Ciphertext-Independent and Unbounded Depth RSA-Based Updatable Encryption Scheme

Mostefa Kara
Interdisciplinary Research Center for Intelligent Secure Systems (IRC-ISS), King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, 31261,
Saudi Arabia
mostefa.kara@kfupm.edu.sa

Abdelkader Laouid
LIAP Laboratory, El Oued University, PO Box 789, El Oued 39000,
El Oued, Algeria
abdelkader-laouid@univ-eloued.dz

Ahcène Bounceur
College of Computing and Informatics, University of Sharjah, Sharjah,
UAE
abounceur@sharjah.ac.ae

Mohammad Hammoudeh
Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, M15 6BH,
UK
m.hammoudeh@mmu.ac.uk

Abid Mohamed Nadhir
LIAP Laboratory, El Oued University, PO Box 789, El Oued 39000,
El Oued, Algeria
abid-mohamednadhir@univ-eloued.dz

Kahla Mohamed El Habib
LIAP Laboratory, El Oued University, PO Box 789, El Oued 39000,
El Oued, Algeria
kahla-mohammedelhabib@univ-eloued.dz

***Abstract*—We introduce a novel updatable encryption scheme based on the RSA cryptosystem to address the challenges of key rotation in secure communication systems. Updatable encryption allows ciphertexts encrypted under an old key to be transformed into ciphertexts under a new key without decrypting the plaintext, thereby enhancing security without compromising data integrity or confidentiality. Our RSA-based scheme leverages the mathematical properties of RSA to enable efficient and seamless key updates. We provide a security analysis demonstrating that our scheme is not vulnerable to information leaking. Experimental results indicate that our method offers a practical and efficient solution for large-scale systems that require regular key updates, with minimal impact on computational resources and performance.**

***Index Terms*—Encryption key rotation, Untrusted cloud Re-encryption.**

I. Introduction

In today's digital landscape, the rapid exchange of sensitive information over networks underscores the critical need for robust encryption mechanisms to protect data from unauthorized access and tampering. Cryptographic schemes serve as the cornerstone of secure communication, ensuring data confidentiality and integrity while enabling trust in digital interactions [1], [2]. Among these, the RSA cryptosystem has emerged as a longstanding pillar of cybersecurity, celebrated for its mathematical rigor and extensive application in securing digital communications [3]. Its combination of simplicity and effectiveness has made RSA a foundational tool in protecting sensitive information across diverse domains.

Despite RSA's resilience and reliability, evolving computational capabilities and emerging cyber threats necessitate adaptive security measures [4], [5]. One key challenge lies in the management and rotation of cryptographic keys, a critical process for maintaining the security of long-term data storage and ongoing transmissions. Regular key updates are essential to mitigate the risks of key compromise or cryptographic aging. However, traditional key rotation methods typically involve decrypting data encrypted under an old key and re-encrypting it under a new one. This approach is not only computationally expensive but also introduces vulnerabilities during the key transition process, as sensitive data is exposed in plaintext form.

Updatable encryption (UE) schemes address these limitations by enabling the transformation of ciphertexts encrypted under an old key to ciphertexts under a new key without ever exposing the plaintext [6]. This capability ensures that data confidentiality is maintained throughout the key rotation process, making UE an attractive solution for dynamic security needs [7]. While various UE schemes have been developed, many rely on symmetric key cryptography or involve intricate constructions, potentially limiting their practicality in certain use cases. Consequently, ongoing research is essential to refine these schemes, ensuring they meet the growing demands for security, efficiency, and scalability in today's interconnected world.

A. Contribution

In this paper, we introduce a novel updatable encryption scheme based on the RSA cryptosystem. Our approach leverages the well-established properties of RSA to facilitate secure and efficient key updates in a public-key setting. By enabling ciphertext transformation without decryption, the scheme ensures both forward and backward security—protecting past and future messages even if a key is compromised.

We begin by revisiting the fundamentals of RSA encryption and the concept of updatable encryption. We then detail the design of our RSA-based updatable encryption scheme, including the algorithms for key generation, encryption, key update, and decryption. A thorough security analysis is provided to demonstrate resistance against adaptive chosen-ciphertext attacks and to verify that the scheme meets essential security properties.

To assess the practicality of our proposed scheme, we conduct extensive experiments evaluating its performance in comparison to traditional key rotation methods. The results indicate that our scheme significantly reduces the computational overhead associated with key updates, making it suitable for large-scale systems and applications where efficiency is critical.

Our contribution not only offers a viable solution for secure key management in RSA-based systems but also expands the applicability of updatable encryption schemes in real-world scenarios. By combining the robustness of RSA with the flexibility of updatable encryption, we aim to enhance the security infrastructure of organizations dealing with sensitive data and frequent key rotations.

We conclude with a discussion of potential applications, limitations, and directions for future research, emphasizing the scheme's potential to become an integral part of modern cryptographic practices.

II. RSA and Updatable encryption

We show the two fundamental concepts we use.

A. I. RSA

RSA (Rivest–Shamir–Adleman) is a widely used public-key cryptographic system that secures digital communication. It relies on the mathematical difficulty of factoring large composite numbers, making it a cornerstone of modern encryption. RSA is used in secure data transmission, digital signatures, and authentication protocols. Its security depends on key size, with larger keys offering stronger protection but requiring more computational resources. Therefore, it is based on the mathematical properties of large prime numbers and modular arithmetic. The RSA algorithm involves key generation, encryption, and decryption processes.

1. Key Generation

Choose Two Large Prime Numbers p and q, these primes should be randomly chosen and kept secret. Then, compute the Modulus n as shown in Equation 1. The modulus n is used in both the public and private keys.

$$n = p \times q \tag{1}$$

The function computes Euler's Totient Function $\phi(n)$ (Equation 2).

$$\phi(n) = (p-1) \times (q-1) \tag{2}$$

This value represents the number of integers less than n that are coprime to n. Then, it chooses the Public Exponent e where $1 < e < \phi(n)$; e must be coprime with $\phi(n)$, meaning $gcd(e, \phi(n)) = 1$. Finally, it computes the Private Exponent d, d is the modular multiplicative inverse of e modulo $\phi(n)$, this means $e \times d \equiv 1 \mod \phi(n)$. Public and Private Keys are respectively (n, e) and (d).

2. Encryption

To encrypt a plaintext message m where $0 \leq m < n$, we use Equation 3.

$$c = m^e \mod n \tag{3}$$

3. Decryption

To decrypt a ciphertext c, we use Equation 4.

$$m = c^d \mod n \tag{4}$$

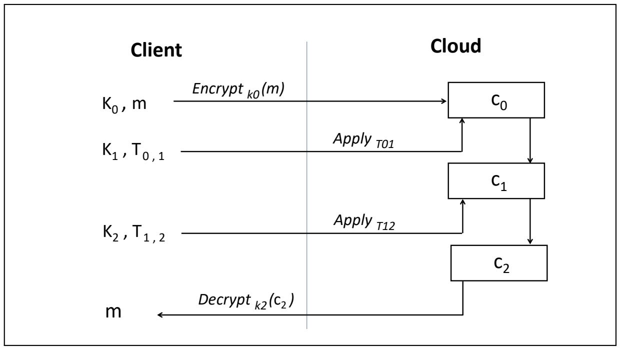

Fig. 1. Updatable Encryption illustration

B. II. Updatable Encryption

UE is a cryptographic scheme that enables ciphertexts encrypted under an old encryption key to be transformed into ciphertexts under a new encryption key without decrypting the underlying plaintext. This transformation is performed using an update token, which is generated based on the old and new keys. The key features of updatable encryption include:

- Key Rotation Without Decryption: Allows for secure key updates without the need to decrypt and re-encrypt data, preserving data confidentiality throughout the process.
- Forward and Backward Security: Ensures that even if an old or new key is compromised, past and future ciphertexts remain secure. Attackers cannot access plaintexts from other periods.
- Security Against Adversaries: The update process does not expose plaintexts or encryption keys, and the update token itself does not reveal any sensitive information.
- Efficiency: Designed to handle large volumes of data efficiently, making it practical for systems that require regular key rotations, such as cloud storage services and secure communication protocols.

To work, UE has four main steps. Firstly, Initial Encryption: data (plaintext) is encrypted using an initial key K_1 to produce ciphertext C_1. Second, Key Update Preparation: when a key update is required, a new key K_2 is generated. An update token T is created using K_1 and K_2. Then, Ciphertext Update phase: the update token T is applied to the original ciphertext

C_1 to produce an updated ciphertext C_2 that is now associated with K_2. The last step is Continued Decryption: the updated ciphertext C_2 can be decrypted using the new key K_2 to retrieve the original plaintext.

UE is applicable in many aspects including Cloud Storage Services i.e., securely managing encrypted data when updating encryption keys; Secure Communication Protocols that ensure the long-term security of messages in systems where keys are periodically updated; Data Compliance i.e., meeting regulatory requirements that mandate regular key changes without compromising data accessibility; Authentication and data integrity [8], [9]; Signature schemes [10]; and IoT application [11]. Updatable encryption provides a practical and secure method for key management in cryptographic systems. Allowing ciphertexts to be updated to new keys without decryption enhances security protocols and reduces the risks associated with key exposure during transitions.

Key Features Illustrated are: 1. Seamless Key Rotation: The figure (Fig. 1) shows how ciphertexts can be updated to a new key without decrypting them. 2. Security Preservation: At no point is the plaintext exposed during the key update process, maintaining data confidentiality. And 3. Forward and Backward Security: Even if an old key K_1 or a new key K_2 is compromised, the security of past and future ciphertexts remains intact.

The Update Token T is crucial for the updatable encryption scheme. It ensures that only authorized parties can perform the ciphertext update. This process is efficient and scalable, making it suitable for systems that handle large volumes of data requiring regular key updates.

III. Related work

Updatable Encryption (UE) is a cryptographic method that allows data encrypted under one key to be securely transformed to another key without decrypting the data. It is particularly valuable in scenarios like key rotation, where periodic key updates are necessary for security compliance or mitigating potential key compromises. The security concepts for Updatable Encryption have significantly progressed since their initial introduction in [12]. The paper [13] presented two security concepts under the chosen plaintext attack (CPA) model, allowing an adversary to distort keys and tokens adaptively. The presented IND-ENC concept requires that fresh encapsulation remain indistinguishable, while IND-UPD applies this indistinguishability requirement to updated ciphertexts. Building on this, Kloob et al. [14] extended these notions to include chosen ciphertext attack (CCA) and ensure integrity protection. Later, Ref. [15] introduced a stronger security concept called IND-UE, that demands that fresh encapsulation be indistinguishable from updated encrypted texts.

In the context of classical cryptography, various constructions for Updatable Encryption (UE) have been proposed. The RISE scheme introduced in [13] represents an updatable adaptation of the ElGamal encryption system, incorporating the public key into its update token. In [14], the authors proposed two generic UE frameworks: one following the encrypt-and-MAC paradigm, secured under the Decisional Diffie-Hellman (DDH) assumption, and another based on the Naor-Yung transformation, secured under the Symmetric External Diffie-Hellman (SXDH) assumption. Additionally, [15] presented the SHINE schemes, permutation-based constructions that achieve a robust detIND-UE-CCA security notion in the ideal cipher model, also relying on the DDH assumption.

Transitioning to post-quantum cryptography, Jiang [16] introduced LWEUE, the first UE scheme secured under the Learning With Errors (LWE) assumption. Later, [17] proposed the RtR scheme, another LWE-based UE construction. Notably, RtR was the first ciphertext-independent UE scheme, designed to prevent adversaries from deriving the new key using both the update token and the old key. Nishimaki demonstrated that this property provides stronger security compared to schemes lacking such independence. Unlike RtR, the UE schemes presented in this work do not include this feature.

Both LWEUE and RtR leverage homomorphic operations [18], [19] to re-randomize updated ciphertexts. However, this approach introduces two major limitations. First, ciphertext noise accumulates with each key update, restricting the schemes to a finite number of updates. Second, due to the homomorphic nature and the exposure of the update token, an adversary can craft ciphertexts corresponding to related messages. As a result, these schemes fail to achieve security under chosen ciphertext attacks and are instead limited to randIND-UE-CPA security.

Ciphertext-dependent UE, as presented in [20]–[22], generates tokens for the next ciphertexts. While this permits more efficient constructions, it needs the client (data owner) to download a section of the ciphertexts to compute the update token, so it depends on ciphertext size [23].

We try to overcome these issues by using the RSA cryptosystem to build a lightweight UE.

IV. Proposed protocol

The proposed protocol is applicable and very simple, it consists of generating a new RSA pair key, multiplying the previous private key by the new public key, and this obtained value represents the basic token. After adding a random number of $\phi(n)$, a new token is calculated, finally, the cloud raises all ciphertexts to the power of T (Figure).

A. Keygen Algorithm

Like the original RSA, Keygen Algorithm selects two random numbers p and q, computes n and $\phi(n)$, then the initial pair key (e, d) for public encryption and private decryption respectively (Algorithm 1).

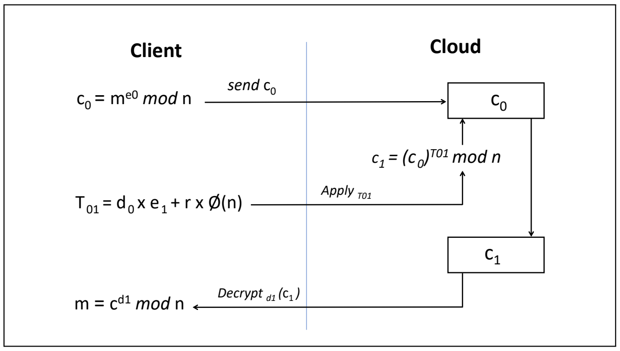

Fig. 2. RSA based proposed updatable encryption technique

Algorithm 1 Keygen

```
Require:
function KG
    $pick\ two\ large\ numbers\ p\ and\ q$
    $n \leftarrow p \times q$
    $\phi \leftarrow (p-1) \times (q-1)$
    $pick\ randomly\ a\ number\ e_0\ (enc\ public\ key)$
    $d_0 \leftarrow (e_0^-)_\phi\ ,\quad (dec\ key)$
    return $(n, \phi, e_0, d_0)$
end function
```

B. Client Update Algorithm

Algorithm 2 shows this process, updating function requires the previous decryption key d_i. After generating the next key pair (e_{i+1}, d_{i+1}), it multiplies the current Dec key by the new Enc key e_{i+1} and adds to them a random number of ϕ to prevent anyone from getting the old private key d_i because e_{i+1} is public. The calculated value called Token $T_{i,i+1}$ to pass from $Enc - Dec_i$ to $Enc - Dec_{i+1}$.

Algorithm 2 Client Update Algorithm

```
Require: $d_i, \phi$
function CUA
    $pick\ randomly\ a\ number\ e_{i+1}\ (enc\ public\ key)$
    $d_{i+1} \leftarrow ((e_{i+1})^-)_\phi\ ,\quad (dec\ key)$
    $generate\ a\ random\ numbber\ r$
    $Token\ T_{i,i+1} \leftarrow d_i \times e_{i+1} + r \times \phi$
    return $(T_{i,i+1}, e_{i+1}, d_{i+1})$
end function
```

C. Cloud update Algorithm

As shown in Algorithm 3, the cloud needs only the new token $T_{i,i+1}$ to update the encryption. The cloud raises all stored ciphertexts to $T_{i,i+1}$.

Algorithm 3 Cloud Update Algorithm

```
Require: $T_{i,i+1}$
function CDUA
    $for\ all\ ciphertexts\ c_{x,i}\ (encrypted\ with\ e_i)$
    $c_{x,i+1} \leftarrow (c_{x,i})^{T_{i,i+1}} \mod n$
    return $c_{x,i+1}$
end function
```

D. Proposal correctness

Our proposal is defined by Algorithms 1, 2, and 3.

Let $c_i = m^{e_i} \mod n$ be an encryption number i, when updating, the cloud computes $c_{i+1} = (c_i)^{T_{i,i+1}} \mod n$. If $T_{i,i+1} = d_i \times e_{i+1} + r \times \phi$, this gives:

$c_{i+1} = (c_i)^{d_i \times e_{i+1} + r \times \phi} \mod n$
$= (m)^{e_i \times (d_i \times e_{i+1} + r \times \phi)} \mod n$
$= (m)^{e_i \times d_i \times e_{i+1} + e_i \times r \times \phi} \mod n$

sine $m^{e_i \times d_i} \mod n = m$ and $m^{e_i \times r \times \phi} \mod n = 1$, therefore:

$c_{i+1} = m^{e_{i+1}} \mod n$, which gives:
$m = c^{d_{i+1}} \mod n$.

V. Analysis and performance

UE schemes can be categorized based on their dependence on ciphertext during token generation. The first category includes ciphertext-dependent protocols, where clients must either pre-store or retrieve ciphertext, or parts of it, from the cloud to compute tokens. The second category consists of ciphertext-independent protocols, where clients can generate tokens without needing access to the ciphertext. Our proposed approach adopts the ciphertext-independent protocol.

UE techniques can also be classified by how ciphertext updates are applied. In deterministic update (det-update) schemes, the update algorithm is deterministic, ensuring that the same ciphertext is always updated to the same output when using the token. In contrast, randomized update (rand-update) schemes employ a randomized algorithm, resulting in different ciphertexts being generated from the same original ciphertext through the token. Our approach utilizes the rand-update technique.

Finally, UE schemes can be differentiated by the directionality of their updates. In bidirectional update schemes, clients can both update and revert (downgrade) ciphertext using the token. Conversely, unidirectional update schemes only allow ciphertext updates, with no ability to revert to a previous version. In our proposed scheme, clients have the flexibility to either update or downgrade ciphertext using the token.

With respect to key update directions, UE schemes are classified into many (4) types [24]. The bidirectional key update technique, where the client can derive the new key (k_{e+1}, e for epoch) from the previous key (k_e) and the token (T_{e+1}), or reverse the process to derive k_e using k_{e+1} and T_{e+1}. The forward-leak unidirectional key modification technique (f-uni-directional), where clients can only derive k_{e+1} based on k_e and T_{e+1}. The the backward-leak unidirectional key update technique (b-uni-directional), where clients can only derive k_e

using k_{e+1} and T_{e+1}. The no-directional key update technique is where keys cannot be derived based on the token. In our proposal, the novel key k_{e+1} is generated randomly and is involved in computing T_{e+1} ($T_{e,e+1}$).

Regarding the size of the updated CT, we defined two types of UE. The first is Leveled UE, where encryption noise increases with each update, resulting in a finite number of updates that the schemes support. The second is Fully UE which allows the evaluation of arbitrary circuits with an unbounded depth i.e., an unlimited number of updates. Our technique is of the second type.

Keys security relies upon RSA cryptosystem, the attacker needs to factorize n to p and q. In updating, the token T is the addition of two random numbers ($d_i \times e_{i+1}$ and $r \times \phi$), so it is not possible to extract any private information from T. On the other hand, the cloud has m^{e_i} and $m^{e_{i+1}}$ where e_i and e_{i+1} are random and have no relation between them, therefore, no sensible data will be leaked in updating process.

TABLE I
COMPARISON OF PERFORMANCE METRICS, WHERE P, Q: 1024 BITS, N: 2048 BITS

Operation	**Our proposal**	**Ref. [24] (2024)**
Encryption	0.00047 s	0.00016 s
Token generation time	0.00011 s	0.0014 s
CT update time	0.0015 s	0.0016 s
Decryption	0.0010 s	0.0000517 s

Table I shows the comparison of some performance factors. The test platforms were as follows. Ours: CPU: Intel Core i7-10610U 2.30 GHz, Ref. [24]: CPU: Inter Core i5-7200U 2.50GHz.

VI. CONCLUSION AND FUTURE WORK

In this work, we have introduced a novel RSA-based updatable encryption scheme designed to streamline the key rotation process in secure communication systems. By leveraging the mathematical properties of the RSA cryptosystem, our scheme enables ciphertexts to be securely updated from a previous key to a new key without decrypting the underlying plaintext. This feature ensures both forward and backward security, mitigating risks associated with key compromise. Our security analysis confirms that the proposed scheme is not vulnerable to information leaking, meeting the high-security standards required for modern encryption protocols. Experimental evaluations demonstrate that the proposed method is efficient and practical for large-scale systems, imposing minimal overhead on computational resources and system performance during key updates. The RSA-based updatable encryption scheme presented here offers a compelling solution for organizations requiring regular key rotations without sacrificing security or efficiency. Future research may focus on optimizing the scheme further and making it post-quantum encryption.

REFERENCES

[1] B. Pahlevanzadeh, S. Koleini, and S. I. Fadilah, "Security in iot: Threats and vulnerabilities, layered architecture, encryption mechanisms, challenges and solutions," in *International Conference on Advances in Cyber Security*. Springer, 2020, pp. 267–283.

[2] M. E. H. Kahla, M. Beggas, A. Laouid, M. AlShaikh, and M. Hammoudeh, "An iomt image crypto-system based on spatial watermarking and asymmetric encryption," *Multimedia Tools and Applications*, pp. 1–26, 2024.

[3] C. Zhang, Y. Liang, A. Tavares, L. Wang, T. Gomes, and S. Pinto, "An improved public key cryptographic algorithm based on chebyshev polynomials and rsa," *Symmetry*, vol. 16, no. 3, p. 263, 2024.

[4] H. N. Noura, A. Chehab, and R. Couturier, "Overview of efficient symmetric cryptography: dynamic vs static approaches," in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*. IEEE, 2020, pp. 1–6.

[5] R. Adee and H. Mouratidis, "A dynamic four-step data security model for data in cloud computing based on cryptography and steganography," *Sensors*, vol. 22, no. 3, p. 1109, 2022.

[6] X. Wang, K. Zhang, J. Gong, S.-F. Sun, and J. Ning, "Updatable searchable symmetric encryption: Definitions and constructions," *Theoretical Computer Science*, vol. 983, p. 114304, 2024.

[7] A. Leroux and M. Roméas, "Updatable encryption from group actions," in *International Conference on Post-Quantum Cryptography*. Springer, 2024, pp. 20–53.

[8] M. Kara, K. Karampidis, L. Kenioua, M. Karampidis, M. AlShaikh, G. Papadourakis, and A. Laouid, "An authenticated method for a secure changing password," in *2024 5th International Conference in Electronic Engineering, Information Technology & Education (EEITE)*. IEEE, 2024, pp. 1–6.

[9] M. Kara, K. Karampidis, G. Papadourakis, M. Hammoudeh, and M. AlShaikh, "An enhanced learning with error-based cryptosystem: A lightweight quantum-secure cryptography method," *J*, vol. 7, no. 4, pp. 406–420, 2024.

[10] K. Chait, A. Laouid, M. Kara, M. Hammoudeh, O. Aldabbas, and A. T. Al-Essa, "An enhanced rsa-based aggregate signature scheme to reduce blockchain size," *IEEE Access*, 2023.

[11] S. Medileh, M. Kara, A. Laouid, A. Bounceur, and I. Kertiou, "A secure clock synchronization scheme in wsns adapted for iot-based applications," in *Proceedings of the 7th International Conference on Future Networks and Distributed Systems*, 2023, pp. 674–681.

[12] D. Boneh, K. Lewi, H. Montgomery, and A. Raghunathan, "Key homomorphic prfs and their applications," in *Annual Cryptology Conference*. Springer, 2013, pp. 410–428.

[13] A. Lehmann and B. Tackmann, "Updatable encryption with post-compromise security," in *Advances in Cryptology–EUROCRYPT 2018: 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29-May 3, 2018 Proceedings, Part III 37*. Springer, 2018, pp. 685–716.

[14] M. Klooß, A. Lehmann, and A. Rupp, "(r) cca secure updatable encryption with integrity protection," in *Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part I 38*. Springer, 2019, pp. 68–99.

[15] C. Boyd, G. T. Davies, K. Gjøsteen, and Y. Jiang, "Fast and secure updatable encryption," in *Annual International Cryptology Conference*. Springer, 2020, pp. 464–493.

[16] Y. Jiang, "The direction of updatable encryption does not matter much," in *Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part III 26*. Springer, 2020, pp. 529–558.

[17] R. Nishimaki, "The direction of updatable encryption does matter," in *IACR International Conference on Public-Key Cryptography*. Springer, 2022, pp. 194–224.

[18] M. Kara, A. Laouid, R. Euler, M. A. Yagoub, A. Bounceur, M. Hammoudeh, and S. Medileh, "A homomorphic digit fragmentation encryption scheme based on the polynomial reconstruction problem," in *Proceedings of the 4th International Conference on Future Networks and Distributed Systems*, 2020, pp. 1–6.

[19] M. AlShaikh, M. Alzaqebah, N. Gmati, N. Alrefai, M. K. Alsmadi, I. Almarashdeh, R. M. A. Mohammad, S. Alamri, and M. Kara, "Image encryption algorithm based on factorial decomposition," *Multimedia Tools and Applications*, pp. 1–21, 2024.

[20] H. Chen, Y. J. Galteland, and K. Liang, "Cca-1 secure updatable encryption with adaptive security," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2023, pp. 374–406.

[21] L. Chen, Y. Li, and Q. Tang, "Cca updatable encryption against malicious re-encryption attacks," in *Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part III 26*. Springer, 2020, pp. 590–620.

[22] A. Everspaugh, K. Paterson, T. Ristenpart, and S. Scott, "Key rotation for authenticated encryption," in *Advances in Cryptology–CRYPTO 2017: 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20–24, 2017, Proceedings, Part III 37*. Springer, 2017, pp. 98–129.

[23] R. Kebache, A. Laouid, A. Bounceur, M. Kara, K. Karampidis, G. Papadourakis, and M. Hammoudeh, "Reducing the encrypted data size: Healthcare with iot-cloud computing applications." *Computer Systems Science & Engineering*, vol. 48, no. 4, 2024.

[24] Y. Song, H. Gao, S. Wang, C. Ma, and K. Sun, "Token open secure and practical ntru-based updatable encryption," *The Journal of Supercomputing*, pp. 1–34, 2024.

[redacted]