TEKNOKOM : Jurnal Teknologi dan Rekayasa Sistem Komputer
Vol. 4, No. 1, March 2021, pp. 1 - 6
ISSN: 2686-3219, DOI: 10.31943/teknokom.v4i1.53

RAW DATA SECURITY BY USING ELGAMAL AND SHA 256 PUBLIC KEY ALGORITHM

Indra Surya Permana[1], Taufik Hidayat[2], and Rahutomo Mahardiko[3]*
[1]Department of Economic Science, Universitas Nahdlatul Ulama Cirebon, Cirebon, Indonesia
[2]Department of Computer Engineering, Universitas Wiralodra, Indramayu, Indonesia
[3]Department of Software Service, Platinumetrix Pte. Ltd, Jakarta, Indonesia

ARTICLE INFO
History of the article:
Received March 21, 2021
Revised April 2, 2021
Accepted April 2, 2021
Published April 3, 2021

Keywords:
Cryptography
ElGamal
SHA256
Raw Data Security

ABSTRACT

The development of information technology has grown exponentially and various of data collections and its method has been obtained. In the era of big data, data has now become an asset that held important values, while in the implementation of data delivery, it clearly is not always safe. One of the method to secure data delivery is data encryption using Cryptography. Cryptography provides an encryption service to secure data delivery by transforming it to random values so that it can no longer be read. The goal in this study was to produce an application that could be used to encrypt data, using ElGamal's cryptography method and hash checking using the SHA256 algorithm. After encryption, to ensure the encrypted data is still the original data without any changes or manipulation by unauthorized 3rd party then done by checking the hash generated using SHA256 algorithm. The data used in this study was a sample of raw data from the ATPWTP survey (ability to pay and willing to pay) conducted by the BPS Cirebon (Central Statistics) in 2019 and the data was in the form of Excel and txt files. The encryption process resulted in a cipher larger than the plaintext and takes longer for the data encryption process than during the data cipher decryption process.

Correspondece:
Rahutomo Mahardiko
Department of Software Service,
Platinumetrix Pte. Ltd, Indonesia,
Email : rahutomo.mahardiko@gmail.com

INTRODUCTION

Along with advances in the field of information technology, data became an invaluable asset. Numerous methods are developed in terms of data security, and cryptography is one of them to transform data [1], [2], as an attempt to hide semantic content from a message or data from being incomprehensible, preventing it from unknown changes or preventing it from unauthorized use [3], [4], [5].

Badan Pusat Statistik/Central Bureau of Statistics (BPS) Cirebon as the Working Unit of the Central Statistics Agency is a government agency based on Government Law UU No. 16/1997 to organize statistical activities and has a role to provide data needs, not only for the government but also for the community. BPS obtained various data from censuses or surveys that self-conducted and also from other departments or government agencies as secondary data. It has a vital role, considering the results of the data collection is used as a consideration to make for government policy [6].

Unfortunately, most of data delivery in BPS were less secure delivered because they did not use encryption and open for public (generally in the form of excel spreadsheet). Therefore, to support the confidentiality of raw data, the method such as Cryptographic is needed to provide more reliable and to ensure the security of data delivery [7], [8]. Cryptographic Systems is a set consisting of encryption algorithms, decryption algorithms, key rooms, all plaintext and possible ciphetext [9], [10]. A cryptographic algorithm is said to be safe when it meets the following three criteria's: (a) Mathematical equations that describe the operation of cryptographic algorithms that are so complex that algorithms are impossible to solve analytically. (b) The cost of solving the cipher text exceeds the value of the information contained in the ciphertext. (c) The time required to break the cipher text exceeds the length of time that such information must be kept confidential [11].

LITERATURE REVIEW

Modern cryptography deals with the construction of powerful information systems against malicious attempts to make these systems deviate from specified functions. Indeed, the scope of modern cryptography is vast, and in stark contrast to "classic" cryptography (which focuses on one issue to enable confidential communication through insecure communication media) [12], [13].

Cryptography aimed to provide security services in the form of (a) confidentiality aimed at preventing messages from being read by others [14], [15], [16]. (b) Data Integrity: a service that guarantees that messages are genuine/intact or have never been altered/manipulated during delivery. (c) Authentication is a service that is related to the identification of the truth of the communicating party. (d) Anti-denial which is a service to prevent the parties communicating from denying the message sent or received [12], [17].

Problem Analysis in this study is to merge between elGamal's public key algorithm cryptographic method and SHA256 [18]. ElGamal algorithm is used to encrypt and decrypt raw data files then generate first hash value using SHA256 algorithm. After the encrypted data is received by the recipient, it should be decrypted [19]. As for the decryption process, it should be matched hash first that sent by the sender with the aim to find out if the encrypted data has changed at the time of sender or is the original data of the sender [9]. Because if the data received does not have the same hash value it means that the data received (which has been encrypted) is not the original data or has changed [2], [20]. Therefore it could not be decrypted and to be read by unauthorized receiver [21].

RESEARCH METHODS

ElGamal Cryptographic Algorithm

The ElGamal algorithm was created by Taher ElGamal in 1984. This algorithm is used for digital signatures, but then modified so that it can also be used for encryption and decryption. As with most public key cryptographic algorithms, ElGamal Cryptographic algorithms use different keys to perform encryption and decryption [22], [23].

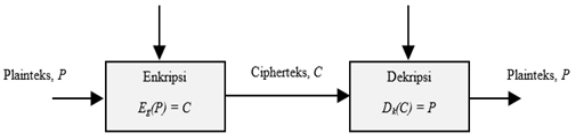


Figure 1. Algorithm ElGamal Processing [23]

ElGamal algorithm consists of 3 (three) processes, namely the process of key formation, encryption process and decryption process. ElGamal's algorithmic scheme requires a pair of keys to be used for the encryption and decryption process. For the Encryption process, it is raised from the values p, g and y (is a public key) while the decryption key consists of the value x (private), p. However, the encryption process time is longer than the decryption time due to mathematical calculations in the encryption process [22], [24], [25]. The steps and requirements that must be met for key generation are as follows:

1. Select any prime number p
2. Select a random number g with the condition $g < p$
3. Select a random number x with the condition $1< x < p-2$
4. Calculate value :

$$y = g^x \bmod p \tag{1}$$

The value p, g, y is a public key is not confidential, while the value x is a private key used to decrypt raw data and that only held by the party receiving raw data.

Encryption Processing

In the encryption process is used public key triplets (p, g, and y). The steps in encryption are as follows:

1. Cut plaintext into blocks m1, m2, m3, m4, mn value every block inside the hose $[0, p – 1]$
2. Convert message block values into ASCII (Decimal) values
3. Select a random number k, with conditions $1 > k > p – 1$ as much as m
4. Each m block is encrypted with the following formula:

$$a = g^k \bmod p \tag{2}$$

$$b = y^k m \bmod p \tag{3}$$

5. Arrange chipertext in the order a1, b1, a2, b2, an, bn.

Pairs of a and b are chipertext for message block m. The result obtained from the encryption process in the form of secret messages (chipertext), from the equation (2) and (3) it appears that a message block m will be two blocks of messages a and b. Therefore the result of the ciphertext will be twice the size of plaintext.

Decryption Processing

In ElGamal Algorithm, decrypting messages is done using private key (x,p). Here are the steps in decrypting:

1. Calculate Plaintext m with the following equation:

$$m_i = b\,(a^p)^{-1-x} \bmod p \tag{4}$$

2. The value m_i obtained from the equation (4) in the form of ASCII, then changed in plaintext form.
3. Arrange plaintext into order m1, m2, m3, ..., mn.
4. By rearranging it, we can then regain the original plaintext.

Secure Hash Algorithm (SHA)-256

The Secure Hash Algorithm (SHA) developed by the National Institute of Standards and Technology (NIST) was also designed on the same principle as MD4 and published as the Federal Information Processing Standard (FIPS 180) in 1993. The revised version was issued as FIPS180-1 in 1995 and is commonly referred to as SHA-1. When the revised version of SHA-1 was published, no details of weaknesses found in SHA-0 (originally SHA) were provided. SHA-1 returns a 160-bit hash value. In 2002, NIST produced a revised version of the standard known as FIPS180-2 and defined three new versions of SHA with core lengths of 256, 384 and 512 and were known as SHA-256, SHA-384, and SHA-512, respectively.

Hash function is a Cryptographic algorithm that compresses messages (compression) of any sized message into a fixed message digest. Some hash functions, including MD (Message Digest) 5, SHA-1, SHA-2, Keccak (SHA-3), RIPEMD, and so on.SHA256 is the 2nd (second) generation of SHA. In 2011 – 2015 SHA-1 was established as the main algorithm in the SSL (Secure Socket Layer) industry but since collution was discovered then in 2016 it was established SHA-2 to be the new standard in the use of SSL certificates. Collution is a security loophole where a different message can have the same hash. The hash function is one-way, meaning that a message that is compressed into a hash (cipher text) then the message cannot be restored to the original message.

The hash function h retrieves the message m with an arbitrary length and returns the core message h (m) with a fixed length. The hash function becomes an important part of cryptography, with the hash function of a message security can be tested for integrity, because two different messages (although only one character differ) will produce significantly different hashes.

Table 1. SHA256 example

Plaintext	Chipertext	Length
COVID19	c1dbef6a54e9cb78c4c411323ba8881e9b3e1d4b2b9f0802b4c1a98bce9eab87	64
cOVID19	12ee57c63805c252120321c7856f451fddb2b4a25174e9534ef2f70b0858dc99	64
C	6b23c0d5f35d1b11f9b683f0b0a617355deb11277d91ae091d399c655b87940d	64
c	2e7d2c03a9507ae265ecf5b5356885a53393a2029d241394997265a1a25aefc6	64

From table 1 above we can see even if only 1 letter change in the word "COVID19” to cOVID19 (with changes in the letter “C” to "c") will produce a whole hash different from the fixed length (64). Similarly, if the plaintext becomes only 1 (one) letter "C" or “c” it will be converted into hashes with a fixed length (64). SHA256 can be used to test the authenticity of a message, because of its fixed characteristics and case sensitive, whether the message has changed or not.

RESULTS AND DISCUSSION

Research Results

We present the implementation of ElGamal and SHA256 cryptographic algorithms in this article. The Steps of implementation are as follows:

1. Public Key Generation
2. Data encryption and decryption
3. Hash Generating

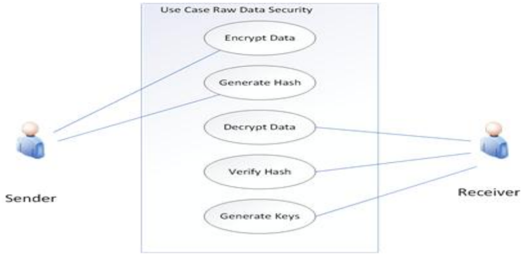


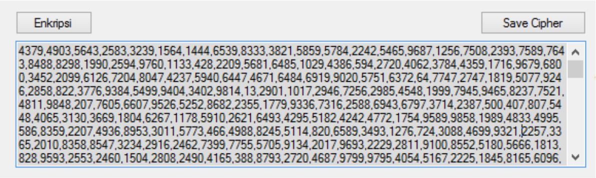
Figure 2. Processing raw data security

Public key is generated by the Receiver, while raw data encryption is done by entering the public key. Once raw data is encrypted, a hash generating process performed to ensure message security. Hash strings are sent along with raw encrypted data.

Raw data recipients who have been encrypted will check the received hash value, if the hash value contained in the encrypted raw data is the same as the received hash value then it can be ascertained that the data cipher does not change so that it can be continued for the data decryption process. After the data is decrypted, the raw data can be converted into a database. However, if the hash value is not the same as that sent by the sender, then the raw data from the encryption has changed, therefore the data received is not valid, so it cannot be continued to the decryption process.

Raw data encryption testing is done through an application. The encryption process is preceded by generating a public key. The public key is obtained from the recipient who will receive the encryption data. After the public key generate only then can be done raw data encryption. In this study, which is used for case studies is the data of ATPWTP survey results (ability to pay and willing to pay) as an excel spreadsheet. This data is intended to obtain an overview of the purchasing power of the community in Cirebon in 2019. Because all raw data is confidential for that reason it requires a process of data encryption so that it cannot be read by the authorities

Once raw data is encrypted then the cipher will be formed, this cipher is sent to the party that should receive. Figure 3 encryption processing.



```
4379,4903,5643,2583,3239,1564,1444,6539,8333,3821,5859,5784,2242,5465,9687,1256,7508,2393,7589,764
3,8488,8298,1990,2594,9760,1133,428,2209,5681,6485,1029,4386,594,2720,4062,3784,4359,1716,9679,680
0,3452,2099,6126,7204,8047,4237,5940,6447,4671,6484,6919,9020,5751,6372,64,7747,2747,1819,5077,924
6,2858,822,3776,9384,5499,9404,3402,9814,13,2901,1017,2946,7256,2985,4548,1999,7945,9465,8237,7521,
4811,9848,207,7605,6607,9526,5252,8682,2355,1779,9336,7316,2588,6943,6797,3714,2387,500,407,807,54
48,4065,3130,3669,1804,6267,1178,5910,2621,6493,4295,5182,4242,4772,1754,9589,9858,1989,4833,4995,
586,8359,2207,4936,8953,3011,5773,466,4988,8245,5114,820,6589,3493,1276,724,3088,4699,9321,2257,33
65,2010,8358,8547,3234,2916,2462,7399,7755,5705,9134,2017,9693,2229,2811,9100,8552,5180,5666,1813,
828,9593,2553,2460,1504,2808,2490,4165,388,8793,2720,4687,9799,9795,4054,5167,2225,1845,8165,6096,
```

Figure 3. Encryption Processing

Raw data that has been encrypted will only be in the form of irregular integer numbers. So that it can no longer be read. Before being sent to the authorities, the cipher raw data must be generated first in order for the hash string to form.

The generated Hash string is sent so that the recipient can match the raw data cipher with the formed hash. The recipient of the raw data cipher file will first match the hash value with the data received, if it is suitable then it can be decrypted with a Private Key.

After the cipher raw data is returned to plain raw data, then the data can be converted into a database or spreadsheet, so that the analysis can be done for publication.

The results obtained from this study are data in the form of cipher raw data that cannot be read by the authorities.

Table 2. Variable Testing Data

Variable Test	Plain Raw Data	Cipher Raw Data
Size	4 Kb	6 Kb
Readability	Readable	Unreadable
Security	Insecure	Secure

From the three test factors above obtained the following information:

1. The size of raw data cipher is much larger when compared to plain raw data, this is because every 1 (one) message block will be converted into 2 (two) cipher blocks.
2. Cipher raw data cannot be understood if it is not decrypted cipher.
3. In terms of security, raw data that has been encrypted is more secure to be sent to the recipient because it is not understandable.

For large files (bigger than 100 MB), it is recommended to split into several files, because the encryption process will be long, but the time required for the decryption process is much shorter when compared to the time required for the encryption process. Encryption and decryption were successfully performed, it was seen that the results of encryption can change raw data to be incomprehensible, and the decryption results showed the results of decryption according to the original data, without experiencing changes. For the encryption and decryption process is influenced by the size or small amount of data that is encryption, as well as the influence of the key used, the longer the key used, the longer the encryption and decryption process will be, but in terms of security it is considered more secure to use a long key.

Discussions

In IEEE digital library, ElGamal also is not a common research for the raw data security. We collected during 1985 – 2021, ElGamal topic in IEEE reached 85 papers and no research regarding raw data security using ElGamal was detected. It means the topic is still wide opened for the research. For an instance, the research published in IEEE focuses on digital signature scheme [26], [28], [30] for cryptosystem [26], [28], Cloud-ElGamal [27], the modification of ElGamal cryptosystem [29], matrix based and ElGamal in curve cryptography [31], and so on.

In ACM digital library, ElGamal also is not a common research for the raw data security recently. We collected during 1992 – 2021, ElGamal topic in ACM reached 6 papers and no research regarding raw data security using ElGamal was detected. For an example, the research published in ACM focuses on online voting using ElGamal cryptosystem [32], meta-ElGamal [33] signature schemes [34], [35] and so on.

We also checked that some researches on ElGamal were found without ACM and IEEE digital libraries. Data sharing with ElGamal increases the security through additive similarity [16]. Hybrid combination of ElGamal, AES and RSA for the cryptosystems [22] and ElGamal with Hill Cypher

4x4 for image security [23] were detected. The use of ElGamal can be found during the vector-module method for the radio [24], [36].

CONCLUSION

The problem collected on raw data security in Excel and plaintext (txt) files because of less secure. This means the data can be accessed publicly even though a confidential file. So that, we hypothesize the raw data should be encrypted to ensure the security. From the results of the research that has been done, the conclusions that can be drawn are ElGamal cryptographic algorithm and SHA256 checksum can be applied to secure raw data at the Office of the Central Statistics Agency of Cirebon Regency. However, because the mathematical formula used when encryption is more complex (involving two equations) results in the size of the data file resulting from encryption (cipher data) larger than the original file (plain data) and the encryption process is longer than the decryption process, but this can be overcome by splitting raw data into several smaller parts and increasing the capacity of the hardware and software used.

ACKNOWLEDGMENT

We, the authors, would like to thank Department of Economic Science, Universitas Nahdlatul Ulama Cirebon and Department of Computer Engineering Universitas Wiralodra for support of this research.

REFERENCES

[1] T. Wang, Y. Mei, W. Jia, X. Zheng, G. Wang, and M. Xie, “Edge-based differential privacy computing for sensor–cloud systems,” *J. Parallel Distrib. Comput.*, vol. 136, pp. 75–85, Feb. 2020.

[2] S. Madhavapandian and P. MaruthuPandi, “FPGA implementation of highly scalable AES algorithm using modified mix column with gate replacement technique for security application in TCP/IP,” *Microprocess. Microsyst.*, vol. 73, p. 102972, 2020.

[3] K. Rani and R. K. Sagar, “Enhanced Data Storage Security in Cloud Environment using Encryption , Compression and Splitting technique,” 2017.

[4] S. Patii and N. Rai, “An effectual information probity with two TPAS in cloud storage system,” *2017 Third Int. Conf. Sci. Technol. Eng. Manag.*, pp. 432–434, 2017.

[5] T. Hidayat and R. Mahardiko, “A Systematic Literature Review Method On AES Algorithm for Data Sharing Encryption On Cloud Computing,” *Int. J. Artif. Intell. Res.*, vol. 4, no. 1, pp. 49–57, Apr. 2020.

[6] V. K. Soman and V. Natarajan, “An Enhanced hybrid Data Security Algorithm for Cloud,” no. July, pp. 421–424, 2017.

[7] M. I. S. Reddy and A. P. S. Kumar, “Secured Data Transmission Using Wavelet Based Steganography and Cryptography by Using AES Algorithm,” *Procedia Comput. Sci.*, vol. 85, no. Cms, pp. 62–69, 2016.

[8] T. Hidayat, D. Sianturi Tigor Franky, and R. Mahardiko, “Forecast Analysis of Research Chance on AES Algorithm to Encrypt during Data Transmission on Cloud Computing,” in *2020 2nd International Conference on Broadband Communications, Wireless Sensors and Powering (BCWSP)*, pp. 163–166, Sep 2020.

[9] K. Hariss, H. Noura, and A. E. Samhat, “Fully Enhanced Homomorphic Encryption algorithm of MORE approach for real world applications,” *J. Inf. Secur. Appl.*, vol. 34, pp. 233–242, Jun. 2017.

[10] D. Nuñez, I. Agudo, and J. Lopez, “Proxy Re-Encryption: Analysis of constructions and its application to secure access delegation,” *J. Netw. Comput. Appl.*, vol. 87, pp. 193–209, Jun. 2017.

[11] S. Amamou, Z. Trifa, and M. Khmakhem, “Data protection in cloud computing: A Survey of the State-of-Art,” *Procedia Comput. Sci.*, vol. 159, pp. 155–161, 2019.

[12] W. Al Etaiwi and S. Hraiz, “Structured encryption algorithm for text cryptography,” *J. Discret. Math. Sci. Cryptogr.*, vol. 21, no. 7–8, pp. 1559–1572, Nov. 2018.

[13] B.-H. Lee, E. K. Dewi, and M. F. Wajdi, “Data security in cloud computing using AES under HEROKU cloud,” in *2018 27th Wireless and Optical Communication Conference (WOCC)*, pp. 1–5, Apr. 2018.

[14] Y. Liu, W. Gong, and W. Fan, “Application of AES and RSA Hybrid Algorithm in E-mail,” in *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, pp. 701–703, Jun. 2018.

[15] K. M. Akhil, M. P. Kumar, and B. R. Pushpa, “Enhanced cloud data security using AES algorithm,” in *2017 International Conference on Intelligent Computing and Control (I2C2)*, vol. 2018-Janua, pp. 1–5, Jun. 2017.

[16] M. D. Boomija, “Secure data sharing through additive similarity based ElGamal like encryption,” in *2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, pp. 652–655, Feb. 2016.

[17] Q. Wu, B. Qin, L. Zhang, J. Domingo-Ferrer, O. Farras, and J. A. Manjon, “Contributory

broadcast encryption with efficient encryption and short ciphertexts,” *IEEE Trans. Comput.*, vol. 65, no. 2, pp. 466–479, 2016.

[18] A. S. Babrahem and M. M. Monowar, “Preserving confidentiality and privacy of the patient’s EHR using the OrBAC and AES in cloud environment,” *Int. J. Comput. Appl.*, vol. 7074, pp. 1–12, Aug. 2018.

[19] P. Sivakumar, M. NandhaKumar, R. Jayaraj, and A. S. Kumaran, “Securing Data and Reducing the Time Traffic Using AES Encryption with Dual Cloud,” in *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, pp. 1–5, Mar. 2019.

[20] J. C. S. dos Anjos *et al.*, “Fast-Sec: an approach to secure Big Data processing in the cloud,” *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 34, no. 3, pp. 272–287, May 2019.

[21] Z. Deng, K. Li, K. Li, and J. Zhou, “A multi-user searchable encryption scheme with keyword authorization in a cloud storage,” *Futur. Gener. Comput. Syst.*, vol. 72, pp. 208–218, Jul. 2017.

[22] E. Jintcharadze and M. Iavich, “Hybrid Implementation of Twofish, AES, ElGamal and RSA Cryptosystems,” in *2020 IEEE East-West Design & Test Symposium (EWDTS)*, pp. 1–5, Sep.2020.

[23] D. Rachmawati, M. A. Budiman, and M. I. Wardhono, “Hybrid Cryptosystem for Image Security by Using Hill Cipher 4x4 and ElGamal Elliptic Curve Algorithm,” in *2018 IEEE International Conference on Communication, Networks and Satellite (Comnetsat)*, pp. 49–54, Nov. 2018.

[24] I. Yakymenko, M. Kasianchuk, O. Gomotiuk, G. Tereshchuk, S. Ivasiev, and P. Basistyi, “Elgamal cryptoalgorithm on the basis of the vector-module method of modular exponentiation and multiplication,” in *2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET)*, pp. 926–929, Feb. 2020.

[25] T. R. Poojitha and S. Santhanalakshmi, “Assured privacy and authentication of health data in cloud using cryptographic algorithm,” in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2018 - Proceedings*, pp. 273–280, May. 2018.

[26] Z. Hua and F. Xia, “A new cryptosystem and digital signature scheme based on ElGamal cryptosystem,” in *2011 International Conference on Computer Science and Service System (CSSS) - Proceedings*, pp. 998-1000, 2011.

[27] K. El Makkaoui, A. Beni-Hssane and A. Ezzati, "Cloud-ElGamal: An efficient homomorphic encryption scheme," in *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM) - Proceedings*, pp. 63-66, 2016.

[28] Maxrizal, S. Irawadi and Sujono, "Discrete Logarithmic Improvement for ElGamal Cryptosystem Using Matrix Concepts," in *2020 8th International Conference on Cyber and IT Service Management (CITSM) - Proceedings*, pp. 1-5, 2020.

[29] P. Sharma, S. Sharma and R. S. Dhakar, "Modified Elgamal Cryptosystem Algorithm (MECA)," in *2011 2nd International Conference on Computer and Communication Technology (ICCCT-2011)*, *- Proceedings*, pp. 439-443, 2011.

[30] R. A. Haraty, A. N. El-Kassar and B. M. Shebaro, "A Comparative Study of Elgamal Based Digital Signature Algorithms," in *2006 World Automation Congress*, *- Proceedings*, pp. 1-6, 2006.

[31] R. Balamurugan, V. Kamalakannan, G. D. Rahul and S. Tamilselvan, "Enhancing security in text messages using matrix based mapping and ElGamal method in elliptic curve cryptography," in *2014 International Conference on Contemporary Computing and Informatics (IC3I)*, *- Proceedings*, pp. 103-106, 2014.

[32] S. Singh and P. Ahlawat, "Candidate-resolved online voting protocol using distributed ElGamal Cryptosystem," in *Proceeding of the Cube International Information Technology Conference*, pp. 759-763, 2012.

[33] P. Holster, H. Petersen and M. Michels, "Meta-ElGamal signature schemes," in *Proceeding* of *the 2nd ACM Conference on Computer and communication security*, pp. 96-107, 1994.

[34] A. G. Chefranov and A. Y. Mahmoud, "Elgamal public key cryptosystem and signature scheme in GU (m,p,n)," in *Proceeding* of the 3rd international conference on Security of information and networks, pp. 164-167, 2010.

[35] T. Hidayat, “Encryption Security Sharing Data Cloud Computing by Using AES Algorithm: A Systematic Review,” TEKNOKOM, vol. 2, no. 2, pp. 11–16, Dec. 2019.

[36] L. Sodikin and T. Hidayat, “Analisa Keamanan E-Commerce Menggunakan Metode AES Algoritma,” TEKNOKOM, vol. 3, no. 2, pp. 8–13, Dec. 2020.