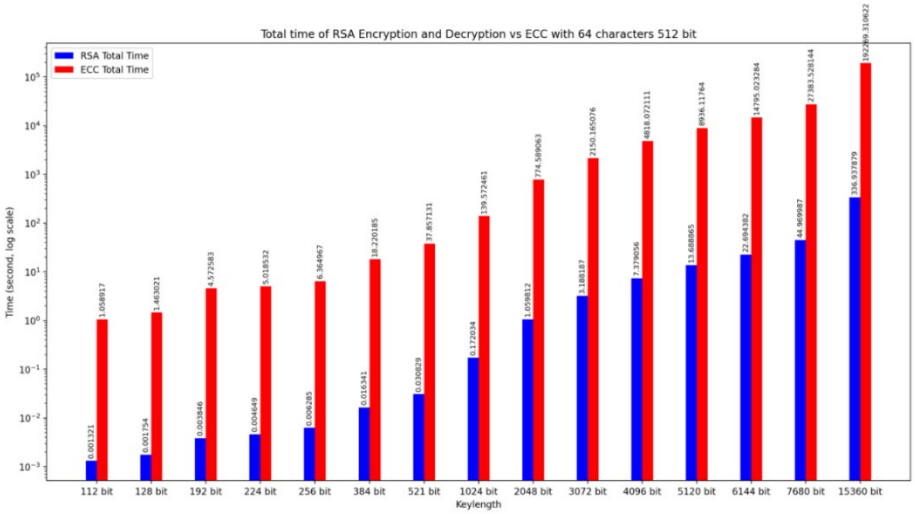

Figure 2. Total Time of Encryption and Decryption Process (512-Bit Data Input)

In Figure 2, the comparison between the RSA and ECC algorithms is clearly visible, showing that ECC requires longer encryption and decryption time than RSA with the use of the same key length as RSA. The longer the key, the longer the total encryption and decryption time. The total time of ECC is superior in terms of security level only. When ECC has the same level of security as RSA but requires a shorter key length than RSA, the total time is clearly inefficient if tested with the same key length, too complicated and long, so the total time is superior in terms of security level, which is certain if the time is that long and the key length is that large, the level of security is certainly also high. Furthermore, the increasing number of characters also affects the encryption and decryption time. This is because the more characters, the more things that must be encrypted and decrypted or the more things that must be done.

4 Conclusion

The results of this study are in line with previous studies showing that in the RSA algorithm, the longer the key used, the longer the processing time, especially in the decryption process and to obtain a high level of security, the use of a long key is required. In contrast, in the ECC algorithm, the use of a shorter key can still provide a high level of security, but the encryption time is longer than the decryption process, which reflects the greater complexity of the ECC encryption process.

The RSA and ECC algorithms, as asymmetric key algorithms, use different keys for encryption and decryption, with RSA relying on factoring non-prime numbers into their prime factors, while ECC relies on solving the elliptic curve discrete logarithm problem.

From this study, it can be seen that the RSA encryption time is faster than the decryption time, while the ECC encryption time is actually longer than the decryption time. In addition, the total encryption and decryption time using the same key length shows that RSA is faster than ECC because of its lower complexity, while ECC takes longer because of its higher complexity. Larger key lengths in both algorithms increase encryption and decryption processing time. To overcome these drawbacks, recommendations for RSA include the use of smaller private exponents or modular exponential optimization algorithms to speed up decryption. Hybrid approaches that combine RSA with other encryption methods can also be applied to speed up the decryption process. Meanwhile, in ECC, encryption optimization can be done using more efficient dot multiplication techniques, such as dot compression or hardware-based methods. The use of hybrid algorithms or parallel processing can also be considered to increase encryption speed without sacrificing security.

References

[1] H. Mukhtar, Kriptografi untuk Keamanan Data, 1st ed. Yogyakarta: Deepublish, 2018.

[2] Dr. M. Gobi, R. Sridevi, and R. Rahini priyadharshini, "A Comparative Study on the Performance and the Security of RSA and ECC Algorithm," International Journal Of Advanced Networking and Applications (IJANA), pp. 168–171, Mar. 2015.
[3] J. Simarmata, Sriadhi, and R. Rahim, Kriptografi Teknik Keamanan Data dan Informasi, 1st ed. Yogyakarta: CV. ANDI OFFSET, 2019.
[4] Monika, T. Tomar, V. Kumar, and Y. Kumar, "IMPLEMENTATION OF ELLIPTIC - CURVE CRYPTOGRAPHY," International Journal of Electrical Engineering and Technology (IJEET), vol. 11, no. 2, pp. 178–189, Apr. 2020, Accessed: Jun. 08, 2024. [Online]. Available: http://iaeme.com/Home/journal/IJEET
[5] K. Sujatha, A. Arjuna Rao, P. V. Nageswara Rao, and L. V. Rajesh, "Renowned Information Security Algorithms: A Comparative Study," International Journal of Engineering Research & Technology (IJERT), vol. 5, no. 2, pp. 216–224, Feb. 2016, Accessed: Jun. 30, 2024. [Online]. Available: http://www.ijert.org
[6] M. Wameedh Abdulnabi, R. A. Muhajjar, and M. Al-Zubaidie, "Elliptic Curve Implementation and its Applications: A Review," Iraqi Journal of Intelligent Computing and Informatics (IJICI), vol. 2, no. 2, pp. 90–100, Dec. 2023.
[7] N. J. Gbètoho Saho and E. C. Ezin, "Comparative Study on the Performance of Elliptic Curve Cryptography Algorithms with Cryptography through RSA Algorithm," Proceedings of CARI 2020, Aug. 2020, Accessed: Jun. 30, 2024. [Online]. Available: https://hal.science/hal-02926106
[8] D. M. Mani and N. P H, "A Comparison Between RSA And ECC In Wireless Sensor Networks," International Journal of Engineering Research & Technology (IJERT), vol. 2, no. 3, pp. 1–5, Mar. 2013, Accessed: Jul. 03, 2024. [Online]. Available: www.ijert.org
[9] Dr. M. Kaur Bhatia, R. S. Rajpurohit, Gulafshan, and B. Joseph, "A Study of Cryptographic Algorithms," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 10, no. 12, pp. 170–181, Dec. 2022, Accessed: Jul. 03, 2024. [Online]. Available: https://doi.org/10.22214/ijraset.2022.47848
[10] Ms. R. S. Satpute, Prof. R. S. Mangrulkar, and Prof. A. N. Thakare, "Performance Analysis of Wireless Sensor Networks using Elliptical Curves Cryptography," International Journal of Engineering Research & Technology (IJERT), vol. 3, no. 7, pp. 773–780, Jul. 2014, Accessed: Jul. 03, 2024. [Online]. Available: www.ijert.org
[11] J. Bao, "Research on the Security of Elliptic Curve Cryptography," Advances in Economics, Business and Management Research, vol. 215, pp. 984–988, 2022.
[12] E. Setyaningsih, Kriptografi dan Implementasinya menggunakan MATLAB, 1st ed. Yogyakarta: CV. ANDI OFFSET, 2020.
[13] R. Munir, Kriptografi, 2nd ed. Bandung: Informatika Bandung, 2019.
[14] S. K. Verma and Dr. D. B. Ojha, "A Discussion on Elliptic Curve Cryptography and Its Applications," IJCSI International Journal of Computer Science Issues, vol. 9, no. 1, pp. 74–77, Jan. 2012, Accessed: Jul. 03, 2024. [Online]. Available: www.IJCSI.org
[15] M. Rafeek Khan et al., "Analysis of Elliptic Curve Cryptography & RSA," Journal of ICT Standardization, vol. 11, pp. 355–378, Nov. 2023.
[16] Y. Yan, "The Overview of Elliptic Curve Cryptography (ECC)," J Phys Conf Ser, pp. 1–8, 2022.
[17] Dr. B.S Dhaliwal and V. Soi, "Wireless Sensor Networks using Elliptical Curves Cryptography: Performance Analysis," International Journal of Research in Engineering and Applied Sciences (IJREAS), vol. 6, no. 6, pp. 284–291, Jun. 2016, Accessed: Jun. 30, 2024. [Online]. Available: http://www.euroasiapub.org
[18] A. Karki, "A Comparative Analysis of Public Key Cryptography," International Journal of Modern Computer Science (IJMCS), vol. 4, no. 6, pp. 30–35, Dec. 2016, Accessed: Jun. 30, 2024. [Online]. Available: http://ijmcs.info
[19] Gururaja. H.S, M. Seetha, A. K. Koundinya, Shashank. A.M, and Prashanth. C.A, "Comparative Study and Performance Analysis of Encryption in RSA, ECC and Goldwasser- Micali Cryptosystems," International Journal of Application or Innovation in Engineering & Management (IJAIEM), vol. 3, no. 1, pp. 111–118, Jan. 2014, Accessed: Jul. 04, 2024. [Online]. Available: www.ijaiem.org
[20] B. K. Alese, Philemon E. D., and S. O. Falaki, "Comparative Analysis of Public-Key Encryption Schemes," Int J Eng Technol, vol. 2, no. 9, pp. 1552–1568, 2012.