Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 28 (2025), No. 4-B, pp. 1317–1329
DOI : 10.47974/JDMSC-2270

A novel cryptosystem using integer power

Fatehiya Abdul Kareem Abed [†]
Zaynab Anwer Ahmed *
Department of Mathematics
College of Science for Women
University of Baghdad
Baghdad
Iraq

Abir Jaafer Hussain [§]
College of Engineering
University of Sharjah
26666
U.A.E.

Abstract

Various methods are utilized providing complexity for cryptosystem with the aim to increase the security and avoiding hacker attack. Hybrid cryptosystem is one of these cryptosystems which is used two types of cryptosystems and has many applications in data transmitted. This research, proposed a novel method that used power exponent instead of using the prime number directly and also providing complexity of asymmetric cryptosystems. This method has been applied theoretically in two public systems RSA and EL-Gamal. Power RSA and Power EL-Gamal are modified asymmetric cryptosystems, in which the power number is kept by the sender and the receiver. Moreover, we use group theory to prove that these cryptosystems work properly. Our extensive studies indicated that our proposed encrypt and decrypt method provides increased security against hackers.

Subject Classification: *68M25, 11F37.*

Keywords: *Cryptosystem, Encryption, Decryption, RSA algorithm, EL-Gamal algorithm.*

[†] *E-mail:* **fathiya.abd2203m@csw.uobaghdad.edu.iq**
* *E-mail:* **zainabaa_math@csw.uobaghdad.edu.iq (Corresponding Author)**
[§] *E-mail:* **abir.hussain@sharjah.ac.ae**

© Taru Publications

1. Introduction

Encryption is the process of making messages unreadable to the general public. It is an art and science dedicated to protecting information [1]. Cryptography is a science based on mathematics [2] in which most encryption algorithms depend on mathematical algorithms [3].

There are three different types of cryptography including symmetric, asymmetric and hybrid. In symmetric cryptography the sender and the receiver utilise the same key for encryption and decryption [4], [5] while for asymmetric the receiver generates the public and private keys. The sender encrypts using public key while the receiver decrypts utilising private key [6] In hybrid cryptography symmetric and asymmetric cryptosystems are combined to provide high security and more complexity [7].

The most famous algorithms for asymmetric cryptosystems are RSA and EL-Gamal algorithms, [3]. These two algorithms were designed according to three basic stages. The first stage is to generate public and private keys, which are used to encrypt and decrypt the information [8]. The second stage is the encryption process, which is the process of converting plaintext into encrypted text, and the third stage is the decryption process, which is the opposite of the encryption, in which the encrypted text is returned to plain text [1].

The idea of RSA security depends on decomposing an integer into its prime factors, while EL-Gamal security depends on separating logarithm problem for message cryptosystem [9], [1].

This paper proposed two novel cryptosystems, the first said Power RSA and the second said Power EL-Gamal. Both used power exponent based on prime numbers in encryption and decryption for leading further security.

The novelty of our proposed cryptosystems are as follows:

(1) A new Power RSA algorithm is proposed which is able to increase the complexity of encryption and decryption. This will decrease the chance of hackers in finding the private key due to the power RSA do not use the prime numbers directly.

(2) Increasing the power to N improves the selection of p and q values with keeping the requirements of Euler's theorem. It will be enough to select p and q greater than the integers that correspond to the alphabet letters.

(3) This paper used group theory to prove the existence of a unique multiplicative inverse of $e\ modulo\ \varphi(N^k)$ that need to satisfy the condition of Euler's theorem.

(4) A new Power EL-Gamal algorithm is proposed to increase the cryptosystem's complexity by Discrete Logarithms Problem (DLP) and power modulo. Increasing power of the prime number of the public key leads to using it indirectly and provides more security against hackers.

The remainder of this research is organized as follows. Section 2 presents basic concepts and related works, while the proposed cryptosystems Power RSA

and Power EL-Gamal are illustrated in Sections 3 and 4 respectively. Finally, Section 5 provides details about conclusions and future works.

2. Literature Review and Related Works

The most famous algorithms for asymmetric cryptosystems are RSA and EL-Gamal. Various work has been developed them. such as in 2015, Swami, Singh, and Choudhary [10] used RSA algorithm to increase security, by a method called Dual RSA that uses double encryption and decryption. In 2017, Manu and Aarti [4] proposed a method that used four keys for the RSA algorithm, two keys for encryption, and two keys for decryption. In 2019, Intila, Gerardo, and Medina [11] presented a new modified version of RSA where modifying the N and the public key e is created. In 2024, Suhael, Ahmed, and Hussain [7] developed the RSA algorithm, which is called the square of RSA technique.

As in RSA cryptosystem, some researchers tried to improve EL-Gamal cryptosystem. In 2018, Ordonez, Medina, and Gerardo [12] modified the EL-Gamal system, which allows a message to be encrypted by multiple senders and sent to a single recipient for decryption, using several different private keys, which are also used for decryption. Because of these keys, the encrypted message is highly secure. In 2018, Dissanayake [13] improved the EL-Gamal encryption system by changing the structure of the encrypted message sent. This method analyzes the corresponding number of the plaintext into its prime factors and then the number of prime factors (may be repeated) is used as a power of the plaintext. In 2020, Ranasinghe and Athukorala [14] introduced a generalization of the improved EL-Gamal system by Dissanayake. In 2021, Al-Naamee and Ali [15] used a 3D chaotic system to generate prime numbers as public and private keys.

In 2021 Emmanuel et al. showed in [5] that the time and space complexity of RSA cryptosystem are better during the encryption process while during the decryption process El-Gamal performs better in time complexity

Quick development in Internet applications needs various types of cryptosystems to exchange information in a secure channel. Therefore, the secure channel requires continuous improvement to guarantee the strength of the cryptosystem. This research proposed two cryptosystems of the public key, which are Power RSA and Power El-Gamal. Our proposed novel methods provide a high level of security and complexity for the cryptosystem. Hence users will utilize them to transmit information as detailed in Sections 3 and 4.

3. Power RSA

With the advancement of technology, various researchers have been trying to improve RSA to improve security against hackers. In this work, the Power RSA cryptosystem was proposed to use the public key indirectly by expanding it to the power. This power value remains secret between the sender and the receiver to provide high security against hackers. The proposed system as in RSA has three stages, public and private key generation, encryption, and

3.1 *The Keys generation of our proposed Power RSA*

Step 1: Select the power positive integer k, by the sender and the receiver together in a private channel.

Step 2: Two different prime numbers p and q values are selected by the receiver, then let $N = p \times q$. Then calculate $N^k = p^k * q^k$ where k is the positive integer. Large values for p and q will provide greater security since it is difficult to factorise N, however, this will require complex calculations. Therefore, using power to N in Power RSA gives improvement in selecting p, q values. It should be mentioned that to satisfy the requirement of Euler's theorem, p and q must be chosen to be greater than the integers that represent the alphabet letters. (that is the numbers corresponding to the plaintext will be relatively prime to p and q).

For the next step, we need the following remark and proposition:

Remark 1: $\forall N$, an integer number let $Z^*_{\varphi(N)} = \{a < \varphi(N) : GCD(a, \varphi_{(N)}) = 1\}$, hence $Z^*_{\varphi(N^k)} = \{b < \varphi(N^k): GCD\,(b, \varphi_{(N^k)}) = 1$, where k is the positive integer. Since the set Z^*_N is a finite commutative group under multiplication (mod N)[16] , then $Z^*_{\varphi(N)}$ as well as $Z^*_{\varphi(N^k)}$ are groups.

Proposition 1: Let p and q be prime numbers, $N = pq$, $e \in Z^*_{\phi(N)}$ and $p \nmid e, q \nmid e$ then $e \in Z^*_{\phi(N^k)}$where k is any positive integer.

Proof: Assume GCD (e,$\varphi(N^k)$) =s. That is $s|\varphi(N^k)$ and $s|e$

By the fundamental theorem of arithmetic, there exists a prime number t such that $t \mid s$, then $t \mid e$ and $t \mid \varphi(N^k)$.

Since $N^k = p^k q^k$.

Then $\varphi(N^k) = (p^k - p^{k-1})(q^k - q^{k-1}) = p^{k-1}q^{k-1}(p-1)(q-1)$

Since $e \in Z^*_{\varphi(N)}$ then $t \nmid (p-1)(q-1)$and hence $t \mid p^{k-1}q^{k-1}$ implies either $t \mid p^{k-1}$ or $t \mid q^{k-1}$.

Then either $t = p$ or $t = q$ that is either $p \mid e$ or $q \mid e$ a contradiction. □

Step 3: Select integer e>1 such that $e \in Z^*_{\varphi(N)}$ and $p \nmid e, q \nmid e$ then by proposition (1) *GCD* (e,$\varphi(N^k)$) =1, where k is any positive integer. Hence (e, N) is a public key.

Step 4: Since $Z^*_{\varphi(N)}$ is a group then there exists a unique multiplicative inverse of e modulo $\varphi(N^k)$ let called $it\ d \in Z^*_{\varphi(N)}$. Then (d, N^k) is the private key.

Step 5: The receiver sends the public key to the general public.

Algorithm 1 shows the construction of generated keys of the proposed power RSA at the receiver.

Algorithm 1: Keys generation of the proposed Power RSA at the receiver

Input

- k the positive integer
- p and q prime numbers

Key generated

- Compute the N,N^k such that N=p*q and $N^k = p^k q^k$
- Calculate $\varphi(\mathrm{N}) = (\mathrm{p}-1)(\mathrm{q}-1)$ and
- $\varphi(\mathrm{N}^{\mathrm{k}}) = (\mathrm{p}^{\mathrm{k}} - \mathrm{p}^{\mathrm{k}-1})(\mathrm{q}^{\mathrm{k}} - \mathrm{q}^{\mathrm{k}-1})$
- Select public key e where $1 < e < \varphi(N)$, $p \nmid e, q \nmid e$ and GCD(e, $\varphi(N)$)=1$then$ GCD(e, $\varphi(N^k)$)=1
- Compute the private key d where $ed \equiv 1\ mod\varphi(N^k)$

Output

- The public key is (e, N) ,The private key is (d, N^k)

3.2 *Encryption and Decryption of the Power RSA Cryptosystem*

Let us consider the following scenario, the sender Bob utilizes the key (e, N) that received from Alice in the Power RSA algorithm. Bob needs to encrypt the plaintext M by the public key to get the ciphertext C as follows: $C \equiv m^e (mod\ N^k)$ where m is the corresponding number to massage M, and k the selected power

Alice received the ciphertext C from Bob and decrypts it by using her private key (d, N^k) to get plaintext M as follows: $m \equiv C^d (mod N^k)$

Alice transfers m to the corresponding message M by the reversible mapping function.

Algorithms 2 and 3 show the encryption and decryption of the proposed Power RSA at the sender and the receiver, respectively.

Algorithm 2: Encryption by Using Power RSA at the Sender

Input

- (e,N)the public key
- k is a positive integer.
- M the plaintext

Power RSA Encryption

- Convert M to the corresponding numbers m
- Compute $\mathrm{C} \equiv \mathrm{m}^{\mathrm{e}}(\ \mathrm{mod}\ \mathrm{N}^{\mathrm{k}})$

Output

- Ciphertext C

Algorithm 3: Decryption by using Power RSA at the Receiver

Input

- k the positive integer
- $(\mathrm{d}, \mathrm{N}^{\mathrm{k}})$ the privet key
- C the ciphertext

Power RSA Decryption

- Compute $\mathrm{m} \equiv \mathrm{c}^{\mathrm{d}} \ (\mathrm{mod}\ \mathrm{N}^{\mathrm{k}})$
- Convert m to corresponding letters M

Output

- Plaintext M

Example 1: Let $p = 31$,$q = 37, k = 3$, $M = FUNCTION$.

The public and private keys generated at the receiver

- $\mathrm{N} = \mathrm{pq} = 1147.$
- $\varphi(\mathrm{N}) = (\mathrm{P}-1)(\mathrm{q}-1) = (30)(36) = 1080$
- $\mathrm{N}^{\mathrm{k}} = \mathrm{N}^3 = \mathrm{p}^3\mathrm{q}^3 = (31)^3(37)^3 = 1509003523$
- $\varphi(\mathrm{N}^3) = (\mathrm{p}^3 - \mathrm{p}^2)(\mathrm{q}^3 - \mathrm{q}^2) = (28830)(49284) = 1420857720$
- Choose e such that $\mathrm{e} \in \mathrm{Z}^*_{\varphi(\mathrm{N})}$ then $\mathrm{e} \in \mathrm{Z}^*_{\varphi(\mathrm{N}^{\mathrm{k}})}$ let $\mathrm{e} = 11$
- $(\mathrm{e}, \mathrm{N}) = (11{,}1147)$ the public key
- Compute d as the inverse multiplication of $\mathrm{e} \bmod (\varphi(\mathrm{N}^3))$, that is $\mathrm{ed} \equiv 1 \bmod \varphi(\mathrm{N}^3)$. $\mathrm{d} = 387506651$
- The private key$(\mathrm{d}, \mathrm{N}^3) = (387506651{,}1509003523)$

Remark 2: Due to the arithmetic complexity, the American Standard Code for Information Interchange (ASCII) table was not utilized in this work. It was replaced by converting 26 capital letters to integers corresponding to values ranging between 2 and 27.

Encryption: Compute the ciphertext C by utilizing in Power RSA data encryption

- M =FUNCTION equivalent to m =7 22 15 4 21 10 16 15
- $C \equiv m^e(\ mod\ N^k)$
- $C \equiv (7\ 22\ 15\ 4\ 21\ 10\ 16\ 15)^{11}(mod\ 1509003523)$
- C =468323220 448231745 147665539 4194304 57765846 405767482 222973282 147665539

Decryption: Get ciphertext C by using Power RSA decryption

- $m \equiv C^d\ (mod\ N^k)$
- $m \equiv (468323220\ 448231745\ 147665539\ 4194304\ 57765846\ 405767482\ 22297328\ 2\ 147665539)^{387506651}(mod 1509003523)$

- $m = 7\ 22\ 15\ 4\ 21\ 10\ 16\ 15$
- M =FUNCTION

4. Power EL-Gamal:

The EL-Gamal cryptosystem is a popular public key cryptosystem based on the computational challenge of calculating discrete logarithms in finite fields. EL-Gamal cryptography using the sender's private key l more than once and knowing one of the messages m_j by the attack leads easily to knowing all the rest of the messages m_i. If m_1 is known, m_2 can be calculated using the following formula:

Since $c_{2,1} \equiv m_1(g^b)^l(mod\ p)$ and $c_{2,2} \equiv m_2(g^b)^l(mod\ p)$.

Then $m_2 \equiv\ m_1 \frac{c_{2,2}}{c_{2,1}}(mod\ p)$

Therefore, for high security against attack, a new random private key l is recommended in each single encrypted message m_i,[17]. This work proposes a modified EL-Gamal called Power EL-Gamal gives more security than EL-Gamal since knowing one message m_j and the public key (q, α, h) will not give m_i when the power k is unknown. This method is based on the DLP and the power of the modulo where the power is chosen by the sender and receiver. Using power will increase the complexity of the mathematical calculations. Similar to the original EL-Gamal algorithm, our proposed method consists of three parts: the first is keys generation, the second is encryption, and the third is decryption.

4.1 *Keys generation of Power EL-Gamal is illustrated in Algorithm 4*

- Choose the power which is the positive integer k by the sender and the receiver together in a private channel.
- Alice generates pairs of keys as follows: she chooses a large prime number q that gives a guarantee of relatively prime to the corresponding numbers of the plaintext.
- Alice chooses a generator α of multiplicative group Z_q^* , then $GCD(\alpha, q^k) = 1$ and hence $\alpha \in Z_{q^k}^*$
- Also, chooses the integer number d such that $1 < d < \varphi(q^k)$.
- Let $h \equiv \alpha^d(\ mod q^k)$
- The public key will be the triple values (q, α, h) that Bob will receive.
- The private key will be only (d) which must be kept secret by Alice.

Algorithm 4: Generation of Keys for the Power EL-Gamal at the Receiver

Input

- q a large prime number.
- α a generator of multiplicative group Z_q^*.
- k the positive integer.

Key generated

- Compute $\varphi(q^k) = q^k - q^{k-1}$
- Choose d such that $1 < d < \varphi(q^k)$
- Compute $h \equiv \alpha^d (mod q^k)$

Output

- (q, α, h) as the public key
- (d) as the private key of the reserve

4.2 *The process of encryption by the Power EL-Gamal*

Bob encrypts the plaintext M by her public key (q, α, h) as follows:

- Map the message M to the corresponding numbers m by using a mapping function.
- Bob chooses a random integer t such that $1 < t < q^k - q^{k-1}$ (which is kept secret).
- Computes $C_1 \equiv \alpha^t (mod\ q^k)$ and $S = h^t\ (mod\ q^k)$.

  $C_2 \equiv m * S\ (mod\ q^k)$.
- Bob sends the ciphertext (C_1, C_2) to Alice.

4.3 *The process of decryption of the Power EL-Gamal algorithm*

Alice decrypts the ciphertext (C_1, C_2) using her private key d to get the decryption m.

The following proof will show how to get the decryption m from $C_2 * {C_1}^{-d}$.

Proof:

$$
\begin{aligned}
C_2 * {C_1}^{-d} &\equiv (m * S){C_1}^{-d} (mod q^k) \equiv m(h^t)(C_1)^{\phi(q^k)-d} (mod q^k) \\
&\equiv m(h^t)(C_1)^{q^k - q^{k-1} - d} (mod q^k) \equiv m(\alpha^d)^t (\alpha^t)^{q^k - q^{k-1} - d} (mod q^k) \\
&\equiv m\ (\alpha)^{td} (\alpha^{q^k - q^{k-1}})^t (\alpha)^{-td} (mod\ q^k) \\
&\equiv m(\ \alpha)^{td} (1)^t (\alpha)^{-td} (mod\ q^k) \\
&\qquad \text{[by Euler's theorem } \alpha^{q^k - q^{k-1}} = \alpha^{\varphi(q^k)} \equiv 1\ mod\ q^k \quad \equiv m]
\end{aligned}
$$

Alice transfers m to the corresponding message M by the reversible mapping function.

Algorithms 5 and 6 shows the encryption and decryption of the proposed Power El-Gamal at the sender and the receiver, respectively.

Example 2 :

- Let q =29 , k =3 , $M = FUNCTION$

The public and private keys generated at the receiver

- q^3=24389 ,$\varphi(q^3) = \mathrm{q}^3 - \mathrm{q}^2$ =23548
- Choose $\alpha \in Z_q^*$ then $\alpha \in Z_{q^k}^*$, let α =2
- Choose d such that $1 < d < \varphi(q^3)$, let d =5

- Calculate $h \equiv \alpha^d (mod\ q^3) \equiv 2^5 (mod 24389) = 32$
- The public key will be the triple (13,2,32)
- The private key will be only (5)

Encryption: Find the ciphertext (C_1, C_2) by using the Power EL-Gamal encryption

- M =FUNCTION equivalent to m =7 22 15 4 21 10 16 15
- Choose t such that $1 < t < \varphi(q^3)$, let $t = 7$
- Calculate $C_1 \equiv \alpha^t (mod\ q^k) \equiv 2^7 (mod\ 24389) \equiv 128$
  $C_2 \equiv m * h^t\ (mod\ q^k). \equiv (7\ 22\ 15\ 4\ 21\ 10\ 16\ 15)(32)^7 (mod\ 24389)$
  $C_2 \equiv$(20993 17200 20596 11996 14201 5601 23595 20596)
- Then (C_1, C_2) =[128,(20993 17200 20596 11996 14201 5601 23595 20596))]

Decryption

- Using the ciphertext (C_1, C_2) that gets from Power EL-Gamal *en*cryption
- Calculate $Z \equiv C_1^d\ (mod q^k) \equiv 128^5 (mod\ 24389) = 2999$
- $m \equiv C_2 Z^{q^k - q^{k-1}} (\ mod\ 24389) \equiv$(20993 17200 20596 11996 14201 5601 23595 20596)$(2999)^{23547} (mod\ 24389) = (7\ 22\ 15\ 4\ 21\ 10\ 16\ 15)$
  M =FUNCTION

Algorithm 5: Encryption by Using Power EL-Gamal at the Sender

Input

- The public key (q,α,h)
- k the positive integer
- t sender's private key such that $1 < t < q^k - q^{k-1}$
- The plaintext M

Power EL-Gamal Encryption

- Converte M to the corresponding numbers m
- Compute $c_1 \equiv \alpha^t (\text{ mod } q^k)$ and $s \equiv h^t (mod\ q^k)$
- $c_2 \equiv m^* s (\text{ mod } q^k)$

Output

- Ciphertext (c_1 ,c_2)

Algorithm 6: Decryption by Using Power EL-Gamal at the Receiver

Input

- The public key (q,α,h)
- d receiver's private key
- k the positive integer
- The ciphertext (c1 , c2)

Power EL-Gamal Decryption

- Compute $m \equiv c_2 * c_1^{-d} (\bmod q^k)$.
- Converte m to corresponding letters M

Output

- The plaintext M

5. Simulation Results

In this section, simulations were conducted to test the performance of the two proposed algorithms and to compare each proposed algorithm with the standard algorithm using Python code. The time complexity to encrypt, and decrypt is given in seconds (s). The space used for the keys and the cipher text. is given in bits. The code was executed on a computer with the following specifications: Processor: Intel(R) Core(TM) i7-8850H CPU @ 2.60GHz 5.59GHz, RAM: 32.0 GB. Also, it should be noted that only 26 uppercase letters were used for cryptosystem which correspond to integers ranging from 2-27.

Table 1

Time Complexity of Encryption

Characters of FUNCTION	Time RSA Encryption	Time Power RSA Encryption	Time ElGamal Encryption	Time Power EL-Gamal Encryption
total	0.06039988	0.06689993	0.04749966	0.88750012

Table 2

Time Complexity of Decryption

Characters of FUNCTION	Time RSA Decryption	Time Power RSA Decryption	Time ElGamal Decryption	Time Power EL-Gamal Decryption
Total	0.13489975	0.89399971	0.03599993	0.72910008

Table 3

Space Complexity

ASPECT	RSA	POWER RSA	EL-GAMAL	POWER EL-GAMAL
Public Key Size	15	15	9	13
Privet Key Size	20	60	3	3
Cipher text size	77	218	35	114

It is noted from Table 1 that the time complexity of encryption by Power RSA algorithm is slightly similar to that of RSA. While we note that the time complexity to encrypt the Power EL-Gamal is higher than what it is in EL-Gamal by 84% and Power RSA by 82%. Table 2 shows that the time complexity to decrypt by Power RSA algorithm is higher than that of RSA by 76% and

power EL-Gamal 17%. Also, the value of the time complexity to decrypt the Power EL-Gamal is higher than that of EL-Gamal algorithm by up to 69%.

Table 3 shows that the space complexity of the ciphertext by Power RSA and Power EL-Gamal is greater than that of RSA and EL-Gamal respectively. Moreover, the space complexity of the privet key and ciphertext in Power RSA is higher than that in others.

6. Conclusion and Future Works:

Encryption plays a vital role in modern digital technology for secure data storage and communication. In this paper, two improvements to the well-known encryption algorithms RSA and EL-Gamal are proposed to increase the complexity and improve the security of these algorithms. Both improvements depend on the use of a power exponent based on prime numbers, hence the name Power RSA and Power EL-Gamal. In both cryptosystems the power is exchanged between the sender and the receiver in a secured channel. This modification leads to increased complexity and gives more security against hackers. Increasing the power to the public key N in Power RSA improves the selection of the prime numbers p and q. It is sufficient to choose p and q larger than the integers that correspond to the letters of the alphabet to keep satisfying the requirements of Euler's theorem.

Power EL-Gamal gives more security than EL-Gamal since knowing one message and the public key will not give the original message when the power of the prime number in the public key is unknown. This method is based on the DLP and the power of the modulo where the power is chosen by the sender and receiver. Power EL-Gamal will decrease the chance of the hacker getting the original message.

Future work involving the use of Power El-Gamal and Power RSA with symmetric cryptosystems such as (AES, DES) can form hybrid cryptosystems that are more secure, less time consuming and more complex.

References

[1] Z. Liu, "Evaluating RSA Encryption: Primality Testing, Pollards Algorithms, and Security challenges," *Theor. Nat. Sci.*, vol. 13, no. 1, pp. 287–292 (2023), doi: 10.54254/2753-8818/13/20240864.

[2] D. Kahrobaei, R. Flores, and M. Noce, "Group-based Cryptography in the Quantum Era," *Not. Am. Math. Soc.*, vol. 70, no. 05, pp. 1 (2023), doi: 10.1090/noti2684.

[3] P. K. Arhin Jnr, M. Asante, and L. Otoo, "A Comparative Study of RSA and EL-Gamal Cryptosystems," *Int. J. Comput. Eng.*, vol. 4, no. 1, pp. 33–41 (2023), doi: 10.47941/ijce.1291.

[4] Manu and A. Goel, "Encryption Algorithm Using Dual Modulus," in 3rd IEEE International Conference on "Computational Intelligence and Communication Technology" (IEEE-CICT 2017), pp. 1–4 (2017). doi: 10.1109/CIACT.2017.7977331.

[5] A. A. Emmanuel, O. A. Eniola, A. M. Okunade, and A. E. Olanrewaju, "A Note on Time and Space Complexity of RSA and ElGamal Cryptographic Algorithms," *Int. J. Adv. Comput. Sci. Appl.,* vol. 12, no. 7, pp. 143–147 (2021), doi: 10.14569/IJACSA.2021.0120716.

[6] A. Shetty, K. Shravya, and K. Krithika, "A Review on Asymmetric Cryptography – RSA and ElGamal Algorithm," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 2, no. 5, pp. 98–105 (2014).

[7] S. M. Suhael, Z. A. Ahmed, and A. J. Hussain, "Proposed Hybrid Cryptosystems Based on Modifications of Playfair Cipher and RSA Cryptosystem," *Baghdad Sci. J.,* vol. 21, no. 1, pp. 151–160 (2024), doi: 10.21123/bsj.2023.8361.

[8] G. H. J. Lanel, T. M. K. K. Jinasena, and B. A. K. Welihinda, "A Survey of Public-Key Cryptography over Non-Abelian Groups," *Int. J. Comput. Sci. Netw. Secur.,* vol. 21, no. 4, pp. 289–300 (2021), doi: /10.22937/IJCSNS.2021.21.4.36.

[9] A. Thakkar and R. Gor, "Cryptographic Method to Enhance Data Security Using RSA Algorithm and Mellin Transform," *Int. J. Eng. Sci. Technol.,* vol. 7, no. 2, pp. 63–72 (2023), doi: 10.29121/ijoest.v7.i2.2023.490.

[10] B. Swami, R. Singh, and S. Choudhary, "Dual Modulus RSA Based on Jordan-Totient Function," *Procedia Technol.,* vol. 24, pp. 1581–1586 (2016), doi: 10.1016/j.protcy.2016.05.143.

[11] C. Intila, B. Gerardo, and R. Medina, "A Study of Public key 'e' in RSA Algorithm," in IOP Conference Series: Materials Science and Engineering, IOP Publishing, pp. 012016 (2019). doi: 10.1088/1757-899X/482/1/012016.

[12] A. J. Ordonez, R. P. Medina, and B. D. Gerardo, "Modified El Gamal algorithm for multiple senders and single receiver encryption," ISCAIE 2018 - 2018 IEEE Symp. Comput. Appl. Ind. Electron., no. October 2019, pp. 201–205 (2018), doi: 10.1109/ISCAIE.2018.8405470.

[13] W. D. M. G. M. Dissanayake, "An Improvement of The Basic El-Gamal Public Key Cryptosystem," *Int. J. Comput. Appl. Technol. Res.,* vol. 7, no. 2, pp. 40–44 (2018).

[14] R. Ranasinghe and P. Athukorala, "A generalization ofThe ElGamal public-key cryptosystem," *J. Discret. Math. Sci. Cryptogr.,* vol. 25, no. 8, pp. 2395–2403 (2022), doi: 10.1080/09720529.2020.1857902.

[15] M. K. Al-Naamee and S. M. Ali, "Improved El Gamal Public Key Cryptosystem Using 3D Chaotic Maps," *Bull. Electr. Eng. Informatics,* vol. 10, no. 1, pp. 404–411 (2021), doi: 10.11591/eei.v10i1.2124.

[16] A. Kumar, Basic Group Theory. BFC Publications, India (2023).

[17] M. El Laz, B. Gregoire, and T. Razk, "Security Analysis of ElGamal Implementations," in Secrypt 2020 - 17th International Conference on Security and Cryptography., Paris, France, pp. 310–321 (2020). doi: 10.5220/0009817103100321.

Received, February, 2025